

# **TALYS-1.96/2.0**

Simulation of nuclear reactions

**Arjan Koning, Stephane Hilaire, Stephane Goriely**

Copyright © 2021 Arjan Koning

NDS.IAEA.ORG/TALYS

TALYS is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version. This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License in Appendix I for more details.

Typesetting: The Legrand Orange Book, LaTeX Template, Version 2.1.1 (14/2/16), downloaded from: [www.LaTeXTemplates.com](http://www.LaTeXTemplates.com). Original author: Mathias Legrand ([legrand.mathias@gmail.com](mailto:legrand.mathias@gmail.com)) with modifications by: Vel ([vel@latextemplates.com](mailto:vel@latextemplates.com)). License: CC BY-NC-SA 3.0 [creativecommons.org/licenses/by-nc-sa/3.0/](http://creativecommons.org/licenses/by-nc-sa/3.0/).

*First printing, December 2021*

## About the author



Arjan Koning is a nuclear physicist with a Masters Degree in Theoretical physics at the Univ. of Amsterdam, a PhD in the Natural Sciences on Multi-step direct reactions at the Univ. of Groningen, and a professorship at the Univ. of Uppsala on theoretical nuclear reactions.

Arjan is currently Head of the Nuclear Data Section at the IAEA in Vienna. Before that, he has worked at ECN/NRG Petten, the Netherlands, on nuclear reaction data for science and technology, and as guest scientist at CEA/Bruyeres-le-Chatel and Los Alamos National Laboratory on the development of nuclear reaction models. He has led various students to PhD degrees, has coordinated and chaired various international nuclear data projects such as the OECD/NEA JEFF and WPEC projects, and has advised governments and international organisations on nuclear research and development. Among his scientific accomplishments are innovations in nuclear reaction physics, especially the optical model and pre-equilibrium reactions, the TALYS nuclear model code, Total Monte Carlo uncertainty propagation and the TENDL nuclear data library. As of end-2021, his h-index is 54 with more than 16 000 citations.

Although Arjan has finally accepted leadership offers that he could no longer refuse, he aims to keep his scientific creativity alive by maintaining and extending TALYS + all products that emerge from that. Pleas from his friends to also spend time on other things are sometimes honored.



## Preface

TALYS is a nuclear reaction program with the objective to provide a complete and accurate simulation of nuclear reactions up to energies of 200 MeV, through an optimal combination of reliable nuclear models, flexibility and user-friendliness. TALYS can be used for the analysis of basic nuclear reaction experiments or to generate nuclear data for applications.

The idea to make TALYS was born in 1998, when I was visiting CEA Bruyères-le-Châtel and Stephane Hilaire and we decided to implement our combined knowledge of nuclear reactions into one single software package. Since then, virtually all TALYS development has taken place at NRG Petten, the Netherlands, CEA Bruyères-le-Châtel, France, University Libre, Brussels, and in recent years at the IAEA in Vienna.

After 20 years, thousands of users, and more than 6000 publications in which TALYS is used, I found it time to produce an entirely new version of the code, rewritten in Fortran-95. At the same time, this entirely new tutorial is made available, with a more logical structure than the old TALYS manual.

Like most scientific projects, TALYS is always under development. Nevertheless, at certain moments in time, we freeze a well-defined version of TALYS and subject it to extensive verification and validation procedures. You are now reading the tutorial which applies to both TALYS-1.96, the usual next version in the row of TALYS-1.xx codes, and version 2.0, which employs with more modern Fortran specifications.

This part of my work I consider as truly open source: I try to minimize the difference between what I know and what I provide to others. At the start of my career, that meant releasing all knowledge in publications. Nowadays, that means releasing all knowledge in publications and in software, if possible quality-assured, retraceable and user-friendly. I aim to do that and I have spent a significant amount of my available time recycling software of myself and others, and the tutorial, to produce something that is hopefully sustainable.

As a PhD student, and also later, I worked myself through theory books of authors like my professor Hans Weidenmüller, Peter Hodgson (who pushed my early career) and Ray Satchler and

I wondered whether I would once be in the position to write a similar book. It did not happen, other priorities prevailed and being employed for my whole career by research and international organisations instead of a university did not help either. Still, one could read this tutorial as a sort of modern nuclear physics "book". Many essential nuclear models will be discussed, with the extra feature (compared to old-style text books) that they are all accompanied with TALYS sample cases and illustrations.

## License, contact and reference

As mentioned on the first page, in the source code and detailed in the last Appendix, TALYS falls in the category of GNU General Public License software.

In addition to the GNU GPL *terms* I have a few *requests*:

- When TALYS is used for your reports, publications, etc., please make a proper reference to the code. At the moment this is:  
A.J. Koning, S. Hilaire and M.C. Duijvestijn, "TALYS-1.0", *Proceedings of the International Conference on Nuclear Data for Science and Technology*, April 22-27, 2007, Nice, France, editors O.Bersillon, F.Gunsing, E.Bauge, R.Jacqmin, and S.Leray, EDP Sciences, 2008, p. 211-214.
- Please inform me about, or send, extensions you have built into TALYS. Of course, proper credit will be given to the authors of such extensions in future versions of the code.

Although I have invested a lot of effort in the validation of our code, I will not make the mistake to guarantee perfection. Therefore, in exchange for the free use of TALYS: If you find any errors, or in general have any comments, corrections, extensions, questions or advice, I would like to hear about it at [A.Koning@iaea.org](mailto:A.Koning@iaea.org). The webpage for TALYS is [nds.iaea.org/talys](http://nds.iaea.org/talys).

## Acknowledgements

Many people have contributed to the present state of TALYS:

- Stephane Hilaire, TALYS co-author from the start. Width fluctuation corrections, the basic fission transmission coefficients, the Blatt-Biedenharn formalism for compound nucleus angular distributions and the recoil formalism are all in TALYS thanks to him, and later he provided almost all microscopic Gogny-Hartree-Fock-Bogoliubov based tables for nuclear structure,
- Stephane Goriely is another TALYS co-author, who has contributed to the development of many reaction models in TALYS, especially for gamma-ray strength functions and fission (and much more), and those models which have an impact on astrophysical reaction rate calculations,
- Marieke Duijvestijn, another contributor from the first TALYS years, with her development of the formalism for exclusive particle spectra and the first model for fission yield prediction (her PhD work),
- Dimitri Rochman for the input of resonance parameters to TALYS and his collaboration with me on the TENDL nuclear data library and Total Monte Carlo uncertainty quantification, two of the most important applications of TALYS,
- Jacques Raynal (who unfortunately passed away in 2020) for extending the coupled-channels code ECIS according to our special wishes, so we could easily include it as a subroutine,
- Jean-Paul Delaroche for his collaboration on the optical model and direct reaction mechanisms,
- Olivier Bersillon (who unfortunately is no longer with us) for support during the initial stages of TALYS development, when I was working in CEA Bruyeres-le-Chatel,
- Emil Betak, Vivian Dimitriou and Connie Kalbach for input on the pre-equilibrium models,

- Eric Bauge for extending the optical model possibilities of TALYS with his JLM contribution and advice on uncertainty quantification,
- Pascal Romain, for helpful advice on the proper use of ECIS,
- Roberto Capote and Mihaela Sin for their contribution on fission models,
- Steven van der Marck for helping to set up the first release of TALYS, including a critical reading of the manual,
- Yi Xu for adding the direct capture model of him and Stephane Goriely to TALYS,
- Gilles Noguere for a better implementation of the unresolved resonance range model,
- Natalia Dzysiuk for parameter optimization to produce better TENDL files,
- Vasily Simutkin and Michail Onegin to help implementing the GEF model for fission yields by Schmidt and Jurado,
- Karl-Heinz Schmidt for helpful advice on his GEF code.
- Vlad Avrigeanu for his contribution on the alpha optical model,
- Marilena Avrigeanu for providing better models for deuteron reactions,
- Emmeric Dupont and Michael Borchard for specific computational advice and code extensions,
- Arjan Plompen, Jura Kopecky and Robin Forrest for testing many of the results of TALYS,
- Mark Chadwick and Mike Herman for helpful discussions on the reaction models of TALYS,
- Toshihiko Kawano for many helpful discussions on the reaction models of TALYS, and those for fission yields and PFNS in particular.
- Stephan Pomp and Ali Al-Adili for their collaboration of fission yields,
- Jean-Francois Lemaitre for his in-depth study of the fission fragment evaporation model in TALYS and his many suggestions for improvements,
- Shin Okumura and Kazuki Fujio for their tests of the quality of TALYS for fission yield predictions,
- Henrik Sjöstrand and Petter Helgesson for their work on uncertainty propagation,
- Sandor Sudar for investigating deviations of TALYS from high-energy residual production cross section measurements and for coming up with a solution.
- Jean-Christophe Sublet for his contribution to the TENDL library,
- numerous users who sent me emails with suggestions and bug fixes.
- Finally, this tutorial is dedicated to the memory of Phil Young who showed me that it is indeed possible to write an all-in-one nuclear model code like his GNASH-code (which included the first good code manual), which helped start the TALYS project.

Probably I have forgotten people who deserve to be in the above list. In that case, let me know!

Arjan Koning





# Contents

<b>I</b>	<b>Part One: TALYS General</b>	
<b>1</b>	<b>Introduction</b> .....	<b>17</b>
1.1	What TALYS can not do	20
1.2	From TALYS-1.95 to TALYS-1.96/TALYS-2.0	20
1.3	How to use this tutorial	21
<b>2</b>	<b>Installation and getting started</b> .....	<b>23</b>
2.1	The TALYS package	23
2.2	Installation	23
2.3	Verification	24
2.4	Getting started	24
<b>3</b>	<b>Input description</b> .....	<b>25</b>
3.1	Basic input rules	26
3.2	Basic input	27
3.2.1	Energy-dependent parameter adjustment .....	28
<b>4</b>	<b>Nuclear reactions: General approach</b> .....	<b>31</b>
4.1	Reaction mechanisms	34
4.1.1	Low energies .....	34
4.1.2	High energies .....	37
4.2	Cross section definitions	66
4.2.1	Total cross sections .....	66

4.2.2	Exclusive cross sections	67
4.2.3	Binary cross sections	70
4.2.4	Total particle production cross sections	71
4.2.5	Residual production cross sections	75
<b>4.3</b>	<b>Spectra and angular distributions</b>	<b>80</b>
4.3.1	Discrete angular distributions	80
4.3.2	Exclusive spectra	82
4.3.3	Binary spectra	83
4.3.4	Total particle production spectra	83
4.3.5	Double-differential cross sections	84
<b>4.4</b>	<b>Fission cross sections</b>	<b>86</b>
<b>4.5</b>	<b>Recoils</b>	<b>91</b>
4.5.1	Qualitative analysis	91
4.5.2	General method	92
4.5.3	Quantitative analysis	92
4.5.4	The recoil treatment in TALYS	93
4.5.5	Method of average velocity	95
4.5.6	Approximative recoil correction for binary ejectile spectra	95
<b>5</b>	<b>Nuclear structure and model parameters</b>	<b>99</b>
<b>5.1</b>	<b>General setup of the database</b>	<b>99</b>
<b>5.2</b>	<b>Integral validation library</b>	<b>99</b>
<b>5.3</b>	<b>Decay data library</b>	<b>100</b>
<b>5.4</b>	<b>Best TALYS input parameters</b>	<b>100</b>

## II

## Part Two: Nuclear Models

<b>6</b>	<b>Nuclear masses and deformations</b>	<b>107</b>
6.1	Nuclear masses	107
6.2	Deformation parameters	109
6.3	Isotopic abundances	110
<b>7</b>	<b>Discrete levels</b>	<b>111</b>
7.1	Discrete level file	111
7.2	Deformation parameters	113
<b>8</b>	<b>Level densities</b>	<b>117</b>
<b>8.1</b>	<b>Effective level density</b>	<b>118</b>
8.1.1	The Fermi Gas Model	118
8.1.2	The level density parameter $a$	119
8.1.3	The spin cut-off parameter	122
8.1.4	Constant Temperature Model	124
8.1.5	The Back-shifted Fermi gas Model	126
8.1.6	The Generalized Superfluid Model	127

<b>8.2</b>	<b>Collective effects in the level density</b>	<b>129</b>
<b>8.3</b>	<b>Microscopic level densities</b>	<b>133</b>
<b>8.4</b>	<b>Average resonance parameters</b>	<b>134</b>
<b>9</b>	<b>Reactions involving photons</b>	<b>139</b>
<b>9.1</b>	<b>Photon strength functions and transmission coefficients</b>	<b>139</b>
<b>9.2</b>	<b>Models for photon strength functions</b>	<b>140</b>
9.2.1	Brink-Axel model	141
9.2.2	Kopecky-Uhl model	141
9.2.3	Skyrme-Hartree-Fock BCS model	142
9.2.4	Skyrme-Hartree-Fock-Bogoliubov model	142
9.2.5	Hybrid model	142
9.2.6	Temperature-dependent Skyrme-Hartree-Fock-Bogoliubov model	142
9.2.7	Temperature-dependent Relativistic Mean Field model	143
9.2.8	Gogny-Hartree-Fock-Bogoliubov model	143
<b>9.3</b>	<b>Renormalization of gamma-ray strength functions</b>	<b>145</b>
<b>9.4</b>	<b>Photoabsorption cross section</b>	<b>146</b>
<b>9.5</b>	<b>Direct capture</b>	<b>147</b>
<b>10</b>	<b>Optical model</b>	<b>149</b>
<b>10.1</b>	<b>Spherical OMP: Neutrons and protons</b>	<b>149</b>
10.1.1	Extension to 1 GeV	154
<b>10.2</b>	<b>Spherical dispersive OMP: Neutrons</b>	<b>155</b>
<b>10.3</b>	<b>Spherical OMP: Complex particles</b>	<b>158</b>
10.3.1	Deuterons	158
10.3.2	Tritons	160
10.3.3	Helium-3	160
10.3.4	Alpha particles	160
<b>10.4</b>	<b>Semi-microscopic optical model (JLM)</b>	<b>164</b>
<b>10.5</b>	<b>Systematics for non-elastic cross sections</b>	<b>167</b>
<b>11</b>	<b>Direct reactions</b>	<b>169</b>
<b>11.1</b>	<b>Deformed nuclei: Coupled-channels</b>	<b>169</b>
11.1.1	Symmetric rotational model	170
11.1.2	Harmonic vibrational model	170
11.1.3	Vibration-rotational model	171
11.1.4	Asymmetric rotational model	171
<b>11.2</b>	<b>Distorted Wave Born Approximation</b>	<b>174</b>
<b>11.3</b>	<b>Odd nuclei: Weak coupling</b>	<b>177</b>
<b>11.4</b>	<b>Giant resonances</b>	<b>177</b>
<b>12</b>	<b>Pre-equilibrium reactions</b>	<b>179</b>
<b>12.1</b>	<b>Exciton model</b>	<b>179</b>
12.1.1	Two-component exciton model	181
12.1.2	One-component exciton model	188

<b>12.2</b>	<b>Photon exciton model</b>	<b>192</b>
<b>12.3</b>	<b>Pre-equilibrium spin distribution</b>	<b>192</b>
<b>12.4</b>	<b>Continuum stripping, pick-up, break-up and knock-out reactions</b>	<b>193</b>
12.4.1	Transfer reactions . . . . .	193
12.4.2	Knockout reactions . . . . .	194
12.4.3	Break-up reactions . . . . .	195
<b>12.5</b>	<b>Angular distribution systematics</b>	<b>196</b>
<b>13</b>	<b>Compound reactions . . . . .</b>	<b>205</b>
<b>13.1</b>	<b>Binary compound cross section and angular distribution</b>	<b>205</b>
<b>13.2</b>	<b>Width fluctuation correction factor</b>	<b>208</b>
13.2.1	The HRTW method . . . . .	209
13.2.2	Moldauer expression . . . . .	210
13.2.3	The GOE triple integral . . . . .	212
<b>14</b>	<b>Multiple emission . . . . .</b>	<b>217</b>
<b>14.1</b>	<b>Multiple Hauser-Feshbach decay</b>	<b>217</b>
<b>14.2</b>	<b>Multiple pre-equilibrium emission</b>	<b>218</b>
14.2.1	Multiple pre-equilibrium emission within the exciton model . . . . .	219
14.2.2	Multiple pre-equilibrium emission with the s-wave transmission coefficient method	220
<b>15</b>	<b>Fission . . . . .</b>	<b>221</b>
<b>15.1</b>	<b>Level densities for fission barriers</b>	<b>221</b>
15.1.1	Explicit treatment of collective effects . . . . .	221
15.1.2	Effective treatment of collective effects . . . . .	222
<b>15.2</b>	<b>Fission transmission coefficients</b>	<b>222</b>
15.2.1	Transmission coefficient for one fission barrier . . . . .	222
<b>15.3</b>	<b>Transmission coefficient for multi-humped barriers</b>	<b>223</b>
<b>15.4</b>	<b>Class II/III states</b>	<b>223</b>
15.4.1	Double humped fission barrier . . . . .	224
15.4.2	Triple humped fission barrier . . . . .	224
<b>15.5</b>	<b>Fission barrier parameters</b>	<b>225</b>
<b>15.6</b>	<b>WKB approximation</b>	<b>225</b>
<b>15.7</b>	<b>Fission fragment properties</b>	<b>227</b>
15.7.1	GEF model . . . . .	227
15.7.2	Temperature-dependent Brosa model . . . . .	228
15.7.3	Fission fragment mass distribution . . . . .	229
15.7.4	Post-scission neutron multiplicities . . . . .	232
15.7.5	Fission fragment charge distribution . . . . .	233
<b>15.8</b>	<b>Implementation, options and sample cases</b>	<b>234</b>
<b>16</b>	<b>Thermal reactions . . . . .</b>	<b>247</b>
<b>16.1</b>	<b>Capture channel</b>	<b>247</b>
<b>16.2</b>	<b>Other non-threshold reactions</b>	<b>248</b>

16.3	Implementation, options and sample cases	248
17	Populated initial nucleus .....	253
18	Astrophysical reaction rates .....	255
19	Medical isotope production .....	261
19.1	Production and depletion of isotopes	261
19.2	Initial condition and stopping power	264
19.3	Nuclear reaction and decay rates	265
19.4	Radioactive activity	266
20	Verification and validation, sample cases and output .....	269
20.1	Robustness test with DRIP	269
20.2	Robustness test with MONKEY	269
20.3	Sample cases	270

### III

## Part Three: Reference Guide

21	Four main keywords .....	277
22	Keywords for basic and numerical parameters .....	285
23	Keywords for masses and deformations .....	329
24	Keywords for discrete levels .....	337
25	Keywords for level densities .....	351
26	Keywords for gamma emission .....	403
27	Keywords for optical model .....	447
28	Keywords for direct reactions and ECIS .....	517
29	Keywords for compound nucleus .....	547
30	Keywords for pre-equilibrium reactions .....	565
31	Keywords for fission .....	603
32	Keywords for astrophysics .....	641
33	Keywords for medical isotope production .....	649

34	Keywords for output .....	661
35	Keywords of TALYS .....	687

**IV**

**Part Four: Miscellaneous**

36	Development of TALYS .....	701
36.1	From TALYS-1.0 to TALYS-1.2	701
36.2	From TALYS-1.2 to TALYS-1.4	702
36.3	From TALYS-1.4 to TALYS-1.6	703
36.4	From TALYS-1.6 to TALYS-1.8	704
36.5	From TALYS-1.8 to TALYS-1.9	705
36.6	From TALYS-1.9 to TALYS-1.95	705
37	Outlook and conclusions .....	707
	Bibliography .....	708
A	Log file of changes since TALYS-1.0 .....	717
B	Log file of changes since TALYS-1.2 .....	729
C	Log file of changes since TALYS-1.4 .....	737
D	Log file of changes since TALYS-1.6 .....	745
E	Log file of changes since TALYS-1.8 .....	751
F	Log file of changes since TALYS-1.9 .....	755
G	Log file of changes since TALYS-1.95 .....	759
H	Derivation of isotope production equations .....	765
I	GNU GENERAL PUBLIC LICENSE .....	767
I.1	Preamble	767
I.2	Terms and Conditions	768



# Part One: TALYS General

<b>1</b>	<b>Introduction .....</b>	<b>17</b>
<b>2</b>	<b>Installation and getting started .....</b>	<b>23</b>
<b>3</b>	<b>Input description .....</b>	<b>25</b>
<b>4</b>	<b>Nuclear reactions: General approach</b>	<b>31</b>
<b>5</b>	<b>Nuclear structure and model parameters</b>	<b>99</b>





# 1. Introduction

TALYS is a computer program for the analysis and prediction of nuclear reactions. The basic objective behind its construction is the simulation of nuclear reactions that involve neutrons, photons, protons, deuterons, tritons,  $^3\text{He}$ - and alpha-particles, in the eV - 200 MeV energy range and for target nuclides of mass 12 and heavier. To achieve this, we have implemented a suite of nuclear reaction models into a single code system. This enables us to evaluate nuclear reactions from the resonance range up to intermediate energies.

There are two main purposes of TALYS, which are strongly connected. First, it is a *nuclear physics* tool that can be used for the analysis of nuclear reaction experiments. The interplay between experiment and theory gives us insight in the fundamental interaction between particles and nuclei, and precise measurements enable us to constrain our models. In return, when the resulting nuclear models are believed to have sufficient predictive power, they can give an indication of the reliability of measurements. The many sample cases we present throughout this tutorial confirm that this software project would be nowhere without the existing (and future) experimental database.

After the nuclear physics stage comes the second function of TALYS, namely as a *nuclear data* tool: Either in a default mode, when no measurements are available, or after fine-tuning the adjustable parameters of the various reaction models using available experimental data, TALYS can *generate* nuclear data for all open reaction channels, on a user-defined energy and angle grid. The nuclear data libraries that are constructed with these calculated and experimental results provide essential information for existing and new nuclear technologies. Important applications that rely directly or indirectly on data generated by nuclear reaction simulation codes like TALYS are: conventional and innovative nuclear power reactors, transmutation of radioactive waste, fusion reactors, accelerator applications, homeland security, medical isotope production, radiotherapy, single-event upsets in microprocessors, oil-well logging, geophysics and astrophysics.

In its first decade, the development of TALYS used to follow the “first completeness, then quality” principle. With this, we certainly never suggested that we use toy models to arrive at some quick and dirty results since several reaction mechanisms coded in TALYS are based on theoretical

models whose implementation is only possible with the current-day computer power. It rather means that, in our quest for completeness, we try to divide our effort equally among all nuclear reaction types. The precise description of *all* possible reaction channels in a single calculational scheme is such an enormous task that we have chosen, to put it bluntly, not to devote several years to the theoretical research and absolutely perfect implementation of one particular reaction channel which accounts for only a few millibarns of the total reaction cross section. Instead, we aim to enhance the quality of TALYS equally over the whole reaction range and always search for the largest shortcoming that remains after the last improvement. We now think that “completeness and quality” has been accomplished for several important parts of the program. The reward of this approach is that with TALYS we can cover the whole path from fundamental nuclear reaction models to the creation of complete data libraries for nuclear applications, with the obvious side note that the implemented nuclear models will always need to be upgraded using better physics. An additional long-term aim is full transparency of the implemented nuclear models, in other words, an *understandable* source program, and a modular coding structure.

The idea to construct a computer program that gives a simultaneous prediction of many nuclear reaction channels, rather than a very detailed description of only one or a few reaction channels, is not new. Well-known examples of all-in-one codes from the past century are GNASH [1], ALICE [2], STAPRE [3], and EMPIRE [4]. They have been, and are still, extensively used, not only for academical purposes but also for the creation of the nuclear data libraries that exist around the world. EMPIRE is still maintained and extended by the original authors, whereas various local versions of GNASH, ALICE and STAPRE exist around the world, all with different extensions and improvements. TALYS is newer in the sense that it has been written completely from scratch in the 21st century (with the exception of one very essential module, the coupled-channels code ECIS), using a consistent set of programming procedures.

As specific features of the TALYS package we mention

- In general, an exact implementation of many of the latest nuclear models for direct, compound, pre-equilibrium and fission reactions.
- A continuous, smooth description of reaction mechanisms over a wide energy and mass range. A safe statement is that these limits are 0.001- 200 MeV and  $12 < A < 339$ , but it is formally possible to go beyond that.
- Completely integrated optical model and coupled-channels calculations by the ECIS-06 code [5].
- Incorporation of recent optical model parameterisations for many nuclei, both phenomenological (optionally including dispersion relations) and microscopical.
- Total and partial cross sections, energy spectra, angular distributions, double-differential spectra and recoils.
- Discrete and continuum photon production cross sections.
- Excitation functions for residual nuclide production, including isomeric cross sections.
- An exact modeling of exclusive channel cross sections, e.g.  $(n, 2np)$ , spectra, and recoils.
- Automatic reference to nuclear structure parameters as masses, discrete levels, resonances, level density parameters, deformation parameters, fission barrier and gamma-ray parameters, generally from the IAEA Reference Input Parameter Library [6].
- Various width fluctuation models for binary compound reactions and, at higher energies, multiple Hauser-Feshbach emission until all reaction channels are closed.
- Various phenomenological and microscopic level density models.
- Various fission models to predict cross sections and fission fragment and product yields, neutron multiplicities and prompt fission neutron and gamma spectra.
- Models for pre-equilibrium reactions, and multiple pre-equilibrium reactions up to any order.
- Generation of parameters for the unresolved resonance range.

- Reconstruction of resonance range into pointwise cross sections using tabulated resonance parameters.
- Astrophysical reaction rates using Maxwellian averaging.
- Option to start with an excitation energy distribution instead of a projectile-target combination, helpful for coupling TALYS with intranuclear cascade codes or fission fragment studies.
- Use of systematics if an adequate theory for a particular reaction mechanism is not yet available or implemented, or simply as a predictive alternative for more physical nuclear models.
- Medical isotope production yields as a function of accelerator energy and beam current.
- Output files for automatic generation of nuclear data in ENDF-6 format using the TEFAL program.
- Flexible input and output to enable automatic optimization to experimental data and generation of covariance data by the TASMAN program.
- A transparent source program.
- Input/output communication that is easy to use and understand.
- An extensive tutorial.
- A large collection of sample cases.

The central message is that we always provide a complete set of answers for a nuclear reaction, for all open channels and all associated cross sections, spectra, angular distributions, etc. It depends on the current status of nuclear reaction theory, and our ability to implement that theory, whether these answers are generated by sophisticated physical methods or by a simpler empirical approach. With TALYS, a complete set of cross sections can already be obtained with minimal effort, through a four-line input file of the type:

```
projectile n
element    Fe
mass       56
energy     14.
```

which, if you are only interested in reasonably good answers for the most important quantities, will give you all you need. If you want to be more specific on nuclear models, their parameters and the level of output, you simply add some of the close to 400 keywords that can be specified in TALYS. We thus do not ask you to understand the precise meaning of all these keywords: you can make your input file as simple or as complicated as you want. Let us immediately stress that we realize the danger of this approach. This ease of use may give the obviously false impression that one gets a *good* description of *all* the reaction channels, with minimum reaction specification, as if we would have solved virtually all nuclear reaction problems (in which case we would have been famous). Unfortunately, nuclear physics is not that simple. Clearly, many types of nuclear reactions are very difficult to model properly and can not be expected to be covered by simple default values. Moreover, other nuclear reaction codes may outperform TALYS on particular tasks because they were specifically designed for one or a few reaction channels. In this light, the sample cases spread out over this tutorial are very important, as it contains many sample cases which should give the user an idea of what TALYS can do. We wish to mention that the above sketched method for handling input files was born out of frustration: We have encountered too many computer codes containing an implementation of beautiful physics, but with an unnecessary high threshold to use the code (if you could compile and run it at all), since its input files are supposed to consist of a large collection of mixed, and correlated, integer and real values, for which values *must* be given, forcing the user to first read the entire manual, which often does not exist.

## 1.1 What TALYS can not do

It maybe helpful to already restrict any extreme expectations in this Introduction. Yes, there are several important things in nuclear physics and nuclear data which TALYS can not handle. If TALYS can not do it, then either we have developed other software to take care of it, or it falls outside the scope of the physics included in TALYS, and software by other people should be used. Some of the restrictions of TALYS are the following

- The heaviest particle is an alpha-particle. Hence, TALYS is not (yet) a heavy-ion code.
- TALYS does not produce complete nuclear data libraries for technological applications. Another software package, TEFAL, which processes the output of TALYS, takes care of that.
- TALYS only provides the central values for a prediction. Full uncertainty quantification, based on random input parameter variation and leading to complete covariance matrices and more general probability distributions for all the output of TALYS, is handled by a program called TASMAN.
- Resonance reactions are not modeled. This is the domain of R-matrix theory and its more or less approximate versions and is handled by codes like SAMMY, REFIT or CONRAD. TALYS does however reconstruct cross sections from resonance parameters as provided by the TARES code.
- Light-nuclide physics, as implemented in R-matrix codes. TALYS is in essence a statistical model code, and that statistics starts to break down for systems with less than 20 nucleons.
- In terms of a complete nuclear reaction output, what is missing is a high-quality prediction of the average number of fission neutrons and the prompt fission neutron spectra. For this, separate codes, TAFIS and TANES, respectively, are available.
- An explicit microscopic treatment of stripping and pick-up mechanism for e.g. (d,p) reactions is not included. There are a few codes, like FRESCO and DEUTACS, which handle that.

## 1.2 From TALYS-1.95 to TALYS-1.96/TALYS-2.0

Until now, a new official version of TALYS is released every 2 years. For some users, it may be useful to see what has been added to the code since the last release. In Part IV the entire development of TALYS is summarized.

On December 30, 2019 the seventh official version of the code, TALYS-1.95, was released. Since then, the code has undergone changes that fall in the usual two categories: significant extensions and corrections that may affect a large part of the user community, and several small bug fixes. As appendix to this tutorial, we add the full log file of changes since the release of TALYS-1.95. Here we list the most important updates:

- Capability to read in the distribution of excited fission fragments, per fragment pair, after which TALYS calculated the fission product yield as well as all neutron observables such as  $\nu(A)$ , nubar, PFNS etc. Currently, three models for excited fission fragments are included: HF3D, GEF and SPY.
- Reading of the OMP for actinides from the Reference Input Parameter Library. These OMP's are now the default for neutrons incident on actinides.
- An improved deuteron break-up model by Marilena Avrigeanu.
- Change of the default of Hauser-Feshbach binning from **equidistant n** to **equidistant y**, which improves high energy excitation functions.
- Change of the default for subactinide fission. Now **fismodel 3** is used for charged-particle induced reactions on all targets and for neutrons on subactinides, which gives a dramatic

improvement for the fission cross sections.

- Change in the default adjustment parameters for break-up and stripping/pick-up reactions to improve charged-particle induced cross sections.
- Changed the default in TALYS for photon strength functions into **strength 9** (SMLO), and **strength1 3** with **upbend y**. This is applied for all incident particles, improving the default prediction for (n,g), (g,n), (p,g), (a,g) considerably. Automatic **wtable** parameters are read in for adjustment to capture cross sections in the fast neutron range.
- Phenomenological models for nubar (Wahl) and PFNS (Iwamoto).

For TALYS-1.96 the following new keywords were introduced: **betafiscoradjust**, see page 622, **block**, see page 664, **Cnubar1**, see page 633, **Cnubar2**, see page 634, **ffmodel**, see page 626, **fisom**, see page 431, **fispartdamp**, see page 614, **Fsadjust**, see page 628, **gadjust**, see page 584, **ldmodelCN**, see page 353, **ngfit**, see page 348, **nnfit**, see page 349, **nafit**, see page 350, **pfnsmodel**, see page 627, **riplomp**, see page 503, **riplrisk**, see page 504, **Rspincutff**, see page 376, **soukho**, see page 502, **Tmadjust**, see page 629, **vfiscoradjust**, see page 623, **yieldfile**, see page 639, while the possibilities of **fymodel**, see page 625, were extended.

### 1.3 How to use this tutorial

Although we would be honored if you would read this tutorial from the beginning to the end, we can imagine that not all parts are necessary, relevant or suitable to you, so here is a short overview.

1. Part I: TALYS general. This gives a global introduction of TALYS, including
  - Chapter 2: Installation guide.
  - Chapter 3: The general rules for the input file.
  - Chapter 4: A general discussion of nuclear reactions and the types of observables that can be obtained.
  - Chapter 5: A description of the various nuclear structure parameters that are used.
2. Part II: TALYS nuclear models for nuclear structure and reactions. This covers
  - Chapter 6: Nuclear masses and deformations
  - Chapter 7: Discrete levels
  - Chapter 8: Level densities
  - Chapter 9: Reactions involving photons
  - Chapter 10: Optical model
  - Chapter 11: Direct reactions
  - Chapter 12: Pre-equilibrium reactions
  - Chapter 13: Compound reactions
  - Chapter 14: Multiple emission
  - Chapter 15: Fission
  - Chapter 16: Thermal reactions
  - Chapter 17: Populated initial nucleus
  - Chapter 18: Astrophysical reaction rates
  - Chapter 19: Medical isotope production
  - Chapter 20: Verification and validation, sample cases and output
3. Part III: Reference Guide. A description of all keywords
4. Part IV: Miscellaneous items, like the historical development of TALYS, outlook and the bibliography.



## 2. Installation and getting started

### 2.1 The TALYS package

In what follows we assume TALYS will be installed on a Linux or MacOS operating system. The total TALYS package is in the *talys/* directory and contains the following directories and files:

- *README* outlines the contents of the package.
- *talys.setup* is a script that takes care of the installation.
- *source/* contains the source code of TALYS: Fortran subroutines.
- *structure/* contains the nuclear structure database in various subdirectories. See Chapter 5 for more information.
- *doc/* contains the documentation: this manual in pdf format and the description of ECIS-06.
- *samples/* contains the input and output files of the sample cases.

In total, you will need about 6 Gb of free disk space to install TALYS. This rather large amount of memory is almost completely due to microscopic level density, radial density, gamma and fission tables in the nuclear structure database. The code has so far been tested by us on various Unix/Linux systems, so we can not guarantee that it works on all operating systems, although we know that some users have easily installed TALYS under Windows. The only machine dependencies we can think of are the directory separators '/' we use in pathnames that are hardwired in the code. If there is any dependence on the operating system, the associated statements can be altered in the subroutine *machine.f*. Also, the output of the execution time in *timer.f* may be machine dependent. The rest of the code should work on any computer.

TALYS has been tested for the gfortran, Intel, Fujitsu and Portland Group compilers.

### 2.2 Installation

The installation of TALYS is straightforward. The first steps are

- Download *talys.tar* file to your hard disk.

- **tar zxf talys.tar**

We here provide the necessary steps to do the installation, For a Unix/Linux system, the installation is expected to be handled by the *talys.setup* script, as follows

- edit *talys.setup* and set the first two variables: the name of your compiler and the place where you want to store the TALYS executable.
- **talys.setup**

If this does not work for some reason, we here provide the necessary steps to do the installation manually. For a Unix/Linux or MacOS system, the following steps should be taken:

- **cd talys/source**
- Ensure that TALYS can read the nuclear structure database. This is done in subroutine *machine.f*. If *talys.setup* has not already replaced the path name in *machine.f*, do it yourself. We think this is the only Unix/Linux machine dependence of TALYS. Apart from a few trivial warning messages for *ecis06t.f*, we expect no complaints from the compiler.
- **gfortran -c \*.f (\*.f90 for the Fortran-95 version)**
- **gfortran \*.o -o talys**
- **mv talys ~/bin** (assuming you have a *~/bin* directory which contains all executables that can be called from any working directory)

The above commands represent the standard compilation options. Consult the manual of your compiler to get an enhanced performance with optimization flags enabled. The only restriction for compilation is that *ecis06t.f* should *not* be compiled in double precision.

## 2.3 Verification

If TALYS is installed, testing the sample cases is the logical next step. The *samples/* directory contains the script *verify* that runs all the test cases. Each sample case has its own subdirectory, which contains a subdirectory *org/*, where we stored the input files and **our** calculated results, obtained with the gfortran compiler on MacOS Catalina. It also contains a subdirectory *new*, where we have stored the input files only and where the *verify* script will produce **your** output files. A full description of the keywords used in the input files is given in Chapter 3. The sample cases are scattered over this tutorial. Note that in each subdirectory a file with differences with our original output is created.

Should you encounter error messages upon running TALYS, like *'killed'* or *'segmentation fault'*, then probably the memory of your processor is not large enough. In the TALYS-1.9xx version, edit *talys.cmb* and reduce the value of **memorypar**.

## 2.4 Getting started

If you have created your own working directory with an input file named e.g. *talys.inp*, then a TALYS calculation can easily be started with:

```
talys < talys.inp > talys.out
```

where the names *talys.inp* and *talys.out* are not obligatory: you can use any name for these files.



## 3. Input description

For people who want to go straight to work and will not wait for sample cases later in this tutorial, we give the most important input rules here. For the communication between TALYS and its users, we have constructed an input/output method which shields beginners from all the possible options for nuclear model parameters that can be specified in TALYS, while enabling at the same time maximal flexibility for experienced users.

An input file of TALYS consists of keywords and their associated values. We have devoted the entire part III of this tutorial to the input possibilities with the Reference Guide, but let us illustrate the use of the input by the following example. It represents a minimum input file for TALYS:

```
projectile n
element    al
mass      27
energy    14.
```

This input file represents the simplest question that can be asked to TALYS: if a  $^{27}\text{Al}$  nucleus is hit by 14 MeV neutrons, what happens? Behind this simple input file, however, there are more than a few hundred default values for the various nuclear models, parameters, output flags, etc., that you may or may not be interested in. When you use a minimal input file like the one above, you leave it to the authors of TALYS to choose all the parameters and models for you, as well as the level of detail of the output file. If you want to use specific nuclear models other than the default, adjust parameters or want to have more specific information in the output file(s), more keywords are required. Obviously, more keywords means more flexibility and, in the case of adequate use, better results, though often at the expense of increasing the level of phenomenology. In this Chapter, we will give the basic rules that must be obeyed when constructing an input file for TALYS, while part III specifies all the keywords, which have been categorised in several groups.

### 3.1 Basic input rules

Theoretically, it would be possible to make the use of TALYS completely idiot-proof, i.e. to prevent the user from any input mistakes that possibly can be made and to continue a calculation with “assumed” values after a typo or incomplete information. Although we have invested a relatively large effort in the user-friendliness of TALYS, we have not taken such safety measures to the extreme limit and ask at least some minimal responsibility from the user. Once you have accepted that, only very little effort is required to work with the program. Successful execution of TALYS can be expected if you remember the following simple rules and possibilities of the input file:

1. One input line contains one keyword. Usually it is accompanied by only one value, as in the simple example given above, but some keywords for model parameters need to be accompanied by indices (usually *Z* and *A*) on the same line.
2. A keyword and its value(s) *must* be separated by at least 1 blank character.
3. The keywords can be given in arbitrary order. If, by mistake, you use the same keyword more than once in an input file, the value of the last one will be adopted. This does not of course hold for keywords which are labeled with different *Z* and *A* indices, see the beginning of Section 35.
4. All characters can be given in either lowercase or uppercase.
5. A keyword *must* be accompanied by a value. (There is one exception, the **rotational** keyword). To use default values, the keywords should simply be left out of the input file.
6. An input line starting with a # in column 1 is neglected by TALYS. This is helpful for including comments in the input file or to temporarily deactivate keywords.
7. A minimal input file always consists of 4 lines and contains the keywords **projectile**, **element**, **mass** and **energy**. These 4 keywords *must* be given in any input file.

As an example of rules 2, 3, 4 and 6, it can be seen that the following input file is completely equivalent to the one given in the beginning of this Chapter:

```
# Equivalent input file
energy           14.
projectile n
mass 27
Element AL
#outbasic y
```

In the following erroneous input file, only the first 2 lines are correct, while rules 2 and 5 are violated in the other lines.

```
projectile n
element al
mass27
energy
```

In cases like this, the execution will be stopped and TALYS will give an appropriate error message for the first encountered problem. We like to believe that we have covered all such cases and that it is impossible to let TALYS crash (at least with our compilers, see also Chapter 20), but you are of course invited to prove and let us know about the contrary (Sorry, no cash rewards). Typing errors in the input file will be spotted by TALYS, e.g. if you write **projjectile n**, it will tell you the keyword is not in our list.

### 3.2 Basic input

The four-line input file given above was an example of a minimum input file for TALYS. In general, you probably want to be more specific, as in the following example:

```
projectile  n
element    nb
mass       93
energy     energies
Ltarget    1
relativistic n
widthmode  2
outinverse y
a 41 93 13.115
a 41 94 13.421
```

which will simulate the reaction of a neutron incident on  $^{93}\text{Nb}$ , for a number of incident energies given in the file *energies*, with the target in its first excited state (**Ltarget 1**, a 16-year isomer), using non-relativistic kinematics, the HRTW-model for width fluctuation corrections (**widthmode 2**) in the compound nucleus calculation, with the particle transmission coefficients and inverse reaction cross sections written on the output file (**outinverse y**) and with user-defined level density parameters **a** for  $^{93}\text{Nb}$  and  $^{94}\text{Nb}$ .

In Part III, we will explain all the possible keywords. We have classified them according to their meaning and importance. For each keyword, we give an explanation, a few examples, the default value, and the theoretically allowed numerical range. As the input file above shows, there is usually one value per keyword. Often, however, in cases where several residual nuclides are involved, nuclear model parameters differ from nuclide to nuclide. Then, the particular nuclide under consideration must also be given in the input line. In general, for these model parameters, we read keyword, Z, A, a physical value and sometimes a possible further index (e.g. the fission barrier, index for the giant resonance, etc.), all separated by blanks. As the example above shows for the level density parameter *a*, the same keyword can appear more than once in an input file, to cover several different nuclides. Again, remember that all such keywords, if you don't specify them, have a default value from either the nuclear structure and model parameter database or from systematics. The usual reason to change them is to fit experimental data, to use new information that is not yet in the TALYS database, or simply because the user may not agree with our default values. A final important point to note is that some keywords induce defaults for other keywords. This may seem confusing, but in practice this is not so. As an example, for a  $^{56}\text{Fe}$  target the **fission** keyword is automatically disabled whereas for  $^{232}\text{Th}$  it is by default enabled. Hence, the default value of the **fission** keyword is mass dependent. In the input description, you will find a few similar cases. Anyway, you can always find all adopted default values for all parameters at the top of the *output* file, see which values have been set by the user or by default, and overrule them in a new input file.

Since TALYS-1.96, it is also possible to give the Z,A relative to the compound nucleus. TALYS is not supposed to work for target nuclides up to mass 4, which enabled us to make use of the Z, A indices to give the relative compound nucleus Z, A equivalent to the real Z, A of the nucleus. Hence, the last two lines in the input file above can also be entered as

```
a 0 1 13.115
a 0 0 13.421
```

where the first keyword is for the target nucleus and the second for the compound nucleus. This

feature is especially helpful for automated TALYS scripts in which e.g. only a keyword for the compound nucleus is to be varied, since that one can always be specified by '0 0'.

### 3.2.1 Energy-dependent parameter adjustment

TALYS contains hundreds of parameters and most of them are energy-independent. They are ingredients of phenomenological models which have been constructed with, indeed, energy-independent parameters. The OMP parameter  $r_V$ , or the height of the fission barrier, **fisbar** are examples. There are two reasons to deviate from this. First, since these models are a phenomenological representation of reality, there is no strict law that these parameters should be energy-independent. Suppose for example that one tries to use a simple Hill-Wheeler expression to approximate fission paths coming from microscopic analyses, which are supposedly built on a better physical basis. Heights and widths of the fission barriers which are dependent on (in this case, excitation) energy will probably emerge. Then, one may want to use those in calculations.

A second reason is that if physically adequate models fail to fit experimental data, such energy-dependent adjustment can be invoked as a last resort. As long as the deviation from the original model is not too large, unpleasant surprises in the various reaction channels are avoided.

To accommodate this, no less than 65 TALYS keywords (parameters) now have the possibility to be extended with such an energy-dependence.

On the input line, we read the usual keyword and its energy-independent value. If a local variation is required, one adds (1) 4 extra parameters, or (2) a filename, to the input line. With either method a multiplication factor deviating from 1 can be given, as a function of energy. The 4 extra parameters,  $E_a$ ,  $E_b$ ,  $E_m$  and  $D$ , drive the following multiplication function

$$\begin{aligned} f(E) &= 1 + D' \frac{(E - E_a)^N}{(E - E_a)^N + (C - E_a)^N}, \text{ if } E < E_m \\ &= 1 + D' \frac{(E_b - E)^N}{(E_b - E)^N + (C - E_b)^N}, \text{ if } E \geq E_m \end{aligned} \quad (3.1)$$

where  $N$  is currently fixed to 4, and

$$D' = (D - 1) \left(1 + \frac{1}{2}\right)^N \quad (3.2)$$

and

$$\begin{aligned} C &= E_a + \frac{1}{2}(E_m - E_a), \text{ if } E < E_m \\ C &= E_m + \frac{1}{2}(E_b - E_m), \text{ if } E \geq E_m \end{aligned} \quad (3.3)$$

An input line would then look as follows:

```
keyword value(s) Ea Eb Em D
```

This means the following: the parameter has a constant value for all energies *outside* the begin energy  $E_a$  and end energy  $E_b$ . The multiplication function changes smoothly from  $E_a$  until  $E_m$ , and at  $E_m$  the deviation from the constant value has reached its maximum value  $D$ . Between  $E_m$  and  $E_b$  the function changes in the opposite direction until the constant value is reached again at  $E_b$ . To ensure some smoothness, Eq. (3.1) was borrowed from the KD03 OMP expression for  $W_V(E)$ . An example for the real volume radius of the optical model is

```
rvadjust n 1. 0.4 3. 1.7 1.02
```

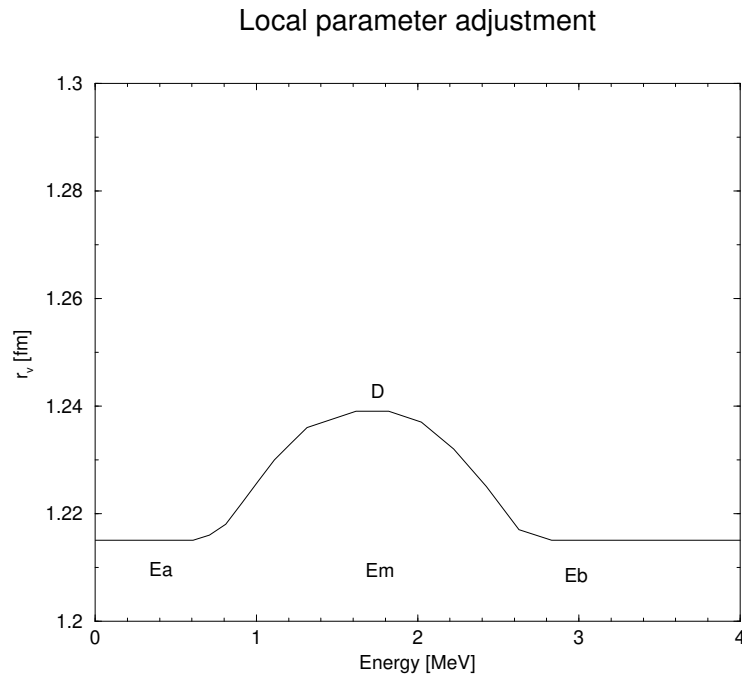


Figure 3.1: Local adjustment of the  $r_v$  parameter for  $^{93}\text{Nb}$ .

Suppose we use this input line for neutrons on  $^{93}\text{Nb}$ . The default value for  $r_v$  is 1.215 fm. We do not change this globally, hence the parameter value 1. Next, we change  $r_v$  between 0.4 and 3 MeV with the top (D is positive in this case) at 1.7 MeV. The result is presented in figure rv. One may use several segments, i.e. several input lines, with such local deviations in the same input file, as long as the energy ranges do not overlap (TALYS tests this).

The other option is to use a table with multiplication factors, e.g.

```
rvadjust n myfile.adj
```

where *myfile.adj* looks e.g. as follows

```
0.0  1.
0.4.  1.
1.7  1.1
3.0  1.
200  1.
```

which will simply produces a triangle with top at 1.7 MeV. For such simple cases, the use of Eq. (3.1) may be better.

The latter option may be interesting if one manages to fit, in some nuclear structure study, a phenomenological prescription available in TALYS to a more complex microscopic description of a nuclear process. For example, maybe a JLM OMP can be simulated by a Woods-Saxon OMP with energy dependent parameters, or a microscopic fission model by a Hill-Wheeler model with energy dependent fission parameters. The resulting fit can then be used as input file to TALYS.

The energy-dependent adjustment is currently possible for the following keywords.

- Optical model: **avadjust**, **avdadjust**, **avsoadjust**, **awadjust**, **awdadjust**, **awsoadjust**, **d1adjust**, **d2adjust**, **d3adjust**, **lv1adjust**, **lvadjust**, **lvsoadjust**, **lw1adjust**, **lwadjust**, **lwsoadjust**, **rcadjust**, **rvadjust**, **rvdadjust**, **rvsoadjust**, **rwadjust**, **rwdadjust**, **rwsadjust**, **tladjust**,

**v1adjust, v2adjust, v3adjust, v4adjust, vso1adjust, vso2adjust, w1adjust, w2adjust, w3adjust, w4adjust, wso1adjust, wso2adjust, adepthcor, aradialcor**

- Level density: **ctable, krotconstant, ptable, rspincut, rspincutff, s2adjust**
- Gamma-ray strength function: **sgr, sgradjust, spr, spradjust, egr, egradjust, wtable, wtableadjust, epr, epradjust, etable, ftable, ggr, ggradjust, gnorm, gpr, gpradjust**
- Pre-equilibrium: **cbreak, cknock, cstrip, m2constant**
- Fission: **fisbar, fisbaradjust, fishw, fishwadjust, bdamp, bdampadjust, betafiscor, betafiscoradjust, vfiscor, vfiscoradjust**

In a previous version of TALYS, this local adjustment was only applicable for the optical model parameters, with a different multiplication function, see e.g. **rvadjustF** on page 505. The latter keywords are now considered obsolete.

## 4. Nuclear reactions: General approach

An outline of the general theory and modeling of nuclear reactions can be given in many ways. A common classification is in terms of time scales: short reaction times are associated with direct reactions and long reaction times with compound nucleus processes. At intermediate time scales, pre-equilibrium processes occur. An alternative, more or less equivalent, classification can be given with the number of intranuclear collisions, which is one or two for direct reactions, a few for pre-equilibrium reactions and many for compound reactions, respectively. As a consequence, the coupling between the incident and outgoing channels decreases with the number of collisions and the statistical nature of the nuclear reaction theories increases with the number of collisions. Figs. 4.1 and 4.2 explain the role of the different reaction mechanisms during an arbitrary nucleon-induced reaction in a schematic manner. They will all be discussed in this manual.

This distinction between nuclear reaction mechanisms can be obtained in a more formal way by means of a proper division of the nuclear wave functions into open and closed configurations, as detailed for example by Feshbach's many contributions to the field. This is the subject of several textbooks and will not be repeated here. When appropriate, we will return to the most important theoretical aspects of the nuclear models in TALYS in Part II.

When discussing nuclear reactions in the context of a computer code, as in this manual, a different starting point is more appropriate. We think it is best illustrated by Fig. 4.3. A particle incident on a target nucleus will induce several *binary* reactions which are described by the various competing reaction mechanisms that were mentioned above. The end products of the binary reaction are the emitted particle and the corresponding recoiling residual nucleus. In general this is, however, not the end of the process. A total nuclear reaction may involve a whole sequence of residual nuclei, especially at higher energies, resulting from multiple particle emission. All these residual nuclides have their own separation energies, optical model parameters, level densities, fission barriers, gamma strength functions, etc., that must properly be taken into account along the reaction chain. The implementation of this entire reaction chain forms the backbone of TALYS. The

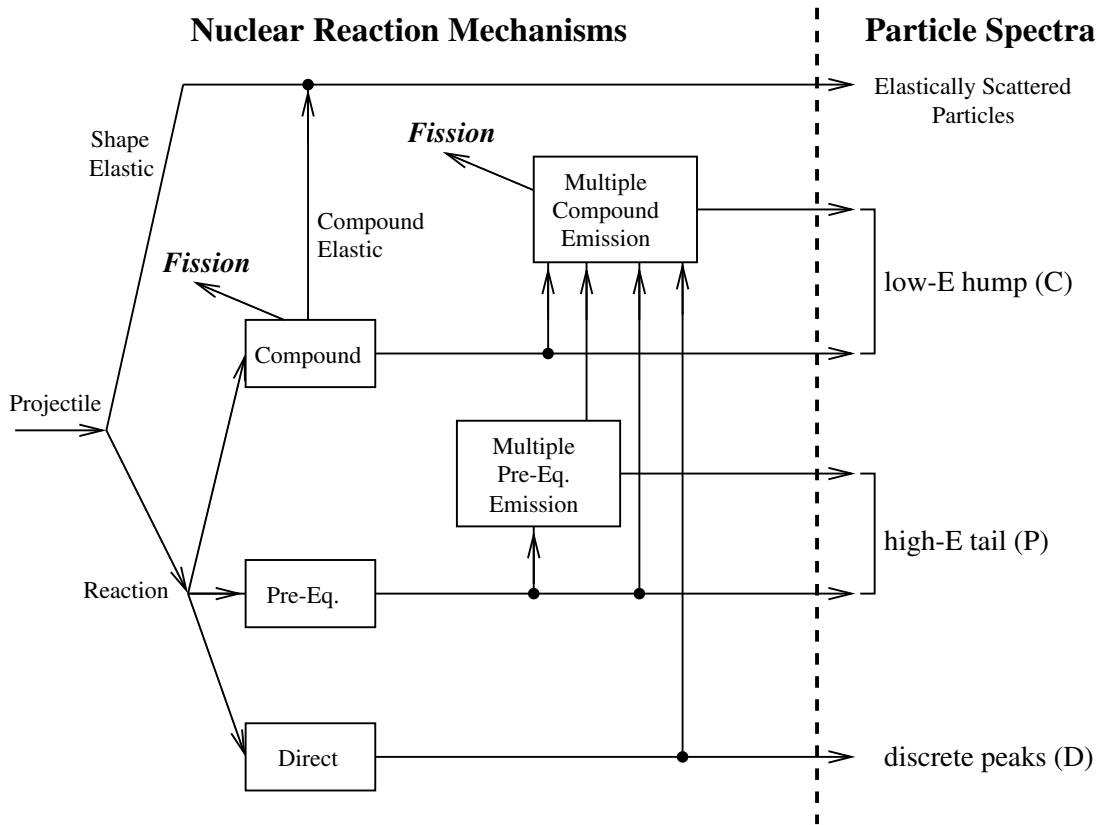


Figure 4.1: The role of direct, pre-equilibrium and compound processes in the description of a nuclear reaction and the outgoing particle spectra. The C, P and D labels correspond to those in Fig. 4.2

program has been written in a way that enables a clear and easy inclusion of all possible nuclear model ingredients for any number of nuclides in the reaction chain. Of course, in this whole chain the target and primary compound nucleus have a special status, since they are subject to *all* reaction mechanisms, i.e. direct, pre-equilibrium, compound and fission and, at low incident energies, width fluctuation corrections in compound nucleus decay. Also, at incident energies below a few MeV, only binary reactions take place and the target and compound nucleus are often the only two nuclei involved in the whole reaction. Historically, it is for the binary reactions that most of the theoretical methods have been developed and refined, mainly because their validity, and their relation with nuclear structure, could best be tested with exclusive measurements. In general, however, Fig. 4.3 should serve as the illustration of a total nuclear reaction at any incident energy. The projectile, in this case a neutron, and the target  $(Z_C, N_C - 1)$  form a compound nucleus  $(Z_C, N_C)$  with a total energy

$$E^{tot} = E_{CM} + S_n(Z_{CN}, N_{CN}) + E_x^0, \quad (4.1)$$

where  $E_{CM}$  is the incident energy in the CM frame,  $S_n$  is the neutron separation energy of the compound nucleus, and  $E_x^0$  the excitation energy of the target (which is usually zero, i.e. representing the ground state). The compound nucleus is described by a range of possible spin ( $J$ ) and parity ( $\Pi$ ) combinations, which for simplicity are left out of Fig. 4.3. From this state, transitions to all open channels may occur by means of direct, pre-equilibrium and compound processes. The



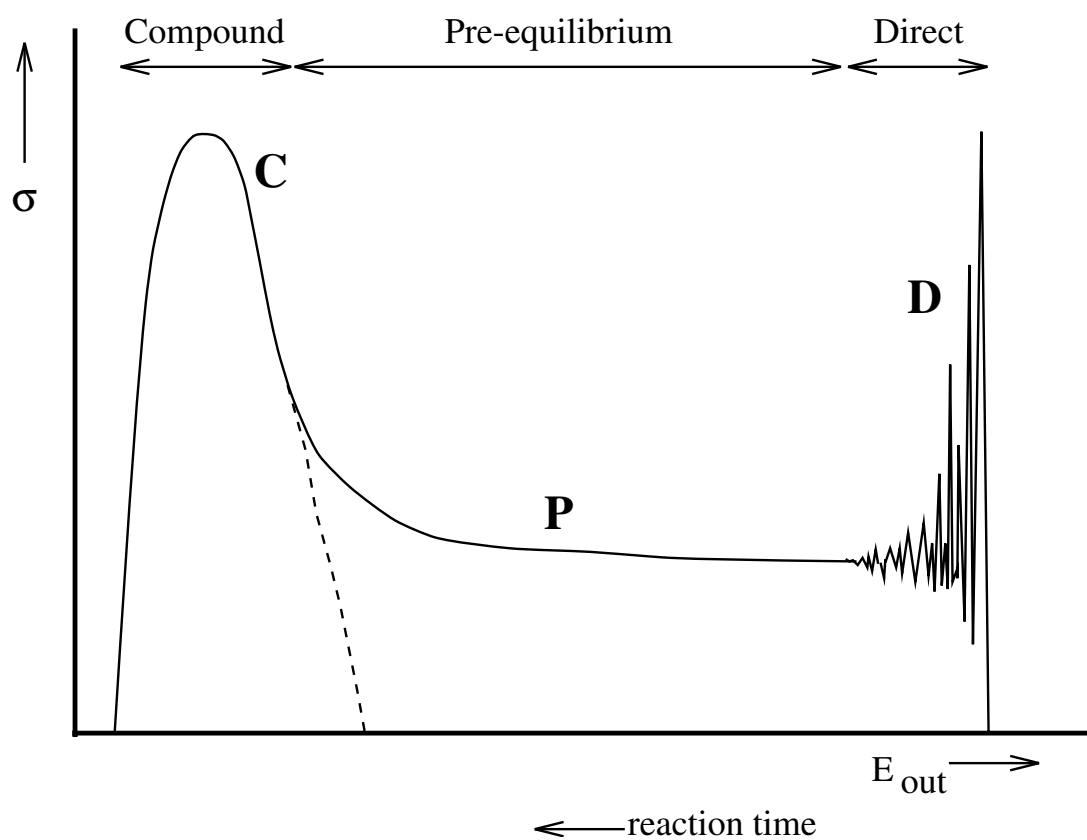


Figure 4.2: Schematic drawing of an outgoing particle spectrum. The energy regions to which direct (D), pre-equilibrium (P) and compound (C) mechanisms contribute are indicated. The dashed curve distinguishes the compound contribution from the rest in the transitional energy region.

residual nuclei formed by these binary reactions may be populated in the discrete level part and in the continuum part of the available excitation energy range. In Fig. 4.3, we have only drawn three binary channels, namely the  $(Z_C, N_C - 1)$ ,  $(Z_C - 1, N_C)$  and  $(Z_C - 1, N_C - 1)$  nuclei that result from binary neutron, proton and deuteron emission, respectively. Each nucleus is characterized by a separation energy per possible ejectile. If the populated residual nucleus has a maximal excitation energy  $E_x^{max}(Z, N)$  that is still above the separation energies for one or more different particles for that nucleus, further emission of these particles may occur and nuclei with lower  $Z$  and  $N$  will be populated. At the end of the nuclear reaction (left bottom part of Fig. 4.3), all the reaction population is below the lowest particle separation energy, and the residual nucleus  $(Z_C - z, N_C - n)$  can only decay to its ground or isomeric states by means of gamma decay. In a computer program, the continuum must be discretized in excitation energy ( $E_x$ ) bins. We can take these bins equidistant or non-equidistant (the default), although we already want to stress the important fact here that the *emission* energy grid for the outgoing particles is non-equidistant in TALYS. After the aforementioned binary reaction, every continuum excitation energy bin will be further depleted by means of particle emission, gamma decay or fission. Computationally, this process starts at the initial compound nucleus and its highest energy bin, i.e. the bin just below  $E_x^{max}(Z_C, N_C) = E^{tot}$ , and subsequently in order of decreasing energy bin/level, decreasing  $N$  and decreasing  $Z$ . Inside each continuum bin, there is an additional loop over all possible  $J$  and  $\Pi$ , whereas for each discrete level,  $J$  and  $\Pi$  have unique values. Hence, a bin/level is characterized by the set  $\{Z, N, E_x, J, \Pi\}$  and by means of gamma or particle emission, it can decay into all accessible  $\{Z', N', E_x', J', \Pi'\}$  bins/levels. In this way, the whole reaction chain is followed until all bins and levels are depleted and thus all channels are closed. In the process, all particle production cross sections and residual production cross sections are accumulated to their final values.

We will now zoom in on the various parts of Fig. 4.3 to describe the various stages of the reaction, depending on the incident energy, and we will mention the nuclear reaction mechanisms that apply.

## 4.1 Reaction mechanisms

In the projectile energy range between 1 keV and several hundreds of MeV, the importance of a particular nuclear reaction mechanism appears and disappears upon varying the incident energy. We will now describe the particle decay scheme that typically applies in the various energy regions. Because of the Coulomb barrier for charged particles, it will be clear that the discussion for low energy reactions usually concerns incident neutrons. In general, however, what follows can be generalized to incident charged particles. The energy ranges mentioned in each paragraph heading are just meant as helpful indications, which apply for a typical medium mass nucleus.

### 4.1.1 Low energies

#### Elastic scattering and capture ( $E < 0.2$ MeV)

If the energy of an incident neutron is below the excitation energy of the first inelastic level, and if there are no  $(n, p)$ , etc. reactions that are energetically possible, then the only reaction possibilities are elastic scattering, neutron capture and, for fissile nuclides, fission. At these low energies, only the  $(Z_C, N_C - 1)$  and  $(Z_C, N_C)$  nuclides of Fig. 4.3 are involved, see Fig. 4.4. First, the shape (or direct) elastic scattering cross section can directly be determined from the optical model, which will be discussed in Section 10. The compound nucleus, which is populated by a reaction population equal to the reaction cross section, is formed at one single energy  $E^{tot} = E_x^{max}(Z_C, N_C)$  and a range

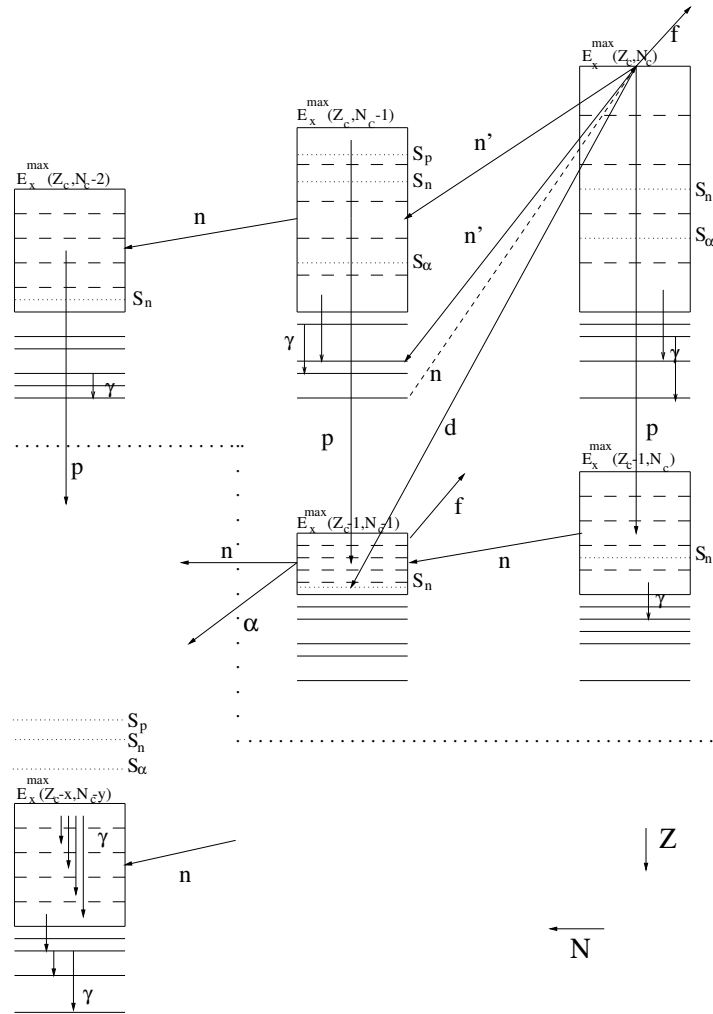


Figure 4.3: Neutron-induced reaction. The dashed arrow represents the incident channel, while the continuous arrows represent the decay possibilities.  $E_x^{\max}$  denotes the maximal possible excitation energy of each nucleus and  $S_k$  is the particle separation energy for particle  $k$ . For each nucleus a few discrete levels are drawn, together with a few continuum energy bins. Spin and parity degrees of freedom are left out of this figure for simplicity. Fission is indicated by an  $f$ .

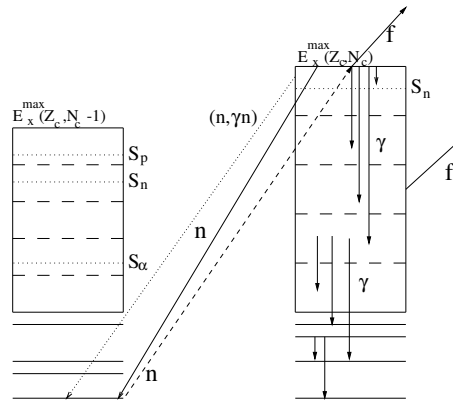


Figure 4.4: Neutron-induced reaction at low energy. The dashed arrow represents the incident channel, while the continuous arrows represents the elastic channel. The only possibilities are elastic scattering and capture of the neutron in the compound nucleus, with subsequent decay to the ground state or an isomeric state of the compound nucleus. A small part of the population may decay to the target nucleus by means of the  $(n, \gamma n)$  channel (dotted arrow). For fissile nuclei, fission may be another open channel.

of  $J, \Pi$ -values. This compound nucleus either decays by means of compound elastic scattering back to the initial state of the target nucleus, or by means of neutron capture, after which gamma decay follows to the continuum and to discrete states of the compound nucleus. The competition between the compound elastic and capture channels is described by the compound nucleus theory, which we will discuss in Section 13. To be precise, the elastic and capture processes comprise the first *binary* reaction. To complete the description of the total reaction, the excited  $(Z_C, N_C)$  nucleus, which is populated over its whole excitation energy range by the primary gamma emission, must complete its decay. The highest continuum bin is depleted first, for all  $J$  and  $\Pi$ . The subsequent gamma decay increases the population of the lower bins, before the latter are depleted themselves. Also, continuum bins that are above the neutron separation energy  $S_n$  of the compound nucleus contribute to the feeding of the  $(n, \gamma n)$  channel. This results in a weak continuous neutron spectrum, even though the elastic channel is the only true binary neutron channel that is open. The continuum bins and the discrete levels of the compound nucleus are depleted one by one, in decreasing order, until the ground or an isomeric state of the compound nucleus is reached by subsequent gamma decay. If a nuclide is fissile, fission may compete as well, both from the initial compound state  $E_x^{max}(Z_C, N_C)$  and from the continuum bins of the compound nucleus, the latter resulting in a  $(n, \gamma f)$  cross section. Both contributions add up to the so called first-chance fission cross section.

#### Inelastic scattering to discrete states ( $0.2 < E < 4$ MeV)

At somewhat higher incident energies, the first inelastic channels open up, see Fig. 4.5. Reactions to these discrete levels have a compound and a direct component. The former is again described by the compound nucleus theory, while the latter is described by the Distorted Wave Born Approximation (DWBA) for spherical nuclei and by coupled-channels equations for deformed nuclei, see Section 11. When the incident energy crosses an inelastic threshold, the compound inelastic contribution rises rapidly and predominates, whereas the direct component increases more gradually. Obviously, the elastic scattering, capture and fission processes described in the previous subsection also apply here. In addition, there is now gamma decay to an isomeric state or the ground state in the target nucleus after inelastic scattering. When there are several, say 10, inelastic levels open to decay,

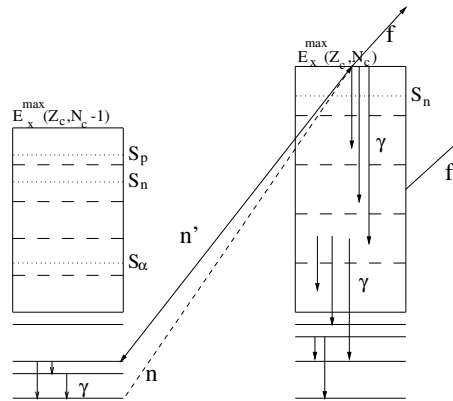


Figure 4.5: Neutron-induced reaction at somewhat higher energy. The dashed arrow represents the incident channel, while the continuous arrows represent the decay possibilities. In addition to the possibilities sketched in the previous figure, there is now inelastic scattering followed by gamma decay in the target nucleus.

the compound contribution to each individual level is still significant. However, the effect of the width fluctuation correction on the compound cross section is already small in this case, as will be outlined in Section 13.

### 4.1.2 High energies

#### Pre-equilibrium reactions ( $E > 4$ MeV)

At higher incident energies, inelastic cross sections to both the discrete states and the continuum are possible, see Fig. 4.3. Like reactions to discrete states, reactions to the continuum also have a compound and a direct-like component. The latter are usually described by pre-equilibrium reactions which, by definition, include direct reactions to the continuum. They will be discussed in Section 12. Also non-elastic channels to other nuclides, through charge-exchange, e.g.  $(n, p)$ , and transfer reactions, e.g.  $(n, \alpha)$ , generally open up at these energies, and decay to these nuclides can take place by the same direct, pre-equilibrium and compound mechanisms. Again, the channels described in the previous subsections also apply here. In addition, gamma decay to ground and isomeric states of all residual nuclides occurs. When many channels open up, particle decay to individual states (e.g. compound elastic scattering) rapidly becomes negligible. For the excitation of a discrete state, the direct component now becomes predominant, since that involves no statistical competition with the other channels. At about 15 MeV, the *total* compound cross section, i.e. summed over all final discrete states and the excited continuum, is however still larger than the summed direct and pre-equilibrium contributions.

#### Multiple compound emission ( $E > 8$ MeV)

At incident energies above about the neutron separation energy, the residual nuclides formed after the first binary reaction contain enough excitation energy to enable further decay by compound nucleus particle emission or fission. This gives rise to multiple reaction channels such as  $(n, 2n)$ ,  $(n, np)$ , etc. For higher energies, this picture can be generalized to many residual nuclei, and thus more complex reaction channels, as explained in the introduction of this Chapter, see also Fig. 4.3. If fission is possible, this may occur for all residual nuclides, which is known as multiple chance fission. All excited nuclides will eventually decay to their isomeric and ground states.

**Multiple pre-equilibrium emission ( $E > 40$  MeV)**

At still higher incident energies, above several tens of MeV, the residual nuclides formed after binary emission may contain so much excitation energy that the presence of further *fast* particles inside the nucleus becomes possible. These can be imagined as strongly excited particle-hole pairs resulting from the first binary interaction with the projectile. The residual system is then clearly non-equilibrated and the excited particle that is high in the continuum may, in addition to the first emitted particle, also be emitted on a short time scale. This so-called multiple pre-equilibrium emission forms an alternative theoretical picture of the intra-nuclear cascade process, whereby now not the exact location and momentum of the particles is followed, but instead the total energy of the system and the number of particle-hole excitations (exciton number). In TALYS, this process can be generalized to any number of multiple pre-equilibrium stages in the reaction by keeping track of all successive particle-hole excitations, see Section 14.2. For these incident energies, the binary compound cross section becomes small: the non-elastic cross section is almost completely exhausted by primary pre-equilibrium emission. Again, Fig. 4.3 applies.

**Sample case 4.1 All results for 14 MeV n +  $^{93}\text{Nb}$** 

We have included 9 different versions of this sample case, in order to give an impression of the various types of information that can be retrieved from TALYS. Most, but not all, output options will be described, while the remainder will appear in the other sample cases. We have chopped the sample output after column 80 to let it fit within this manual. We suggest to consult the output files in the *samples/* directory for the full results.

**Case a: The simplest input file**

The first sample problem concerns the simplest possible TALYS calculation. Consider the following input file that produces the results for a 14 MeV neutron on  $^{93}\text{Nb}$ :

```
#
# n-Nb093-14MeV
#
# General
#
projectile n
element nb
mass 93
energy 14.
```

where the purpose of the lines starting with a “#” is purely cosmetic. This input file called *input* can simply be run as follows:

```
talys < talys.inp > talys.out
```

An output file of TALYS consists of several blocks. Whether these blocks are printed or not depends on the status of the various keywords that were discussed in Chapter 3. By default, the so-called main output is always given (through the default **outmain y**), and we discuss this output in the present sample case. For a single incident energy, a default calculation gives the most important cross sections only. “Most important” is obviously subjective, and probably

every user has an own opinion on what should always appear by default in the output. We will demonstrate in the other sample problems how to extract all information from TALYS. The output file starts with a display of the version of TALYS you are using, the name of the authors, and the Copyright statement. Also the physics dimensions used in the output are given:

```
TALYS-1.96      (Version: December 30, 2021)
```

```
Copyright (C) 2021 A.J. Koning, S. Hilaire and S. Goriely
```

```
Dimensions - Cross sections: mb, Energies: MeV, Angles: degrees
```

The next output block begins with:

```
##### USER INPUT #####
```

Here, the first section of the output is a print of the keywords/input parameters. This is done in two steps: First, in the block

```
USER INPUT FILE
```

```
#
# General
#
projectile n
element nb
mass 93
energy 14.
```

an exact copy of the input file as given by the user is returned. Next, in the block

```
USER INPUT FILE + DEFAULTS
```

Keyword	Value	Variable	Explanation
#			
#	# Four main keywords		
#			
projectile	n	ptype0	type of incident particle
element	Nb	Starget	symbol of target nucleus
mass	93	mass	mass number of target nucleus
energy	14.000	eninc	incident energy in MeV
#			
#	# Basic physical and numerical parameters		
#			
ejectiles	g n p d t h a	outtype	outgoing particles
.....			

a table with all keywords is given, not only the ones that you have specified in the input file, but also all the defaults that are set automatically. The corresponding Fortran variables are also printed, together with a short explanation of their meaning. This table can be helpful as a guide

to change further input parameters for a next run. You may also copy and paste the block directly into your next input file.

In the next output block

```
##### BASIC REACTION PARAMETERS #####

Projectile      : neutron      Mass in a.m.u.   :   1.008665
Target          :  93Nb        Mass in a.m.u.   :  92.906378

Included channels:
                gamma
                neutron
                proton
                deuteron
                triton
                helium-3
                alpha

1 incident energy (LAB):

    14.000

Q-values for binary reactions:

Q(n,g):  7.22755
Q(n,n):  0.00000
Q(n,p):  0.69112
Q(n,d): -3.81879
Q(n,t): -6.19635
Q(n,h): -7.72313
Q(n,a):  4.92559
```

we print the main parameters that characterize the nuclear reaction: the projectile, target and their masses and the outgoing particles that are included as competitive channels. The incident energy or range of incident energies in the LAB system is given together with the binary reaction Q-values.

The block with final results starts with

```
##### RESULTS FOR E= 14.00000 #####

Energy dependent input flags

Width fluctuations (flagwidth)      : n
Preequilibrium (flagpreeq)         : y
Multiple preequilibrium (flagmulpre) : n
Number of continuum excitation energy bins: 40
```

with no further information for the present sample case since no further output was requested. When all nuclear model calculations are done, the most important cross sections are summarized in the main part of the output, in which we have printed the center-of-mass energy, the main



(total) cross sections. the inclusive binary cross sections  $\sigma_{n,k}^{inc,bin}$ , see Eq. (4.19), the total particle production cross sections  $\sigma_{n,xn}$  of Eq. (4.20) and the multiplicities  $Y_n$  of Eq. (4.22), and the residual production cross sections. The latter are given first per produced nuclide and isomer. Next, nuclides with the same mass are summed to give mass yield curves. Also, the sum over all the residual cross sections is compared with the non-elastic cross section. Obviously, these two values should be approximately equal.

```
##### REACTION SUMMARY FOR E= 14.000 #####
```

```
Center-of-mass energy: 13.849
```

#### 1. Total (binary) cross sections

```
Total          = 3.98195E+03
Shape elastic   = 2.21132E+03
Reaction        = 1.77063E+03
Compound elastic= 6.00478E-04
Non-elastic     = 1.77063E+03
  Direct        = 3.13938E+01
  Pre-equilibrium = 4.15372E+02
  Giant resonance = 6.25327E+01
Compound non-el = 1.26133E+03
Total elastic   = 2.21132E+03
```

#### 2. Binary non-elastic cross sections (non-exclusive)

```
gamma    = 4.18076E+00
neutron  = 1.69541E+03
proton   = 3.82455E+01
deuteron = 4.95187E+00
triton   = 1.81580E-01
helium-3 = 7.05611E-10
alpha    = 2.76590E+01
```

#### 3. Total particle production cross sections

```
gamma    = 2.20703E+03      Multiplicity= 1.24647E+00
neutron  = 3.08077E+03      Multiplicity= 1.73993E+00
proton   = 4.06703E+01      Multiplicity= 2.29694E-02
deuteron = 4.95187E+00      Multiplicity= 2.79667E-03
triton   = 1.81580E-01      Multiplicity= 1.02551E-04
helium-3 = 7.05612E-10      Multiplicity= 3.98509E-13
alpha    = 2.83527E+01      Multiplicity= 1.60128E-02
```

#### 4. Residual production cross sections

##### a. Per isotope

Z	A	nuclide	total cross section	level cross section	isomeric cross section	isomeric ratio	lifetime
---	---	---------	------------------------	------------------------	---------------------------	-------------------	----------

```

41  94 ( 94Nb) 1.18322E+00    0    5.99984E-01  0.50708
                                     1    5.83234E-01  0.49292  3.76000E+02 sec.
41  93 ( 93Nb) 3.23646E+02    0    2.77623E+02  0.85780
                                     1    4.60226E+01  0.14220  5.09000E+08 sec.
40  93 ( 93Zr) 2.98798E+01    0    2.98798E+01  1.00000
41  92 ( 92Nb) 1.37164E+03    0    8.51757E+02  0.62098
                                     1    5.19887E+02  0.37902  8.77000E+05 sec.
40  92 ( 92Zr) 1.57424E+01    0    1.57424E+01  1.00000
40  91 ( 91Zr) 1.81576E-01    0    1.81576E-01  1.00000
39  90 ( 90Y ) 2.53059E+01    0    1.28873E+01  0.50926
                                     2    1.24186E+01  0.49074  1.15000E+04 sec.
39  89 ( 89Y ) 3.04682E+00    0    1.23419E+00  0.40508
                                     1    1.81263E+00  0.59492  1.57000E+01 sec.

```

b. Per mass

A cross section

```

94 1.18322E+00
93 3.53525E+02
92 1.38739E+03
91 1.81576E-01
90 2.53059E+01
89 3.04682E+00

```

Total residual production cross section: 1770.62915

Non-elastic cross section : 1770.62939

At the end of the output, the total calculation time is printed, followed by a message that the calculation has been successfully completed:

Execution time: 0 hours 0 minutes 3.88 seconds

The TALYS team congratulates you with this successful calculation.

### Case b: Discrete state cross sections and spectra

As a first extension to the simple input/output file given above, we will request the output of cross sections per individual discrete level. Also, the cumulated angle-integrated and double-differential particle spectra are requested. This is obtained with the following input file:

```

#
# General
#
projectile n
element nb
mass 93
energy 14.
#
# Output
#

```

```

outdiscrete y
outspectra y
ddxmode 2
filespectrum n p a
fileddxa n 30.
fileddxa n 60.
fileddxa a 30.
fileddxa a 60.
    
```

In addition to the information printed for case 1a, the cross sections per discrete state for each binary channel are given, starting with the  $(n, \gamma)$  channel,

5. Binary reactions to discrete levels and continuum

(n,g) cross sections:

Inclusive:

Level	Energy	E-out	J/P	Direct	Compound	Total	Origin
0	0.00000	21.07609	6.0+	0.00000	0.00000	0.00000	Preeq
1	0.04090	21.03519	3.0+	0.00000	0.00000	0.00000	Preeq
.....							

after which the inelastic cross section to every individual discrete state of the target nucleus is printed, including the separation in direct and compound, see Eq. (4.16). These are summed, per contribution, to the total discrete inelastic cross section, see Eq. (4.15). To these cross sections, the continuum inelastic cross sections of Eq. (4.17) are added to give the total inelastic cross section (4.14). Finally, the  $(n, \gamma n)$  cross section is also printed. This output block looks as follows

Inelastic cross sections:

Inclusive:

Level	Energy	E-out	J/P	Direct	Compound	Total	Origin
1	0.03077	13.81777	0.5-	0.00670	0.00013	0.00683	Direct
2	0.68709	13.16145	1.5-	0.01311	0.00025	0.01336	Direct
3	0.74386	13.10468	3.5+	4.28791	0.00048	4.28839	Direct
4	0.80849	13.04005	2.5+	0.08705	0.00037	0.08741	Direct
5	0.81025	13.03829	2.5-	3.20756	0.00036	3.20792	Direct
6	0.94982	12.89872	6.5+	7.43461	0.00065	7.43526	Direct
7	0.97000	12.87854	1.5-	0.02590	0.00024	0.02614	Direct
8	0.97891	12.86963	5.5+	6.36403	0.00062	6.36465	Direct
9	1.08257	12.76597	4.5+	5.27768	0.00056	5.27823	Direct
10	1.12676	12.72178	2.5+	0.31992	0.00036	0.32028	Direct
11	1.28440	12.56414	0.5+	0.01591	0.00012	0.01603	Direct
12	1.29000	12.55854	1.5-	0.24336	0.00024	0.24360	Direct
13	1.29715	12.55139	4.5+	0.26629	0.00055	0.26684	Direct
14	1.31515	12.53339	2.5-	0.02135	0.00035	0.02170	Direct

15	1.33000	12.51854	1.5+	0.03001	0.00024	0.03025	Direct
16	1.33516	12.51338	8.5+	0.96835	0.00060	0.96895	Direct
17	1.36400	12.48454	3.5-	0.42107	0.00044	0.42152	Direct
18	1.36970	12.47884	1.5+	0.03173	0.00024	0.03197	Direct
19	1.39510	12.45344	2.5+	0.08954	0.00035	0.08990	Direct
20	1.45420	12.39434	0.5+	0.10706	0.00012	0.10718	Direct
				-----	-----	-----	
Discrete	Inelastic:			29.21914	0.00727	29.22641	
Continuum	Inelastic:			428.71729	1231.55090	1660.26819	
				-----	-----	-----	
Total	Inelastic:			457.93640	1231.55823	1689.49597	
(n,gn) cross section:				1.05107			

This is repeated for the  $(n, p)$  and other channels,

(n,p) cross sections:

Inclusive:

Level	Energy	E-out	J/P	Direct	Compound	Total	Origin
0	0.00000	14.53966	2.5+	0.33165	0.00026	0.33191	Preeq
1	0.26688	14.27278	1.5+	1.10724	0.00017	1.10741	Preeq
2	0.94714	13.59252	0.5+	0.85027	0.00008	0.85035	Preeq
3	1.01800	13.52166	0.5+	0.24244	0.00008	0.24252	Preeq
.....							

The last column in these tables specifies the origin of the direct contribution to the discrete state. "Direct" means that this is obtained with coupled-channels, DWBA or, as in this case, weak coupling, whereas "Preeq" means that the pre-equilibrium cross section is collapsed onto the discrete states, as an approximate method for more exact direct reaction approaches for charge-exchange and pick-up reactions. We note here that the feature of calculating, and printing, the inelastic cross sections for a specific state is of particular interest in the case of excitations, i.e. to obtain this particular cross section for a whole range of incident energies. This will be handled in another sample case.

Since **outspectra y** was specified in the input file, the composite particle spectra for the continuum are also printed. Besides the total spectrum, the division into direct (i.e. smoothed collective effects and giant resonance contributions), pre-equilibrium, multiple pre-equilibrium and compound is given. First we give the photon spectrum,

#### 7. Composite particle spectra

Spectra for outgoing gamma

Energy	Total	Direct	Pre-equil.	Mult. preeq	Compound
0.001	1.10530E+01	0.00000E+00	2.22494E-13	0.00000E+00	1.10530E+01
0.002	1.11348E+01	0.00000E+00	2.17671E-12	0.00000E+00	1.11348E+01
0.005	1.13792E+01	0.00000E+00	2.99265E-11	0.00000E+00	1.13792E+01

```

0.010 1.17873E+01 0.00000E+00 2.25661E-10 0.00000E+00 1.17873E+01
0.020 1.26069E+01 0.00000E+00 1.71084E-09 0.00000E+00 1.26069E+01
0.050 1.54725E+01 0.00000E+00 2.42110E-08 0.00000E+00 1.54725E+01
0.100 3.27488E+01 0.00000E+00 1.71082E-07 0.00000E+00 3.27488E+01
.....

```

followed by the neutron spectrum

Spectra for outgoing neutron

Energy	Total	Direct	Pre-equil.	Mult. preeq	Compound
0.001	1.77195E+01	7.72138E-03	9.45985E-02	0.00000E+00	1.76172E+01
0.002	3.53846E+01	7.73366E-03	1.42810E-01	0.00000E+00	3.52340E+01
0.005	8.83650E+01	7.77059E-03	2.69012E-01	0.00000E+00	8.80882E+01
0.010	1.45726E+02	7.83249E-03	4.82712E-01	0.00000E+00	1.45236E+02
0.020	2.71007E+02	7.95765E-03	9.71034E-01	0.00000E+00	2.70028E+02
0.050	5.46087E+02	8.34413E-03	2.85497E+00	0.00000E+00	5.43223E+02
0.100	8.71934E+02	9.02631E-03	6.72888E+00	0.00000E+00	8.65196E+02

.....

and the spectra for the other outgoing particles. Depending on the value of **ddxmode**, the double-differential cross sections are printed as angular distributions or as spectra per fixed angle. For the present sample case, **ddxmode 2**, which gives

9. Double-differential cross sections per outgoing angle

DDX for outgoing neutron at 0.000 degrees

E-out	Total	Direct	Pre-equil.	Mult. preeq	Compound
0.001	1.41397E+00	1.50567E-03	1.05321E-02	0.00000E+00	1.40193E+00
0.002	2.82124E+00	1.50806E-03	1.59003E-02	0.00000E+00	2.80383E+00
0.005	7.04130E+00	1.51527E-03	2.99548E-02	0.00000E+00	7.00983E+00
0.010	1.16128E+01	1.52734E-03	5.37603E-02	0.00000E+00	1.15575E+01
0.020	2.15979E+01	1.55174E-03	1.08185E-01	0.00000E+00	2.14882E+01
0.050	4.35484E+01	1.62711E-03	3.18426E-01	0.00000E+00	4.32283E+01
0.100	6.96037E+01	1.76013E-03	7.51868E-01	0.00000E+00	6.88501E+01

.....

followed by the other angles and other particles. A final important feature of the present input file is that some requested information has been written to separate output files, i.e. besides the standard output file, TALYS also produces the ready-to-plot files

```

aspec014.000.tot
nspec014.000.tot
pspec014.000.tot

```

containing the angle-integrated neutron, proton and alpha spectra, and

```

addx030.0.deg

```

```

addx060.0.deg
nddx030.0.deg
nddx060.0.deg

```

containing the double-differential neutron and alpha spectra at 30 and 60 degrees.

### Case c: Exclusive channels and spectra

As another extension of the simple input file we can print the exclusive cross sections at one incident energy and the associated exclusive spectra. This is accomplished with the input file

```

#
# General
#
projectile n
element nb
mass 93
energy 14.
#
# Output
#
channels y
outspectra y

```

Contrary to the previous sample case, in this case no double-differential cross sections or results per separate file are printed (since it only concerns one incident energy). The exclusive cross sections are given in one table, per channel and per ground or isomeric state. It is checked whether the exclusive cross sections add up to the non-elastic cross section. Note that this sum rule, Eq. (4.25), is only expected to hold if we include enough exclusive channels in the calculation. If **maxchannel 4**, this equality should always hold for incident energies up to 20 MeV. This output block looks as follows:

#### 6. Exclusive cross sections

##### 6a. Total exclusive cross sections

Emitted particles						cross section	reaction	level	isomeric cross section
n	p	d	t	h	a				
0	0	0	0	0	0	1.18344E+00	(n,g)	0	5.75062E-01
								1	6.08379E-01
1	0	0	0	0	0	3.26664E+02	(n,n')	0	2.74409E+02
								1	5.22550E+01
0	1	0	0	0	0	2.96168E+01	(n,p)		
0	0	1	0	0	0	1.04112E+01	(n,d)		
0	0	0	1	0	0	6.70822E-01	(n,t)		
0	0	0	0	0	1	2.73777E+01	(n,a)	0	1.35359E+01
								2	1.38418E+01
2	0	0	0	0	0	1.35982E+03	(n,2n)	0	8.46000E+02
								1	5.13817E+02
1	1	0	0	0	0	1.06861E+01	(n,np)		
1	0	0	0	0	1	3.14862E+00	(n,na)		

```

0 1.27412E+00
1 1.87450E+00
0 1 0 0 0 1 1.49401E-05 (n,pa)
Sum over exclusive channel cross sections: 1769.57593
(n,gn) + (n,gp) +...(n,ga) cross sections: 1.05321
Total : 1770.62915
Non-elastic cross section : 1770.62939

```

Note that the  $(n,np)$  and  $(n,d)$  cross sections add up to the residual production cross section for  $^{92}\text{Zr}$ , as given in the first sample case.

Since **outspectra y**, for each exclusive channel the spectrum per outgoing particle is given. This output block begins with:

6b. Exclusive spectra

```

      Emitted particles      cross section reaction      gamma cross section
      n  p  d  t  h  a
      1  0  0  0  0  0  3.26664E+02 (n,n') 1.00870E+03

```

Outgoing spectra

```

Energy  gamma      neutron      proton      deuteron      triton      helium-3
0.001  4.06619E+00  2.14250E-03  0.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
0.002  4.08311E+00  4.22497E-03  0.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
0.005  4.13354E+00  1.04500E-02  0.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
0.010  4.21777E+00  2.08288E-02  0.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
0.020  4.38694E+00  4.16553E-02  0.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
0.050  4.92328E+00  1.04608E-01  0.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
0.100  6.46225E+00  2.10358E-01  0.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
.....

```

```

      Emitted particles      cross section reaction      gamma cross section
      n  p  d  t  h  a
      1  1  0  0  0  0  1.06861E+01 (n,np) 1.97965E+01

```

Outgoing spectra

```

Energy  gamma      neutron      proton      deuteron      triton      helium-3
0.001  6.35050E-02  1.87485E-01  0.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
0.002  6.94942E-02  3.74836E-01  0.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
0.005  8.74445E-02  9.36836E-01  0.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
0.010  1.17368E-01  1.88679E+00  0.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
0.020  1.77238E-01  3.89497E+00  0.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
0.050  3.57037E-01  6.03595E+00  0.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
0.100  6.57333E-01  8.96107E+00  0.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
0.200  1.58990E+00  1.04045E+01  0.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
0.300  1.88712E+00  9.88185E+00  0.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
0.400  2.01034E+00  9.03657E+00  0.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
0.500  1.41701E+00  8.28305E+00  0.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00

```

```

0.600 5.68786E+00 7.59201E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
0.700 8.64822E+00 6.60600E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
0.800 4.28359E+00 5.86945E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
0.900 2.37448E+00 5.09628E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
1.000 2.20641E+00 4.19745E+00 1.23263E-07 0.00000E+00 0.00000E+00 0.00000E+00
.....

```

Note, as explained in Section 4.3.2, that the (n,np) channel is characterized by both a neutron and a proton spectrum.

#### Case d: Nuclear structure

It is possible to have all the nuclear structure information in the output file. The simplest way is to set **outbasic y**, which means that about everything that can be printed, will be printed. This may be a bit overdone if one is only interested in e.g. discrete levels or level densities. If the keywords **outlevels** and/or **outdensity** are set to **y**, discrete level and level density information will always be given for the target nucleus and the primary compound nucleus. With **outgamma y**, photon strength function information is also given. If we would set, in addition, **outpopulation y**, this info will also be given for all the other residual nuclides that are reached in the reaction chain. The input file for this sample case is

```

#
# General
#
projectile n
element nb
mass 93
energy 14.
#
# Output
#
outlevels y
outdensity y
outgamma y

```

In addition to the output of case 1a, the separation energies for the six light particles are printed.

```

NUCLEAR STRUCTURE INFORMATION FOR Z= 41 N= 52 ( 93Nb)
Separation energies:
Particle      S
neutron       8.83126
proton        6.04337
deuteron     12.45360
triton       13.39081
helium-3     15.65202
alpha        1.93144

```

In the next output block, the discrete level scheme is printed for the first **maxlevelstar** levels. The discrete level info contains level number, energy, spin, parity, branching ratios and lifetimes of possible isomers. It is also indicated whether the spin (J) or parity (P) of a level is experimentally known or whether a value was assigned to it (see Section 7.1). The “string” of the original



ENSDF database is also given, so that the user can learn about possible alternative choices for spin and parity. This output block begins with:

```
Discrete levels of Z= 41 N= 52 ( 93Nb)

Number  Energy Spin Parity  Branching  Ratio (%) Lifetime(sec) Assignment

   0     0.0000  4.5  +
   1     0.0308  0.5  -                    5.090E+08
        --->  0  100.0000
   2     0.6871  1.5  -                    --->  1  100.0000
   3     0.7439  3.5  +                    --->  0  100.0000
   4     0.8085  2.5  +                    --->  3   2.1751
        --->  0  97.8249
.....
```

Since **outdensity y**, we print all the level density parameters that are involved, as discussed in Section 8: the level density parameter at the neutron separation energy  $a(S_n)$ , the experimental and theoretical average resonance spacing  $D_0$ , the asymptotic level density parameter  $\tilde{a}$ , the shell damping parameter  $\gamma$ , the pairing energy  $\Delta$ , the shell correction energy  $\delta W$ , the matching energy  $E_x$ , the last discrete level, the levels for the matching problem, the temperature  $T$ , the back-shift energy  $E_0$ , the discrete state spin cut-off parameter  $\sigma$  and the spin cut-off parameter at the neutron separation energy. Next, we print a table with the level density parameter  $a$ , the spin cut-off parameter and the level density itself, all as a function of the excitation energy. This output block begins with:

```
Level density parameters for Z= 41 N= 52 ( 93Nb)

Model: Gilbert-Cameron
Collective enhancement: no

a(Sn)          : 12.33543
Experimental D0 :          0.00 eV +-          0.00000
Theoretical D0  :          77.69 eV
Asymptotic a    : 12.24515
Damping gamma   :  0.09559
Pairing energy   :  1.24434
Shell correction:  0.10845
Last disc. level:          20
Nlow            :          5
Ntop           :          15
Matching Ex     :  7.63849
Temperature     :  0.86816
EO             : -1.36134
Adj. pair shift :  0.00000
Discrete sigma  :  2.78158
Sigma (Sn)     :  4.57840
```

```

Level density per parity for ground state
(Total level density summed over parity)
  Ex      a      sigma      total      JP= 0.5  JP= 1.5  JP= 2.5  JP= 3.5  JP= 4.5

0.25 12.372 2.782 3.685E+00 4.465E-01 7.356E-01 7.987E-01 6.774E-01 4.734E-0
0.50 12.372 2.782 4.915E+00 5.955E-01 9.810E-01 1.065E+00 9.035E-01 6.313E-0
0.75 12.372 2.782 6.555E+00 7.942E-01 1.308E+00 1.421E+00 1.205E+00 8.420E-0
1.00 12.372 2.782 8.742E+00 1.059E+00 1.745E+00 1.895E+00 1.607E+00 1.123E+0
.....

```

Finally, a comparison of the total level density with the cumulative number of discrete levels is given,

Discrete levels versus total level density

```

Energy Level  N_cumulative

0.9498 6      7.125
0.9700 7      7.462
0.9789 8      7.614
1.0826 9      9.491
1.1268 10     10.363
1.2844 11     13.856
1.2900 12     13.992
1.2972 13     14.167
.....

```

Note also that, since **outdensity y** by default implies **filedensity y**, the files *ld041093.tot* and *ld041094.tot* have been created. They contain all level density parameters and a comparison between cumulated discrete levels and the integrated level density.

With **outgamma y** the gamma-ray information is printed. First, all relevant parameters are given: the total radiative width  $\Gamma_\Gamma$ , the s-wave resonance spacing  $D_0$ , the s-wave strength function  $S_0$  and the normalization factor for the gamma-ray strength function. Second, we print the giant resonance information. For each multipolarity, we print the strength of the giant resonance  $\sigma_0$ , its energy and its width. Next, the gamma-ray strength function and transmission coefficients for this multipolarity and as a function of energy are printed. This output block begins with:

```

##### GAMMA STRENGTH FUNCTIONS, TRANSMISSION COEFFICIENTS AND CROSS SECTIO

Gamma-ray information for Z= 41 N= 53 ( 94Nb)

S-wave strength function parameters:

Exp. total radiative width= 0.14500 eV +/- 0.01000 Theor. total radiative wid
Exp. D0                      = 80.00 eV +/- 10.00 Theor. D0
Exp. S-wave strength func.= 0.45000E-4 +/- 0.07000 Theor. S-wave strength fun
Normalization factor         = 0.70061

Gamma-ray strength function model for E1: Kopecky-Uhl

```

Normalized gamma-ray strength functions and transmission coefficients for  $l=1$

Giant resonance parameters :

```
sigma0(M1) = 2.515          sigma0(E1) = 192.148
  E(M1) = 9.017            E(E1) = 16.523
gamma(M1) = 4.000          gamma(E1) = 5.515
  k(M1) = 8.67373E-08      k(E1) = 8.67373E-08
```

E	f(M1)	f(E1)	T(M1)	T(E1)
0.001	0.00000E+00	1.36200E-08	0.00000E+00	8.55769E-17
0.002	7.39798E-13	1.36205E-08	3.71863E-20	6.84642E-16
0.005	1.84950E-12	1.36221E-08	1.45259E-18	1.06988E-14
0.010	3.69900E-12	1.36248E-08	2.32415E-17	8.56071E-14
0.020	7.39805E-12	1.36301E-08	3.71867E-16	6.85126E-13
0.050	1.84960E-11	1.36461E-08	1.45267E-14	1.07176E-11
0.100	3.69981E-11	1.36725E-08	2.32466E-13	8.59069E-11

.....  
which is repeated for each  $l$ -value. Finally, the photoabsorption cross section is printed:

Photoabsorption cross sections

E	reaction
0.001	2.2413E-04
0.002	5.5010E-04
0.005	1.2228E-03
0.010	2.3445E-03
0.020	4.5901E-03
0.050	1.1345E-02
0.100	2.2662E-02
0.200	4.5515E-02
0.300	6.8659E-02
0.400	9.2090E-02
0.500	1.1581E-01

#### Case e: Detailed pre-equilibrium information

The single- and double-differential spectra have already been covered in sample 1b. In addition to this, the contribution of the pre-equilibrium mechanism to the spectra and cross sections can be printed in more detail with the **outpreequilibrium** keyword. With the input file

```
#
# General
#
projectile n
element nb
mass 93
```

```
energy 14.
#
# Output
#
outpreequilibrium y
outspectra y
ddxmode 2
```

we obtain, in addition to the aforementioned output blocks, a detailed outline of the pre-equilibrium model used, in this case the default: the two-component exciton model. First, the parameters for the exciton model are printed, followed by the matrix elements as a function of the exciton number:

```
##### PRE-EQUILIBRIUM #####
+++++++ TWO-COMPONENT EXCITON MODEL ++++++++

1. Matrix element for E= 21.076

Constant for matrix element : 1.000
p-p ratio for matrix element: 1.000
n-n ratio for matrix element: 1.500
p-n ratio for matrix element: 1.000
n-p ratio for matrix element: 1.000

p(p) h(p) p(n) h(n) M2pipi M2nunu M2pinu M2nupi
0 0 1 0 2.63420E-05 3.95131E-05 2.63420E-05 2.63420E-05
0 0 2 1 1.08884E-04 1.63327E-04 1.08884E-04 1.08884E-04
1 1 1 0 1.08884E-04 1.63327E-04 1.08884E-04 1.08884E-04
0 0 3 2 1.76643E-04 2.64964E-04 1.76643E-04 1.76643E-04
1 1 2 1 1.76643E-04 2.64964E-04 1.76643E-04 1.76643E-04
.....
```

Next, the emission rates are printed: first as function of particle type and particle-hole number, and in the last column summed over particles:

```
2. Emission rates or escape widths

A. Emission rates ( /sec)

p(p) h(p) p(n) h(n) gamma neutron proton deuteron triton
0 0 1 0 2.16570E+18 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00
0 0 2 1 5.29285E+17 1.63578E+21 0.00000E+00 0.00000E+00 0.00000E+00
1 1 1 0 5.58040E+17 8.15540E+20 2.32728E+20 2.79804E+19 0.00000E+00
0 0 3 2 1.26381E+17 4.52770E+20 0.00000E+00 0.00000E+00 0.00000E+00
1 1 2 1 1.08464E+17 3.04371E+20 1.11136E+19 3.36637E+18 9.21406E+1
.....
```

Also, the alternative representation in terms of the escape widths, see Eqs. (12.51) and (12.52),

is given,

B. Escape widths (MeV)

p(p)	h(p)	p(n)	h(n)	gamma	neutron	proton	deuteron	triton
0	0	1	0	1.42549E-03	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+0
0	0	2	1	3.48382E-04	1.07669E+00	0.00000E+00	0.00000E+00	0.00000E+0
1	1	1	0	3.67309E-04	5.36798E-01	1.53184E-01	1.84170E-02	0.00000E+0
0	0	3	2	8.31858E-05	2.98019E-01	0.00000E+00	0.00000E+00	0.00000E+0
1	1	2	1	7.13921E-05	2.00341E-01	7.31511E-03	2.21578E-03	6.06481E-0
2	2	1	0	9.92804E-05	8.90114E-02	1.26899E-02	1.26325E-03	0.00000E+0
.....								

The internal transition rates such as those of Eq. (12.17) and the associated damping and total widths are given next,

3. Internal transition rates or damping widths, total widths

A. Internal transition rates ( /sec)

p(p)	h(p)	p(n)	h(n)	lambdapiplus	lambdanuplus	lambdapinu	lambdanup
0	0	1	0	1.45456E+21	1.74977E+21	0.00000E+00	0.00000E+0
0	0	2	1	3.18197E+21	3.67047E+21	0.00000E+00	1.62232E+2
1	1	1	0	1.74347E+21	3.78846E+21	1.30118E+20	0.00000E+0
0	0	3	2	3.26507E+21	3.59251E+21	0.00000E+00	7.12888E+2
1	1	2	1	2.35211E+21	3.64486E+21	1.72552E+20	2.29011E+2
.....							

The lifetimes,  $t(p, h)$  of Eq. (12.40) and the depletion factors  $D_{p,h}$  of Eq. (12.41), are printed next,

4. Lifetimes

p(p)	h(p)	p(n)	h(n)	Strength
0	0	1	0	3.11867E-22
0	0	2	1	6.41109E-23
1	1	1	0	6.88639E-23
0	0	3	2	3.09624E-23
1	1	2	1	7.50433E-23
2	2	1	0	2.43181E-23
.....				

The partial state densities are printed for the first particle-hole combinations as a function of excitation energy. We also print the exciton number-dependent spin distributions and their sum, to see whether we have exhausted all spins. This output block is as follows

+++++++ PARTIAL STATE DENSITIES ++++++

Particle-hole state densities

```

Ex   P(n=3)   gp     gn
      1 1 0 0   0 0 1 1   1 1 1 0   1 0 1 1   2 1 0 0
1.000 0.000  2.733  3.533 7.471E+00 1.248E+01 9.728E+00 1.139E+01 0.000E+00
2.000 0.000  2.733  3.533 1.494E+01 2.497E+01 4.559E+01 5.633E+01 8.212E+00
3.000 0.000  2.733  3.533 2.241E+01 3.745E+01 1.078E+02 1.354E+02 2.627E+01
4.000 0.000  2.733  3.533 2.988E+01 4.994E+01 1.965E+02 2.486E+02 5.453E+01
5.000 0.000  2.733  3.533 3.736E+01 6.242E+01 3.116E+02 3.959E+02 9.301E+01
6.000 0.000  2.733  3.533 4.483E+01 7.491E+01 4.530E+02 5.774E+02 1.417E+02

```

.....  
Particle-hole spin distributions

```

n      J= 0      J= 1      J= 2      J= 3      J= 4      J= 5
1  1.7785E-02  4.3554E-02  4.8369E-02  3.6832E-02  2.1025E-02  9.3136E-03
2  6.3683E-03  1.7261E-02  2.3483E-02  2.4247E-02  2.0773E-02  1.5284E-02
3  3.4812E-03  9.7603E-03  1.4208E-02  1.6237E-02  1.5926E-02  1.3878E-02
4  2.2659E-03  6.4613E-03  9.7295E-03  1.1698E-02  1.2277E-02  1.1642E-02
5  1.6234E-03  4.6764E-03  7.1862E-03  8.9070E-03  9.7354E-03  9.7128E-03

```

.....  
We print a table with the pre-equilibrium cross sections per stage and outgoing energy, for each outgoing particle. At the end of each table, we give the total pre-equilibrium cross sections per particle. Finally the total pre-equilibrium cross section summed over outgoing particles is printed,

+++++++ TOTAL PRE-EQUILIBRIUM CROSS SECTIONS ++++++

Pre-equilibrium cross sections for gamma

```

E      Total      p=1      p=2      p=3      p=4      p=5
0.001 2.2249E-13 9.1810E-14 5.0384E-14 3.8551E-14 4.1750E-14 0.0000E+00 0.000
0.002 2.1767E-12 8.9578E-13 4.9358E-13 3.7794E-13 4.0941E-13 0.0000E+00 0.000
0.005 2.9926E-11 1.2218E-11 6.8126E-12 5.2280E-12 5.6681E-12 0.0000E+00 0.000
0.010 2.2566E-10 9.0941E-11 5.1690E-11 3.9811E-11 4.3221E-11 0.0000E+00 0.000
0.020 1.7108E-09 6.7253E-10 3.9631E-10 3.0739E-10 3.3461E-10 0.0000E+00 0.000
0.050 2.4211E-08 8.9017E-09 5.7581E-09 4.5550E-09 4.9961E-09 0.0000E+00 0.000
0.100 1.7108E-07 5.7425E-08 4.1833E-08 3.4050E-08 3.7774E-08 0.0000E+00 0.000
.....
19.000 1.3505E-01 1.3040E-01 4.3802E-03 2.6369E-04 4.8448E-06 0.0000E+00 0.000
19.500 1.2227E-01 1.1875E-01 3.4007E-03 1.2338E-04 6.6196E-07 0.0000E+00 0.000
20.000 1.1171E-01 1.0895E-01 2.7321E-03 3.1776E-05 0.0000E+00 0.0000E+00 0.000
21.000 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.000
22.000 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.000

      1.2864E+00 1.1291E+00 1.2096E-01 2.6279E-02 1.0140E-02 0.0000E+00 0.000

```

Integrated: 1.28644

Pre-equilibrium cross sections for neutron

E	Total	p=1	p=2	p=3	p=4	p=5	
0.001	9.4598E-02	0.0000E+00	2.6248E-02	2.5416E-02	2.0231E-02	1.4060E-02	8.642
0.002	1.4281E-01	0.0000E+00	3.9637E-02	3.8373E-02	3.0539E-02	2.1220E-02	1.304
0.005	2.6901E-01	0.0000E+00	7.4728E-02	7.2301E-02	5.7513E-02	3.9941E-02	2.453
0.010	4.8271E-01	0.0000E+00	1.3428E-01	1.2979E-01	1.0316E-01	7.1572E-02	4.391
0.020	9.7103E-01	0.0000E+00	2.7090E-01	2.6131E-01	2.0733E-01	1.4358E-01	8.790
.....							
17.500	2.5293E-01	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.000
18.000	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.000
18.500	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.000
19.000	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.000
	2.6727E+01	0.0000E+00	0.0000E+00	0.0000E+00	1.0326E-02	1.1088E-02	4.974

Integrated: 26.72690

Total pre-equilibrium cross section: 415.82114

#### Sample case 4.2 Case f: Discrete direct cross sections and angular distributions

More specific information on the characteristics of direct reactions can be obtained with the following input file,

```
#
# General
#
projectile n
element nb
mass 93
energy 14.
#
# Output
#
outdiscrete      y
outangle         y
outlegendre     y
outdirect       y
outspectra      y
```

Now we obtain, through **outdirect y**, the direct cross sections from inelastic collective scattering and giant resonances. The output block begins with

```
+++++++ DIRECT CROSS SECTIONS ++++++
Direct inelastic cross sections
```

Level	Energy	E-out	J/P	Cross section	Def. par.
1	0.03077	13.81777	0.5-	0.00670	B 0.00239
2	0.68709	13.16145	1.5-	0.01311	B 0.00338
3	0.74386	13.10468	3.5+	4.28791	B 0.04699
4	0.80849	13.04005	2.5+	0.08705	B 0.00873
5	0.81025	13.03829	2.5-	3.20756	B 0.04070
6	0.94982	12.89872	6.5+	7.43461	B 0.06216
.....					
92	12.99300	0.85554	0.5+	0.01342	B 0.01008
93	13.09000	0.75854	3.5-	0.00194	B 0.00797
94	13.54200	0.30654	2.5-	0.00680	B 0.00730
95	13.58100	0.26754	1.5+	0.00089	B 0.00437

Discrete direct inelastic cross section: 29.21914 Level 1- 20  
Collective cross section in continuum : 45.69146

which for the case of  $^{93}\text{Nb}$  gives the results of the weak-coupling model. For every level, the angular distribution is given, since **outangle y** was specified:

Direct inelastic angular distributions

Angle	Ex= 0.031 JP= 0.5-	Ex= 0.687 JP= 1.5-	Ex= 0.744 JP= 3.5+	Ex= 0.808 JP= 2.5+	Ex= 0.810 JP= 2.5-	Ex= 0.950 JP= 6.5+
0.0	1.06392E-03	2.02551E-03	1.31400E+00	1.33822E-02	9.82438E-01	2.27460E+00
2.0	1.06611E-03	2.03034E-03	1.30995E+00	1.34149E-02	9.79407E-01	2.26757E+00
4.0	1.07266E-03	2.04477E-03	1.29828E+00	1.35125E-02	9.70673E-01	2.24731E+00
6.0	1.08348E-03	2.06854E-03	1.28032E+00	1.36731E-02	9.57224E-01	2.21608E+00
8.0	1.09840E-03	2.10115E-03	1.25797E+00	1.38933E-02	9.40465E-01	2.17707E+00
10.0	1.11708E-03	2.14179E-03	1.23324E+00	1.41675E-02	9.21900E-01	2.13374E+00
.....						

The table with total giant resonance results is given next,

+++++++ GIANT RESONANCES ++++++

	Cross section	Exc. energy	Emis. energy	Width	Deform. par.
GMR :	0.00000	16.37500	-2.52646	3.00000	0.02456
GQR :	0.00000	14.34671	-0.49817	4.14092	0.14081
LEOR :	24.05840	6.84228	7.00626	5.00000	0.15788
HEOR :	0.00000	25.38264	-11.53410	7.36250	0.13210

Total: 24.05840

followed, since **outspectra y**, by the associated spectra,

Giant resonance spectra

Energy	Total	GMR	GQR	LEOR	HEOR	Collective
--------	-------	-----	-----	------	------	------------



```

0.001 7.7214E-03 0.0000E+00 0.0000E+00 7.7214E-03 0.0000E+00 0.0000E+00
0.002 7.7337E-03 0.0000E+00 0.0000E+00 7.7337E-03 0.0000E+00 0.0000E+00
0.005 7.7706E-03 0.0000E+00 0.0000E+00 7.7706E-03 0.0000E+00 0.0000E+00
0.010 7.8325E-03 0.0000E+00 0.0000E+00 7.8325E-03 0.0000E+00 0.0000E+00
0.020 7.9577E-03 0.0000E+00 0.0000E+00 7.9577E-03 0.0000E+00 0.0000E+00
0.050 8.3441E-03 0.0000E+00 0.0000E+00 8.3441E-03 0.0000E+00 0.0000E+00
.....

```

The total, i.e. direct + compound cross section per discrete level of each residual nucleus was already described for sample 1b. In addition, we have now requested the angular distributions and the associated Legendre coefficients. First, the angular distribution for elastic scattering, separated by direct and compound contribution, is given. Since **outlegendre y** it is given first in terms of Legendre coefficients. This output block begins with:

8. Discrete state angular distributions

8a1. Legendre coefficients for elastic scattering

L	Total	Direct	Compound	Normalized
0	1.75971E+02	1.75971E+02	4.71035E-05	7.95776E-02
1	1.55728E+02	1.55728E+02	0.00000E+00	7.04232E-02
2	1.39720E+02	1.39720E+02	1.86062E-06	6.31840E-02
3	1.21488E+02	1.21488E+02	0.00000E+00	5.49390E-02
4	1.02659E+02	1.02659E+02	3.25580E-07	4.64241E-02
.....				

where the final column means division of the Legendre coefficients by the cross section. This is followed by the associated angular distribution. This output block begins with:

8a2. Elastic scattering angular distribution

Angle	Total	Direct	Compound
0.0	6.69554E+03	6.69554E+03	6.12260E-05
2.0	6.62134E+03	6.62134E+03	6.11570E-05
4.0	6.40317E+03	6.40317E+03	6.09526E-05
6.0	6.05390E+03	6.05390E+03	6.06204E-05
8.0	5.59369E+03	5.59369E+03	6.01725E-05
10.0	5.04824E+03	5.04824E+03	5.96247E-05
12.0	4.44657E+03	4.44657E+03	5.89949E-05
.....			

Next, the Legendre coefficients for inelastic scattering to each discrete level, separated by the direct and compound contribution, is given. This output block begins with:

8b1. Legendre coefficients for inelastic scattering

Level 1

L	Total	Direct	Compound	Normalized
0	5.43660E-04	5.33254E-04	1.04057E-05	7.95775E-02
1	1.86109E-04	1.86109E-04	0.00000E+00	2.72415E-02
2	6.20916E-05	6.21244E-05	-3.27812E-08	9.08857E-03
3	1.66523E-05	1.66523E-05	0.00000E+00	2.43746E-03
4	-1.37214E-05	-1.35853E-05	-1.36110E-07	-2.00846E-03
5	-1.90397E-05	-1.90397E-05	0.00000E+00	-2.78691E-03
6	-1.61471E-05	-1.61584E-05	1.12224E-08	-2.36352E-03

.....  
 which is also followed by the associated angular distributions. This output block begins with:

8b2. Inelastic angular distributions

Level 1

Angle	Total	Direct	Compound
0.0	1.07320E-03	1.06392E-03	9.28143E-06
2.0	1.07540E-03	1.06611E-03	9.28519E-06
4.0	1.08196E-03	1.07266E-03	9.29647E-06
6.0	1.09280E-03	1.08348E-03	9.31533E-06
8.0	1.10774E-03	1.09840E-03	9.34185E-06
10.0	1.12646E-03	1.11708E-03	9.37618E-06
12.0	1.14844E-03	1.13902E-03	9.41848E-06

.....  
 Finally, the same is given for the  $(n, p)$  and the other channels.

### Case g: Discrete gamma-ray production cross sections

The gamma-ray intensity for each mother and daughter discrete level appearing in the reaction can be obtained with the following input file,

```
#
# General
#
projectile n
element nb
mass 93
energy 14.
#
# Output
#
outgamdis          y
```

For all discrete gamma-ray transitions, the intensity is printed. For each nucleus, the initial level and the final level is given, the associated gamma energy and the cross section. This output block begins with:

10. Gamma-ray intensities

Nuclide: 94Nb

Initial level			Final level			Gamma Energy	Cross section	
no.	J/Pi	Ex	no.	J/Pi	Ex			
2	4.0+	0.0587	--->	1	3.0+	0.0409	0.01780	2.11853E-01
3	6.0+	0.0787	--->	0	6.0+	0.0000	0.07867	2.12745E-01
4	5.0+	0.1134	--->	0	6.0+	0.0000	0.11340	8.56884E-02
4	5.0+	0.1134	--->	2	4.0+	0.0587	0.05470	3.36955E-02
5	2.0-	0.1403	--->	1	3.0+	0.0409	0.09941	2.43882E-01
6	2.0-	0.3016	--->	5	2.0-	0.1403	0.16126	5.42965E-02
7	4.0+	0.3118	--->	2	4.0+	0.0587	0.25311	5.48060E-02

.....

When we discuss multiple incident energy runs in the other sample cases, we will see how the excitation functions for gamma production cross sections per level are accumulated and how they can be written to separate files for easy processing.

#### Case h: The full output file

In this sample case we print basically everything that can be printed in the main output file for a single-energy reaction on a non-fissile nucleus. The input file is

```
#
# General
#
projectile n
element nb
mass 93
energy 14.
#
# Output
#
outbasic          y
outpreequilibrium y
outspectra       y
outangle         y
outlegendre      y
ddxmode          2
outgamdis        y
partable         y
```

resulting in an output file that contains all nuclear structure information, all partial results, and moreover all intermediate results of the calculation, as well as results of intermediate checking. Note that basically all flags in the "Output" block on top of the output file are set to **y**, the only exceptions being irrelevant for this sample case. In addition to the output that is already described, various other output blocks are present. First, since **outbasic y** automatically means **outomp y**, a block with optical model parameters is printed. The optical model parameters for all included particles are given as a function of incident energy. This output block begins with:

```
##### OPTICAL MODEL PARAMETERS #####

      neutron on 93Nb

Energy   V    rv    av    W    rw    aw    Vd   rvd   avd   Wd   rwd   awd

 0.001  51.02 1.215 0.663  0.14 1.215 0.663  0.00 1.274 0.534  3.32 1.274 0.53
 0.002  51.02 1.215 0.663  0.14 1.215 0.663  0.00 1.274 0.534  3.32 1.274 0.53
 0.005  51.02 1.215 0.663  0.14 1.215 0.663  0.00 1.274 0.534  3.32 1.274 0.53
 0.010  51.02 1.215 0.663  0.14 1.215 0.663  0.00 1.274 0.534  3.33 1.274 0.53
 0.020  51.02 1.215 0.663  0.14 1.215 0.663  0.00 1.274 0.534  3.33 1.274 0.53
.....
```

In the next part, we print general quantities that are used throughout the nuclear reaction calculations, such as transmission coefficients and inverse reaction cross sections. The transmission coefficients as a function of energy are given for all particles included in the calculation. Depending upon whether **outtransenergy y** or **outtransenergy n**, the transmission coefficient tables will be grouped per energy or per angular momentum, respectively. The latter option may be helpful to study the behavior of a particular transmission coefficient as a function of energy. The default is **outtransenergy n**, leading to the following output block,

```
##### TRANSMISSION COEFFICIENTS AND INVERSE REACTION CROSS SECTIONS #####

Transmission coefficients for incident neutron at 0.00101 MeV

L   T(L-1/2,L)   T(L+1/2,L)   Tav(L)

0  0.00000E+00  8.82145E-03  8.82145E-03
1  1.45449E-04  2.56136E-04  2.19240E-04

Transmission coefficients for incident neutron at 0.00202 MeV

L   T(L-1/2,L)   T(L+1/2,L)   Tav(L)

0  0.00000E+00  1.24534E-02  1.24534E-02
1  4.11628E-04  7.24866E-04  6.20454E-04
2  2.55785E-08  1.88791E-08  2.15588E-08

Transmission coefficients for incident neutron at 0.00505 MeV

L   T(L-1/2,L)   T(L+1/2,L)   Tav(L)

0  0.00000E+00  1.96205E-02  1.96205E-02
1  1.62890E-03  2.86779E-03  2.45483E-03
2  2.52135E-07  1.86214E-07  2.12582E-07
.....
```

which is repeated for each included particle type. Next, the (inverse) reaction cross sections is given for all particles on a LAB energy grid. For neutrons also the total elastic and total cross

section on this energy grid is printed for completeness. This output block begins with:

Total cross sections for neutron

E	total	reaction	elastic	OMP reaction
0.00101	1.1594E+04	6.2369E+03	5.3566E+03	6.2369E+03
0.00202	1.0052E+04	4.7092E+03	5.3430E+03	4.7092E+03
0.00505	8.8645E+03	3.5518E+03	5.3127E+03	3.5518E+03
0.01011	8.4660E+03	3.1918E+03	5.2741E+03	3.1918E+03
0.02022	8.4357E+03	3.2208E+03	5.2148E+03	3.2208E+03
0.05054	8.9304E+03	3.8249E+03	5.1055E+03	3.8249E+03
0.10109	9.6344E+03	4.5808E+03	5.0536E+03	4.5808E+03
0.20217	1.0198E+04	5.0370E+03	5.1609E+03	5.0370E+03
0.30326	1.0068E+04	4.7878E+03	5.2799E+03	4.7878E+03
0.40434	9.6390E+03	4.3377E+03	5.3013E+03	4.3377E+03
0.50543	9.1158E+03	3.8893E+03	5.2264E+03	3.8893E+03
0.60651	8.5881E+03	3.5056E+03	5.0825E+03	3.5056E+03
0.70760	8.0915E+03	3.1968E+03	4.8946E+03	3.1968E+03

The final column "OMP reaction" gives the reaction cross section as obtained from the optical model. This is not necessarily the same as the adopted reaction cross section of the middle column, since sometimes (especially for complex particles) this is overruled by systematics, see the **sysreaction** keyword, (p. 521). For the incident energy, we separately print the OMP parameters, the transmission coefficients and the shape elastic angular distribution,

+++++++ OPTICAL MODEL PARAMETERS FOR INCIDENT CHANNEL +++++++

neutron on 93Nb

Energy	V	rv	av	W	rw	aw	Vd	rvd	avd	Wd	rwd	awd
14.000	46.11	1.215	0.663	0.98	1.215	0.663	0.00	1.274	0.534	6.84	1.274	0.53

Optical model results

Total cross section : 3.9819E+03 mb

Reaction cross section: 1.7706E+03 mb

Elastic cross section : 2.2113E+03 mb

Transmission coefficients for incident neutron at 14.000 MeV

L	T(L-1/2,L)	T(L+1/2,L)	Tav(L)
0	0.00000E+00	7.46404E-01	7.46404E-01
1	8.02491E-01	7.78050E-01	7.86197E-01
2	7.77410E-01	8.08483E-01	7.96054E-01
3	7.75550E-01	6.94249E-01	7.29092E-01
4	9.15115E-01	9.53393E-01	9.36381E-01
5	6.01374E-01	6.19435E-01	6.11226E-01

```

6 7.00430E-01 4.72026E-01 5.77443E-01
7 1.15102E-01 1.90743E-01 1.55444E-01
8 1.59959E-02 1.88919E-02 1.75291E-02
9 2.36267E-03 2.49532E-03 2.43249E-03
10 3.55525E-04 3.61442E-04 3.58625E-04
11 5.38245E-05 5.39431E-05 5.38864E-05
12 8.20113E-06 8.17260E-06 8.18630E-06
13 1.26174E-06 1.25428E-06 1.25788E-06

```

Shape elastic scattering angular distribution

Angle	Cross section
-------	---------------

0.0	6.69554E+03
2.0	6.62134E+03
4.0	6.40317E+03
6.0	6.05390E+03
8.0	5.59369E+03
10.0	5.04824E+03
12.0	4.44657E+03
14.0	3.81867E+03
16.0	3.19323E+03
18.0	2.59561E+03
20.0	2.04638E+03

.....

At some point during a run, TALYS has performed the direct reaction calculation and the pre-equilibrium calculation. A table is printed which shows the part of the reaction population that is left for the formation of a compound nucleus. Since the pre-equilibrium cross sections are calculated on an emission energy grid, there is always a small numerical error when transferring these results to the excitation energy grid. The pre-equilibrium spectra are therefore normalized. The output block looks as follows

```
##### POPULATION CHECK #####
```

Particle Pre-equilibrium Population

gamma	1.28644	1.27794
neutron	428.71729	410.91696
proton	25.24361	25.20203
deuteron	3.50197	3.49032
triton	0.09480	0.09451
helium-3	0.00000	0.00000
alpha	26.72690	26.64193

```
+++++++ Normalization of reaction cross section ++++++
```

Reaction cross section	: 1770.63000 (A)
Sum over T(j,l)	: 1770.62537 (B)
Compound nucleus formation c.s.	: 1243.46240 (C)
Ratio C/B	: 0.70227

After the compound nucleus calculation, the results from the binary reaction are printed. First, the binary cross sections for the included outgoing particles are printed, followed by, if **outspectra y**, the binary emission spectra. If also **outcheck y**, the integral over the emission spectra is checked against the cross sections. The printed normalization factor has been applied to the emission spectra. This output block begins with:

```
##### BINARY CHANNELS #####
```

```
+++++++ BINARY CROSS SECTIONS ++++++
```

```
gamma    channel to Z= 41 N= 53 ( 94Nb): 2.23665E+00
neutron  channel to Z= 41 N= 52 ( 93Nb): 1.68950E+03
proton   channel to Z= 40 N= 53 ( 93Zr): 3.79183E+01
deuteron channel to Z= 40 N= 52 ( 92Zr): 1.04112E+01
triton   channel to Z= 40 N= 51 ( 91Zr): 6.70826E-01
helium-3 channel to Z= 39 N= 52 ( 91Y ): 2.35820E-08
alpha    channel to Z= 39 N= 51 ( 90Y ): 2.98966E+01
```

Binary emission spectra

Energy	gamma	neutron	proton	deuteron	triton	helium-3
0.001	1.14462E-06	1.89788E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00
0.002	2.28923E-06	3.74165E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00
0.005	5.72311E-06	9.25456E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00
0.010	1.14464E-05	1.84461E+01	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00
0.020	2.28940E-05	3.68901E+01	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00
0.050	5.72550E-05	9.26411E+01	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00
0.100	1.14633E-04	1.86293E+02	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00
0.200	2.30055E-04	3.28686E+02	2.64905E-30	1.35926E-43	0.00000E+00	0.00000E+00
0.300	7.48114E-04	4.14446E+02	3.56019E-23	9.99027E-34	9.03852E-41	0.00000E+00
0.400	1.81798E-03	4.97986E+02	5.12047E-19	1.48517E-27	3.14528E-33	0.00000E+00
0.500	2.88939E-03	5.48371E+02	3.42302E-16	1.47275E-23	2.43692E-28	0.00000E+00
0.600	3.96241E-03	5.39365E+02	4.07677E-14	1.28871E-20	9.22918E-25	0.00000E+00
0.700	5.03710E-03	5.29973E+02	1.64259E-12	2.45195E-18	5.46266E-22	0.00000E+00

.....

```
+++++++ CHECK OF INTEGRATED BINARY EMISSION SPECTRA ++++++
```

	Continuum cross section	Integrated spectrum	Compound normalizatio
gamma	2.19555E+00	2.19555E+00	1.00011E+00
neutron	1.66025E+03	1.64343E+03	1.01805E+00
proton	3.43220E+01	3.43220E+01	1.00226E+00
deuteron	3.51689E+00	3.51689E+00	1.02829E+00
triton	9.54471E-02	9.54471E-02	1.01145E+00
helium-3	2.86109E-10	2.86109E-10	0.00000E+00
alpha	2.86177E+01	2.86176E+01	9.99189E-01

Since **outpopulation y**, the population that remains in the first set of residual nuclides after binary emission is printed,

```

+++++++ POPULATION AFTER BINARY EMISSION ++++++

Population of Z= 41 N= 53 ( 94Nb) after binary gamma emission: 2.23665E+00
Maximum excitation energy: 21.076 Discrete levels: 10 Continuum bins: 40 Conti

bin   Ex     Popul.    J= 0.0-   J= 0.0+   J= 1.0-   J= 1.0+   J= 2.0-   J= 2.0
0     0.000 3.206E-07 0.000E+00 0.000E+00 0.000E+00 0.000E+00 0.000E+00 0.000E+
1     0.041 1.889E-07 0.000E+00 0.000E+00 0.000E+00 0.000E+00 0.000E+00 0.000E+
2     0.059 2.525E-07 0.000E+00 0.000E+00 0.000E+00 0.000E+00 0.000E+00 0.000E+
3     0.079 1.528E-03 0.000E+00 0.000E+00 0.000E+00 0.000E+00 0.000E+00 0.000E+
4     0.113 3.443E-03 0.000E+00 0.000E+00 0.000E+00 0.000E+00 0.000E+00 0.000E+
5     0.140 1.051E-02 0.000E+00 0.000E+00 0.000E+00 0.000E+00 1.051E-02 0.000E+
6     0.302 9.579E-03 0.000E+00 0.000E+00 0.000E+00 0.000E+00 9.579E-03 0.000E+
7     0.312 1.818E-03 0.000E+00 0.000E+00 0.000E+00 0.000E+00 0.000E+00 0.000E+
8     0.334 4.717E-03 0.000E+00 0.000E+00 0.000E+00 0.000E+00 0.000E+00 0.000E+
9     0.396 6.486E-03 0.000E+00 0.000E+00 0.000E+00 0.000E+00 0.000E+00 0.000E+
10    0.450 3.014E-03 0.000E+00 0.000E+00 0.000E+00 0.000E+00 0.000E+00 0.000E+
11    0.708 3.665E-02 2.036E-04 1.520E-04 1.592E-03 1.152E-03 3.293E-03 2.540E-
12    1.224 5.702E-02 2.592E-04 1.935E-04 2.065E-03 1.494E-03 4.428E-03 3.415E-
.....

```

where in this case bins 0-10 concern discrete levels and bins 11-50 concern continuum bins.

After this output of the binary emission, we print for each nuclide in the decay chain the population as a function of excitation energy, spin and parity *before* it decays. This loop starts with the initial compound nucleus and the nuclides formed by binary emission. When all excitation energy bins of the nucleus have been depleted, the final production cross section (per ground state/isomer) is printed. The feeding from this nuclide to all its daughter nuclides is also given. If in addition **outspectra y**, the emission spectra for all outgoing particles from this nucleus are printed. At high incident energies, when generally **multipreeq y**, the result from multiple pre-equilibrium emission is printed (not included in this output). If **outcheck y**, it is checked whether the integral over the emission spectra from this nucleus is equal to the corresponding feeding cross section. This output block begins with:

```

##### MULTIPLE EMISSION #####

Population of Z= 41 N= 53 ( 94Nb) before decay: 3.58813E+00
Maximum excitation energy: 21.076 Discrete levels: 10 Continuum bins: 40 Conti

bin   Ex     Popul.    J= 0.0   J= 1.0   J= 2.0   J= 3.0   J= 4.0   J= 5.0
0     0.000 1.050E-06 0.000E+00 0.000E+00 0.000E+00 0.000E+00 0.000E+00 0.000E+
1     0.041 6.181E-07 0.000E+00 0.000E+00 0.000E+00 6.181E-07 0.000E+00 0.000E+
2     0.059 8.266E-07 0.000E+00 0.000E+00 0.000E+00 0.000E+00 8.266E-07 0.000E+
3     0.079 1.525E-03 0.000E+00 0.000E+00 0.000E+00 0.000E+00 0.000E+00 0.000E+
4     0.113 3.436E-03 0.000E+00 0.000E+00 0.000E+00 0.000E+00 0.000E+00 3.436E-
5     0.140 1.049E-02 0.000E+00 0.000E+00 1.049E-02 0.000E+00 0.000E+00 0.000E+
6     0.302 9.558E-03 0.000E+00 0.000E+00 9.558E-03 0.000E+00 0.000E+00 0.000E+
7     0.312 1.814E-03 0.000E+00 0.000E+00 0.000E+00 0.000E+00 1.814E-03 0.000E+
8     0.334 4.706E-03 0.000E+00 0.000E+00 0.000E+00 4.706E-03 0.000E+00 0.000E+

```



```

 9  0.396 6.472E-03 0.000E+00 0.000E+00 0.000E+00 6.472E-03 0.000E+00 0.000E+
10  0.450 3.008E-03 0.000E+00 0.000E+00 0.000E+00 3.008E-03 0.000E+00 0.000E+
11  0.708 3.657E-02 1.517E-04 1.150E-03 2.536E-03 3.531E-03 3.593E-03 2.702E-
12  1.224 5.693E-02 1.939E-04 1.497E-03 3.419E-03 5.023E-03 5.489E-03 4.513E-
13  1.739 6.580E-02 1.843E-04 1.443E-03 3.395E-03 5.210E-03 6.036E-03 5.335E-
14  2.255 7.347E-02 1.744E-04 1.381E-03 3.323E-03 5.274E-03 6.390E-03 5.973E-

```

.....  
 Emitted flux per excitation energy bin of Z= 41 N= 53 ( 94Nb):

bin	Ex	gamma	neutron	proton	deuteron	triton	heli
0	0.000	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000
1	0.041	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000
2	0.059	2.11853E-01	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000
3	0.079	2.12745E-01	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000
4	0.113	1.19384E-01	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000
5	0.140	2.43882E-01	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000
6	0.302	5.42965E-02	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000

.....  
 Emission cross sections to residual nuclei from Z= 41 N= 53 ( 94Nb):

```

gamma    channel to Z= 41 N= 53 ( 94Nb): 2.66180E+00
neutron  channel to Z= 41 N= 52 ( 93Nb): 1.05107E+00
proton   channel to Z= 40 N= 53 ( 93Zr): 1.67352E-03
deuteron channel to Z= 40 N= 52 ( 92Zr): 1.07608E-06
triton   channel to Z= 40 N= 51 ( 91Zr): 2.06095E-08
helium-3 channel to Z= 39 N= 52 ( 91Y ): 1.07027E-16
alpha    channel to Z= 39 N= 51 ( 90Y ): 4.69800E-04

```

Emission spectra from Z= 41 N= 53 ( 94Nb):

Energy	gamma	neutron	proton	deuteron	triton	helium-3
0.001	1.14218E-02	5.11503E-03	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+0
0.002	1.19276E-02	1.10501E-02	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+0
0.005	1.34434E-02	3.25383E-02	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+0
0.010	1.59698E-02	7.24291E-02	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+0
0.020	2.10227E-02	1.27402E-01	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+0
0.050	4.45166E-02	2.33433E-01	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+0

.....  
 ++++++++ CHECK OF INTEGRATED EMISSION SPECTRA ++++++++

	Cross section	Integrated spectrum	Average emission energy
gamma	2.66180E+00	2.66180E+00	1.776
neutron	1.05107E+00	1.05107E+00	1.174
proton	1.67352E-03	1.67334E-03	5.377
deuteron	1.07608E-06	7.97929E-07	5.635
triton	2.06095E-08	0.00000E+00	0.000
helium-3	1.07027E-16	0.00000E+00	0.000
alpha	4.69800E-04	4.69563E-04	10.831

Final production cross section of  $Z= 41$   $N= 53$  ( $^{94}\text{Nb}$ ):

```
Total      : 1.18344E+00
Ground state: 5.75062E-01
Level 1     : 6.08379E-01
```

Note that once a new nucleus is encountered in the reaction chain, all nuclear structure information for that nucleus is printed as well.

### Case i: No output at all

It is even possible to have an empty output file. With the following input file,

```
#
# General
#
projectile n
element nb
mass 93
energy 14.
#
# Output
#
outmain n
```

it is specified that even the main output should be suppressed. The sample output file should be empty. This can be helpful when TALYS is invoked as a subroutine from other programs and the output from TALYS is not required (if the communication is done e.g. through shared arrays or subroutine variables). We have not yet used this option ourselves. ■

## 4.2 Cross section definitions

In TALYS, cross sections for reactions to all open channels are calculated. Although the types of most of these partial cross sections are generally well known, it is appropriate to define them for completeness. This section concerns basically the book-keeping of the various cross sections, including all the sum rules they obey. The particular nuclear models that are needed to obtain them are described in Part II. Thus, we do not yet give the definition of cross sections in terms of more fundamental quantities. Unless otherwise stated, we use incident neutrons as example in what follows and we consider only photons ( $\gamma$ ), neutrons (n), protons (p), deuterons (d), tritons (t), helium-3 particles (h) and alpha particles ( $\alpha$ ) as competing particles. Also, to avoid an overburdening of the notation and the explanation, we will postpone the competition of fission to the last section of this Chapter.

### 4.2.1 Total cross sections

The most basic nuclear reaction calculation is that with the optical model, which will be explained in more detail in Section 10. Here, it is sufficient to summarize the relations that can be found in many nuclear reaction textbooks, namely that the optical model yields the *reaction cross section*  $\sigma_{\text{reac}}$  and, in the case of neutrons, the *total cross section*  $\sigma_{\text{tot}}$  and the *shape-elastic cross section*

$\sigma_{shape-el}$ . They are related by

$$\sigma_{tot} = \sigma_{shape-el} + \sigma_{reac}. \quad (4.2)$$

If the elastic channel is, besides shape elastic scattering, also fed by compound nucleus decay, the latter component is a part of the reaction cross section and is called the *compound elastic cross section*  $\sigma_{comp-el}$ . With this, we can define the *total elastic cross section*  $\sigma_{el}$ ,

$$\sigma_{el} = \sigma_{shape-el} + \sigma_{comp-el}, \quad (4.3)$$

and the *non-elastic cross section*  $\sigma_{non-el}$ ,

$$\sigma_{non-el} = \sigma_{reac} - \sigma_{comp-el}, \quad (4.4)$$

so that we can combine these equations to give

$$\sigma_{tot} = \sigma_{el} + \sigma_{non-el}. \quad (4.5)$$

The last equation contains the quantities that can actually be measured in an experiment. We also note that the competition between the many compound nucleus decay channels ensures that  $\sigma_{comp-el}$  rapidly diminishes for incident neutron energies above a few MeV, in which case  $\sigma_{non-el}$  becomes practically equal to  $\sigma_{reac}$ .

A further subdivision of the outcome of a nuclear reaction concerns the breakdown of  $\sigma_{non-el}$ : this cross section contains all the partial cross sections. For this we introduce the exclusive cross sections, from which all other cross sections of interest can be derived.

### 4.2.2 Exclusive cross sections

In this manual, we call a cross section *exclusive* when the outgoing channel is precisely specified by the type and number of outgoing particles (+ any number of photons). Well-known examples are the inelastic or  $(n, n')$  cross section and the  $(n, 2n)$  cross section, which corresponds with two, and *only* two, neutrons (+ accompanying photons) in the outgoing channel. We denote the *exclusive cross section* as  $\sigma^{ex}(i_n, i_p, i_d, i_t, i_h, i_\alpha)$ , where  $i_n$  stands for the number of outgoing neutrons, etc. In this notation, where the incident particle is assumed implicit, e.g. the  $(n, 2np)$  cross section is given by  $\sigma^{ex}(2, 1, 0, 0, 0, 0)$ , for which we will also use the shorthand notation  $\sigma_{n,2np}$ . For a non-fissile nucleus, the sum over all exclusive cross sections is equal to the non-elastic cross section

$$\sigma_{non-el} = \sum_{i_n=0}^{\infty} \sum_{i_p=0}^{\infty} \sum_{i_d=0}^{\infty} \sum_{i_t=0}^{\infty} \sum_{i_h=0}^{\infty} \sum_{i_\alpha=0}^{\infty} \sigma^{ex}(i_n, i_p, i_d, i_t, i_h, i_\alpha), \quad (4.6)$$

provided we impose that  $\sigma^{ex}(1, 0, 0, 0, 0, 0)$  is the exclusive *inelastic* cross section  $\sigma_{n,n'}$ , i.e. it does not include shape- or compound elastic scattering.

The precise calculation of exclusive cross sections and spectra is a complicated book-keeping problem which, to our knowledge, has not been properly documented. We will describe the exact formalism here. In what follows we use quantities with a prime for daughter nuclides and quantities without a prime for mother nuclides in a decay chain. Consider an excitation energy bin or discrete level  $E_x$  in a nucleus  $(Z, N)$ . Let  $P(Z, N, E_x)$  represent the population of this bin/level before it decays. Let  $s_k(Z, N, E_x, E_{x'})$  be the part of the population that decays from the  $(Z, N, E_x)$  bin/level to the residual  $(Z', N', E_{x'})$  bin/level, whereby  $(Z, N)$  and  $(Z', N')$  are connected through the particle type  $k$ , with the index  $k$  running from  $\gamma$ -rays up to  $\alpha$ -particles. With these definitions, we can link

the various residual nuclides while keeping track of all intermediate particle emissions. A special case for the population is the initial compound nucleus  $(Z_C, N_C)$ , which contains all the initial reaction population at its total excitation energy  $E_x^{max}$  (projectile energy + binding energy), i.e.

$$P(Z_C, N_C, E_x^{max}) = \sigma_{non-el}, \quad (4.7)$$

while all other population bins/levels are zero. For the initial compound nucleus,  $s_k(Z_C, N_C, E_x^{max}, E_{x'})$  represents the binary feeding to the excitation energy bins of the first set of residual nuclides. This term generally consists of direct, pre-equilibrium and compound components.

The population of any bin in the decay chain is equal to the sum of the decay parts for all particles that can reach this bin from the associated mother bins, i.e.

$$P(Z', N', E_{x'}(i')) = \sum_{k=\gamma, n, p, d, t, h, \alpha} \sum_i s_k(Z, N, E_x(i), E_{x'}(i')), \quad (4.8)$$

where the sum over  $i$  runs over discrete level and continuum energy bins in the energy range from  $E_{x'}(i') + S_k$  to  $E_x^{max}(Z, N)$ , where  $S_k$  is the separation energy of particle  $k$  so that the sum only includes decay that is energetically allowed, and  $E_x^{max}(Z, N)$  is the maximum possible excitation energy of the  $(Z, N)$  nucleus. Note again that the particle type  $k$  determines  $(Z, N)$ .

To obtain the exclusive cross sections, we need to start with the initial compound nucleus and work our way down to the last nucleus that can be reached. First, consider a daughter nucleus  $(Z', N')$  somewhere in the reaction chain. We identify all exclusive channels  $(i_n, i_p, i_d, i_t, i_h, i_\alpha)$  that lead to this residual  $(Z', N')$  nucleus, i.e. all channels that satisfy

$$\begin{aligned} i_n + i_d + 2i_t + i_h + 2i_\alpha &= N_C - N' \\ i_p + i_d + i_t + 2i_h + 2i_\alpha &= Z_C - Z'. \end{aligned} \quad (4.9)$$

For each of these channels, the *inclusive* cross section per excitation energy bin  $S$  is equal to the sum of the feeding from all possible mother bins, i.e.

$$\begin{aligned} S(i_n, i_p, i_d, i_t, i_h, i_\alpha, E_{x'}(i')) &= \sum_{k=\gamma, n, p, d, t, h, \alpha} \sum_i \frac{s_k(Z, N, E_x(i), E_{x'}(i'))}{P(Z, N, E_x(i))} \\ \times S(i_n - \delta_{nk}, i_p - \delta_{pk}, i_d - \delta_{dk}, i_t - \delta_{tk}, i_h - \delta_{hk}, i_\alpha - \delta_{\alpha k}, E_x(i)), \end{aligned} \quad (4.10)$$

where we introduce Kronecker delta's, with characters as subscript, as

$$\begin{aligned} \delta_{nk} &= 1 \text{ if } k = n \text{ (neutron)} \\ &= 0, \text{ otherwise} \end{aligned} \quad (4.11)$$

and similarly for the other particles. Note that  $S$  is still inclusive in the sense that it is not yet depleted for further decay. The summation runs over the excitation energies of the mother bin from which decay into the  $E_{x'}(i')$  bin of the residual nucleus is energetically allowed. Feeding by gamma decay from bins above the  $(Z', N', E_{x'}(i'))$  bin is taken into account by the  $k = \gamma$  term, in which case all of the Kronecker delta's are zero.

With Eq. (4.7) as initial condition, the recursive procedure is completely defined. For a fixed nucleus, Eq. (4.10) is calculated for all excitation energy bins, in decreasing order, until the remaining population is in an isomeric or the ground state of the nucleus. When there is no further decay possible, the exclusive cross section per ground state/isomer, numbered by  $i$ , can be identified,

$$\sigma_i^{ex}(i_n, i_p, i_d, i_t, i_h, i_\alpha) = S(i_n, i_p, i_d, i_t, i_h, i_\alpha, E_i). \quad (4.12)$$

The total exclusive cross section for a particular channel is then calculated as

$$\sigma^{ex}(i_n, i_p, i_d, i_t, i_h, i_\alpha) = \sum_{i=0, isomers} \sigma_i^{ex}(i_n, i_p, i_d, i_t, i_h, i_\alpha). \quad (4.13)$$

The procedure outlined above automatically sorts and stores all exclusive cross sections, irrespective of the order of particle emission within the reaction chain. For example, the  $(n, np)$  and  $(n, pn)$  channels are automatically added. The above formalism holds *exactly* for an arbitrary number of emitted particles.

We stress that keeping track of the excitation energy  $E_x$  throughout this formalism is essential to get the exact exclusive cross sections for two reasons:

- (i) the exact determination of the branching ratios for exclusive isomeric ratios. The isomeric ratios for different exclusive reactions that lead to the same residual product, e.g.  $(n, np)$  and  $(n, d)$ , both leading to  $(Z_C - 1, N_C - 1)$ , are generally different from each other and thus also from the isomeric ratios of the total residual product. Hence, it would be an approximation to apply isomeric branching ratios for residual products, obtained after the full reaction calculation, a posteriori on the exclusive channels. This is avoided with our method,
- (ii) the exclusive spectra, which we will explain in Section 4.3.2.

Suppose one would only be interested in the total exclusive cross sections of Eq. (4.13), i.e. neither in the exclusive isomeric ratios, nor in the exclusive spectra. Only in that case, a simpler method would be sufficient. Since only the total reaction population that decays from nucleus to nucleus needs to be tracked, the total exclusive cross section for a certain channel is easily determined by subtracting the total ongoing flux from the total feeding flux to this channel. This is described in e.g. Section II.E.f of the GNASH manual [1]. We note that the exact treatment of TALYS does not require a large amount of computing time, certainly not when compared with more time-expensive parts of the full calculation.

When TALYS computes the binary reaction models and the multiple pre-equilibrium and Hauser-Feshbach models, it stores both  $P(Z, N, E_x)$  (through the *popexcl* array) and  $s_k(Z, N, E_x, E_{x'})$  (through the *feedexcl* array) for all residual nuclei and particles. This temporary storage enables us to first complete the full reaction calculation, including all its physical aspects, until all channels are closed. Then, we turn to the exclusive cross section and spectra problem afterwards in a separate subroutine: *channels.f*. It is thus considered as an isolated book-keeping problem.

The total number of different exclusive channels rapidly increases with the number of reaction stages. It can be shown that for  $m$  outgoing particles (i.e. reaction stages) which can be of  $k$  different types, the maximum number of exclusive cross sections is  $\binom{m+k-1}{m}$ . In general, we include neutrons up to alpha-particles as competing particles, i.e.  $k = 6$ , giving  $\binom{m+5}{m}$  possible exclusive channels, or 6, 21, 56 and 126, respectively, for the first 4 stages. This clarifies why exclusive channels are usually only of interest for only a few outgoing particles (the ENDF format for evaluated data libraries includes reactions up to 4 particles). At higher energies, and thus more outgoing particles, exclusive cross sections lose their relevance and the cross sections per channel are usually accumulated in the total particle production cross sections and residual production cross sections. Certainly at higher energies, this apparent loss of information is no longer an issue, since the observable quantities to which nuclear models can be tested are of a cumulative nature anyway, when many particles are involved.

In TALYS, the cumulated particle production cross sections and residual production cross sections are always *completely* tracked down until all residual nuclides have decayed to an isomer or the ground state, regardless of the incident energy, whereas exclusive cross sections are only tracked up to a user-defined depth. To elucidate this important point we discuss the low and the

high energy case. For low energies, say up to 20 MeV, keeping track of the exclusive cross section is important from both the fundamental and the applied point of view. It can be imagined that in a  $(n, np)$  measurement both the emitted neutron and proton have been measured in the detector. Hence the cross section is *not* the same as that of an activation measurement of the final residual nucleus, since the latter would also include a contribution from the  $(n, d)$  channel. Another example is the  $(n, 2n)$  channel, distinguished from the  $(n, n')$  or the general  $(n, xn)$  cross section, which is of importance in some integral reactor benchmarks. If, on the other hand, we encounter in the literature a cross section of the type  $^{120}\text{Sn}(p, 7p18n)^{96}\text{Ru}$ , we can be sure that the residual product  $^{96}\text{Ru}$  was measured and not the indicated number of neutrons and protons in a detector. The  $(p, 7p18n)$  symbol merely represents the number of neutron and proton units, while other light ions are generally included in the emission channel. Hence, for high energies the outcome of a nuclear reaction is usually tracked in parallel by two sets of quantities: the  $(proj, xn), \dots (proj, x\alpha)$  particle production cross sections and spectra, and the residual production cross sections. These will be exactly defined in the next sections.

### 4.2.3 Binary cross sections

Some of the exclusive channels need, and get, more attention than others. The exclusive binary cross sections, for reactions that are characterized by one, and only one, particle out, are special in the sense that they comprise both discrete and continuous energy transitions. Inelastic scattering can occur through both direct collective and compound transitions to the first few excited levels and through pre-equilibrium and compound reactions to the continuum. Let us assume that for a target nucleus the basic structure properties (spin, parity, deformation parameters) of the first  $N$  levels are known. Then, the *inelastic cross section*,  $\sigma_{n,n'}$  is the sum of a *total discrete inelastic cross section*  $\sigma_{n,n'}^{disc}$  and a *continuum inelastic cross section*  $\sigma_{n,n'}^{cont}$

$$\sigma_{n,n'} = \sigma_{n,n'}^{disc} + \sigma_{n,n'}^{cont}, \quad (4.14)$$

where  $\sigma_{n,n'}^{disc}$  is the sum over the inelastic cross sections for all the individual discrete states

$$\sigma_{n,n'}^{disc} = \sum_{i=1}^N \sigma_{n,n'}^i. \quad (4.15)$$

A further breakdown of each term is possible by means of reaction mechanisms. The inelastic cross section for each individual state  $i$  has a direct and a compound contribution:

$$\sigma_{n,n'}^i = \sigma_{n,n'}^{i,direct} + \sigma_{n,n'}^{i,comp}, \quad (4.16)$$

where the direct component comes from DWBA or coupled-channels calculations. Similarly, for the inelastic scattering to the continuum we can consider a pre-equilibrium and a compound contribution

$$\sigma_{n,n'}^{cont} = \sigma_{n,n'}^{PE} + \sigma_{n,n'}^{cont,comp}. \quad (4.17)$$

The set of definitions (4.14-4.17) can be given in a completely analogous way for the other binary channels  $\sigma_{n,p}$ , i.e.  $\sigma^{ex}(0, 1, 0, 0, 0, 0)$ ,  $\sigma_{n,d}$ ,  $\sigma_{n,t}$ ,  $\sigma_{n,h}$  and  $\sigma_{n,\alpha}$ . For the depletion of the reaction population that goes into the pre-equilibrium channels, which will be discussed in Section 12, it is helpful to define here the *total discrete direct cross section*,

$$\sigma^{disc,direct} = \sum_i \sum_{k=n',p,d,t,h,\alpha} \sigma_{n,k}^{i,direct}. \quad (4.18)$$

Finally, we also consider an alternative division for the non-elastic cross section. It is equal to the sum of the *inclusive* binary cross sections

$$\sigma_{non-el} = \sum_{k=\gamma,n',p,d,t,h,\alpha} \sigma_{n,k}^{inc,bin}, \quad (4.19)$$

where again at the present stage of the outline we do not consider fission and ejectiles heavier than  $\alpha$ -particles. This is what we actually use in the inclusive nuclear reaction calculations. With the direct, pre-equilibrium and compound models, several residual nuclides can be formed after the binary reaction, with a total population per nucleus that is equal to the terms of Eq. (4.19). The residual nuclides then decay further until all channels are closed. Note that  $\sigma^{inc,bin}$  is not a “true” cross section in the sense of a quantity for a final combination of a product and light particle(s).

#### 4.2.4 Total particle production cross sections

Especially for incident energies higher than about 10 MeV, it is appropriate to define the composite or *total neutron production cross section*,  $\sigma_{n,xn}$ . It can be expressed in terms of the exclusive cross sections as follows

$$\sigma_{n,xn} = \sum_{i_n=0}^{\infty} \sum_{i_p=0}^{\infty} \sum_{i_d=0}^{\infty} \sum_{i_t=0}^{\infty} \sum_{i_h=0}^{\infty} \sum_{i_\alpha=0}^{\infty} i_n \sigma^{ex}(i_n, i_p, i_d, i_t, i_h, i_\alpha), \quad (4.20)$$

i.e. in the more common notation,

$$\sigma_{n,xn} = \sigma_{n,n'} + 2\sigma_{n,2n} + \sigma_{n,np} + 2\sigma_{n,2np} + \dots \quad (4.21)$$

Again,  $\sigma_{n,xn}$  is not a true cross section since the incident and outgoing channels are not exactly defined by its individual reaction components. (Contrary to our definition, in some publications  $\sigma_{n,xn}$  is used to indicate activation measurements of a whole string of isotopes (e.g.  $^{202-208}\text{Pb}$ ) in which case  $x$  is a number that varies case by case. In our work, this is called an exclusive cross section). The *neutron multiplicity*, or *yield*,  $Y_n$  is defined as

$$Y_n = \frac{\sigma_{n,xn}}{\sigma_{non-el}}. \quad (4.22)$$

Similarly, the *total proton production cross section*,  $\sigma_{n,xp}$  is defined as

$$\sigma_{n,xp} = \sum_{i_n=0}^{\infty} \sum_{i_p=0}^{\infty} \sum_{i_d=0}^{\infty} \sum_{i_t=0}^{\infty} \sum_{i_h=0}^{\infty} \sum_{i_\alpha=0}^{\infty} i_p \sigma^{ex}(i_n, i_p, i_d, i_t, i_h, i_\alpha), \quad (4.23)$$

and the proton multiplicity, or yield,  $Y_p$  is defined as

$$Y_p = \frac{\sigma_{n,xp}}{\sigma_{non-el}}, \quad (4.24)$$

and similarly for the other particles. We note that we do not, in practice, use Eq. (4.20) to calculate the composite particle production cross section. Instead, we first calculate the inclusive binary cross section of Eq. (4.19) and then, during the depletion of each residual nucleus by further decay we directly add the reaction flux, equal to the  $s_k(Z, N, E_x, E_{x'})$  term of Eq. (4.8), to  $\sigma_{n,xn}$ ,  $\sigma_{n,xp}$ , etc. This procedure has already been sketched in the multiple decay scheme at the beginning of this Chapter. In the output of TALYS, we include Eq. (4.20) only as a numerical check. For a few outgoing particles Eq. (4.20) should exactly hold. For higher energies and thus more outgoing particles (typically more than 4, see the **maxchannel** (p. 306) keyword) the exclusive cross sections are no longer tracked by TALYS and Eq. (4.20) can no longer be expected to hold numerically. Remember, however, that we always calculate the total particle production cross sections, irrespective of the number of outgoing particles, since we continue the multiple emission calculation until all residual nuclides are in their isomeric or ground states.

**Sample case 4.3 Excitation functions:  $^{208}\text{Pb}$  (n,n'), (n,2n), (n,p) etc**

Often we are not interested in only one incident energy, but in excitation functions of the cross sections. If more than one incident energy is given in the file specified by the **energy** keyword, it is helpful to have the results, for each type of cross section, in a table as a function of incident energy. TALYS will first calculate all quantities that remain equal for all incident energy calculations, such as the transmission coefficients. Next, it will calculate the results for each incident energy. When the calculation for the last incident energy has been completed, the cross sections are collected and printed as excitation functions in the output if **outexcitation y** (which is the default if there is more than one incident energy). Moreover, we can provide the results in separate files: one file per reaction channel. Consider the following input file

```
#
# n-Pb208-xs
#
# General
#
projectile n
element pb
mass 208
energy energies
#
# Parameters
#
gnorm 0.35
Rgamma 2.2
gn 81 208 9.4
gp 81 208 6.2
egr      82 209 12.0    E1 1
optmodfileN 82 n-Pb208.omp
#
# Output
#
channels y
filechannels y
filetotal y
fileresidual y
outdiscrete y
```

which provides all partial cross sections for neutrons incident on  $^{208}\text{Pb}$  for 46 incident energies, from 1 to 30 MeV, as given in the file *energies* that is present in this sample case directory. In the main output file, first the results per incident energy are given. At the end of the output file, there is an output block that begins with:

```
##### EXCITATION FUNCTIONS #####
```

The first table contains the most important total cross sections as a function of incident energy.



This output block begins with:

1. Total (binary) cross sections

Energy	Non-elastic	Elastic	Total	Comp. el.	Shape el.	Reaction
1.000E+00	5.4614E-01	4.8367E+03	4.8372E+03	1.7271E+03	3.1096E+03	1.7276E+03
1.200E+00	6.0181E-01	4.7812E+03	4.7818E+03	1.8195E+03	2.9618E+03	1.8201E+03
1.400E+00	6.7627E-01	4.9406E+03	4.9413E+03	1.9485E+03	2.9921E+03	1.9492E+03
1.600E+00	7.6416E-01	5.2462E+03	5.2469E+03	2.1039E+03	3.1423E+03	2.1046E+03
1.800E+00	8.6914E-01	5.6345E+03	5.6353E+03	2.2653E+03	3.3692E+03	2.2662E+03
2.000E+00	1.0015E+00	6.0464E+03	6.0474E+03	2.4110E+03	3.6354E+03	2.4120E+03
.....						

Next, the binary cross sections are printed. This output block begins with:

2. Binary non-elastic cross sections (non-exclusive)

Energy	gamma	neutron	proton	deuteron	triton	helium-3
1.000E+00	1.9672E-01	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
1.200E+00	2.4416E-01	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
1.400E+00	3.0773E-01	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
1.600E+00	3.8879E-01	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
1.800E+00	4.9418E-01	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
2.000E+00	6.3274E-01	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
.....						

Next, the total particle production cross sections are printed. Parts of this output block look as follows:

3. Total particle production cross sections

gamma production

Energy	Cross section	Multiplicity
1.000E+00	2.47648E-01	1.43346E-04
1.200E+00	3.14552E-01	1.72825E-04
1.400E+00	4.06594E-01	2.08595E-04
1.600E+00	5.23101E-01	2.48546E-04
1.800E+00	6.76670E-01	2.98596E-04
2.000E+00	8.80247E-01	3.64945E-04
2.200E+00	1.14449E+00	4.52788E-04
2.400E+00	1.48925E+00	5.70753E-04
.....		

neutron production

Energy	Cross section	Multiplicity
1.000E+00	4.26190E-03	2.46692E-06

```

1.200E+00 9.15765E-03 5.03151E-06
1.400E+00 1.71089E-02 8.77738E-06
1.600E+00 3.18516E-02 1.51339E-05
1.800E+00 5.42068E-02 2.39200E-05
2.000E+00 8.89349E-02 3.68719E-05
2.200E+00 1.41778E-01 5.60908E-05
2.400E+00 2.20738E-01 8.45971E-05
2.600E+00 3.30648E-01 1.24267E-04
2.800E+00 3.16809E+02 1.17802E-01
.....

```

Next in the output are the residual production cross sections. The output block begins with:

#### 4. Residual production cross sections

Production of Z= 82 A=209 (209Pb) - Total

Q-value = 3.937307  
E-threshold= 0.000000

Energy Cross section

```

1.000E+00 1.92452E-01
1.200E+00 2.35002E-01
1.400E+00 2.90625E-01
1.600E+00 3.56940E-01
1.800E+00 4.39971E-01
2.000E+00 5.43801E-01
2.200E+00 6.70818E-01
2.400E+00 8.26660E-01
2.600E+00 1.02344E+00
.....

```

and the remaining isotopes follow in decreasing order of mass and isotope.

The final part of the output, for this input file at least, concerns the exclusive reaction cross sections. This output block begins with:

#### 6. Exclusive cross sections

Emitted particles						reaction
n	p	d	t	h	a	
0	0	0	0	0	0	(n,g)

Q-value = 3.937307  
E-threshold= 0.000000

Energy Cross section Gamma c.s. c.s./res.prod.cs

```

1.000E+00 1.92452E-01 3.48872E-01 1.00000E+00
1.200E+00 2.35002E-01 4.35691E-01 1.00000E+00
1.400E+00 2.90625E-01 5.54769E-01 1.00000E+00

```

```

1.600E+00 3.56940E-01 6.97846E-01 1.00000E+00
1.800E+00 4.39971E-01 8.82793E-01 1.00000E+00
2.000E+00 5.43801E-01 1.12074E+00 1.00000E+00
.....
      Emitted particles      reaction
      n   p   d   t   h   a
      1   1   0   0   0   0      (n,np)

Q-value      =  -8.003758
E-threshold=   8.042576

Energy      Cross section Gamma c.s. c.s./res.prod.cs

1.000E+00 0.00000E+00 0.00000E+00 0.00000E+00
1.200E+00 0.00000E+00 0.00000E+00 0.00000E+00
1.400E+00 0.00000E+00 0.00000E+00 0.00000E+00
1.600E+00 0.00000E+00 0.00000E+00 0.00000E+00
1.800E+00 0.00000E+00 0.00000E+00 0.00000E+00
2.000E+00 0.00000E+00 0.00000E+00 0.00000E+00
2.200E+00 0.00000E+00 0.00000E+00 0.00000E+00
.....
2.000E+01 3.12374E+00 2.18877E+00 5.08526E-01
2.200E+01 8.55243E+00 8.69348E+00 6.06418E-01
2.400E+01 1.70454E+01 2.30865E+01 6.79074E-01
2.600E+01 2.70534E+01 4.51648E+01 7.31295E-01
2.800E+01 3.62134E+01 6.92506E+01 7.66934E-01
3.000E+01 4.31314E+01 9.02171E+01 7.89711E-01

```

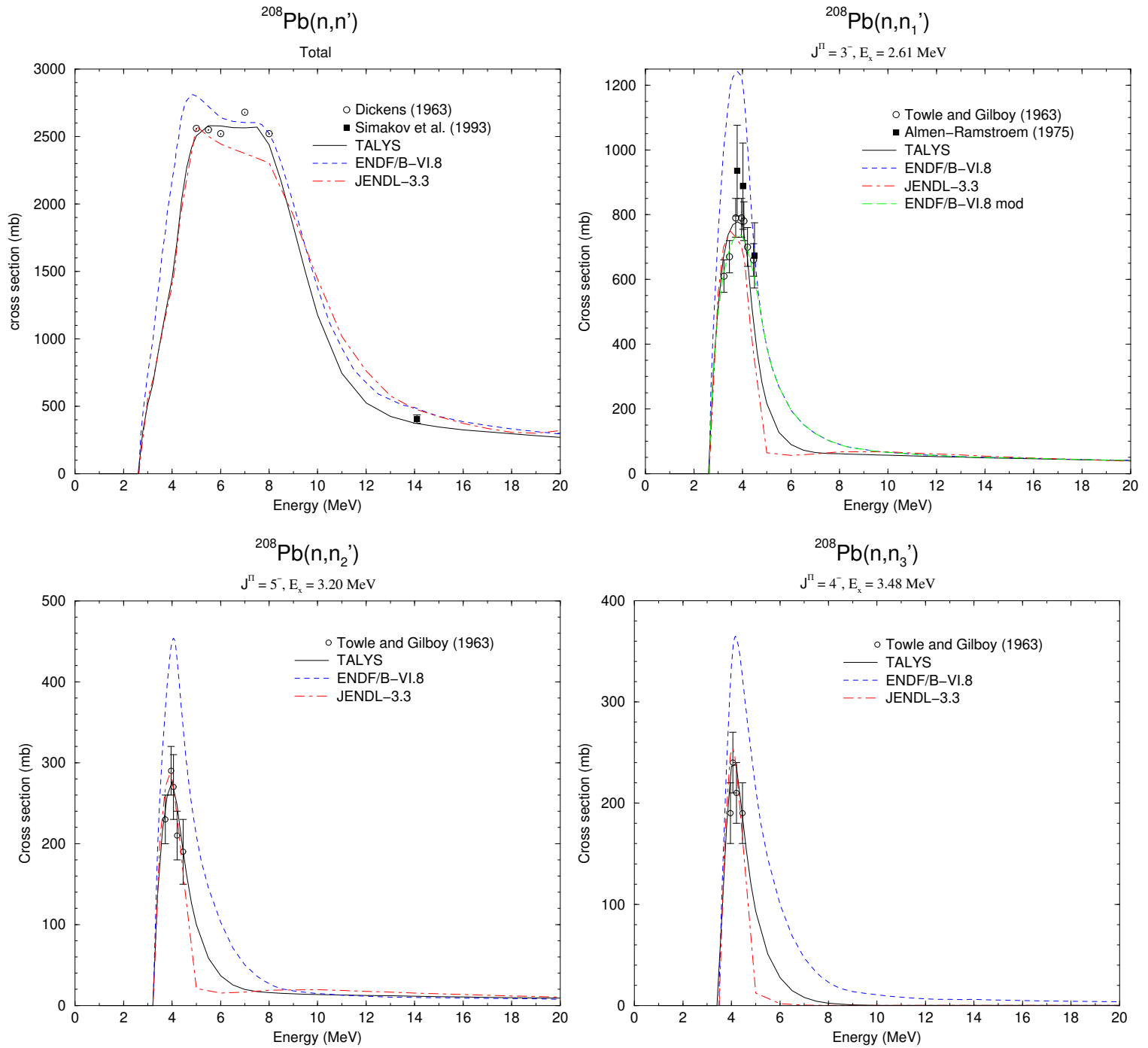
For plotting data, or processing into ENDF-6 data files, it is more practical to have the data in individual output files. Note that, since **filechannels y**, several files with names as e.g. *xs200000.tot* have been created in your working directory. These files contain the entire excitation function per reaction channel. Besides these exclusive cross sections, residual production cross section files are produced (**fileresidual y**). Note that for this reaction, *rp082207.tot* and *xs200000.tot* obviously have equal contents.

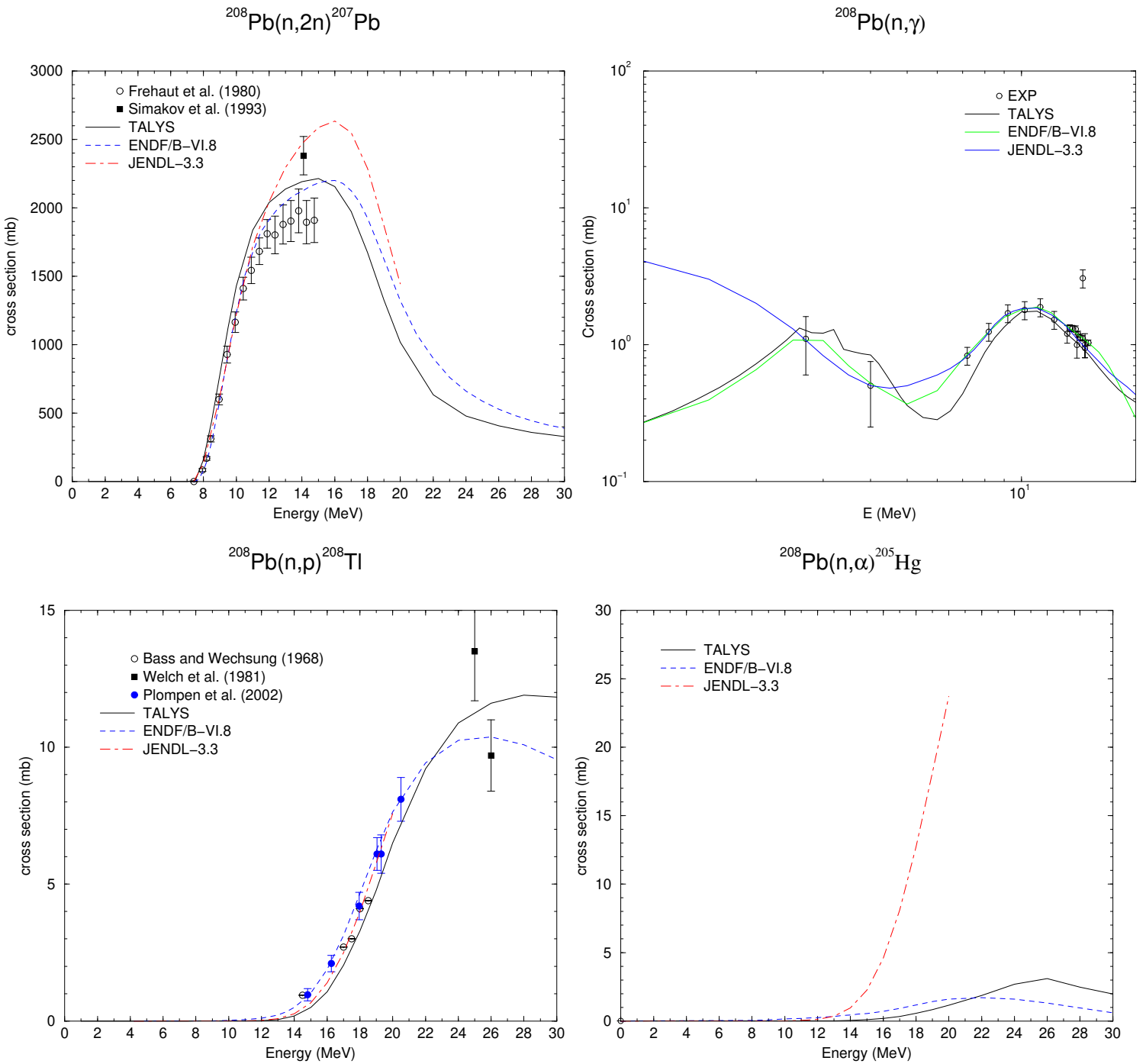
We illustrate this sample case with various comparisons with measurements. Since **filetotal y**, a file *total.tot* is created with, among others, the total cross section. The resulting curves are shown in Figs. 4.6 and 4.7. ■

### 4.2.5 Residual production cross sections

We can define another important type of derived cross section using the exclusive cross section, namely the residual production cross section  $\sigma_{prod}$ . All exclusive cross sections with the same number of neutron and proton units in the outgoing channel sum up to the same residual nucleus production cross section for the final nucleus ( $Z, N$ ), i.e.

$$\sigma_{prod}(Z, N) = \sum_{i_n=0}^{\infty} \sum_{i_p=0}^{\infty} \sum_{i_d=0}^{\infty} \sum_{i_t=0}^{\infty} \sum_{i_h=0}^{\infty} \sum_{i_\alpha=0}^{\infty} \sigma^{ex}(i_n, i_p, i_d, i_t, i_h, i_\alpha) \delta_N \delta_Z, \quad (4.25)$$

Figure 4.6: Partial cross sections for neutrons incident on  $^{208}\text{Pb}$ .

Figure 4.7: Partial cross sections for neutrons incident on  $^{208}\text{Pb}$ .

where the Kronecker delta's are defined by

$$\begin{aligned}\delta_N &= 1 \text{ if } i_n + i_d + 2i_t + i_h + 2i_\alpha = N_C - N \\ &= 0 \text{ otherwise} \\ \delta_Z &= 1 \text{ if } i_p + i_d + i_t + 2i_h + 2i_\alpha = Z_C - Z \\ &= 0 \text{ otherwise,}\end{aligned}\tag{4.26}$$

where the first compound nucleus that is formed from the projectile and target nucleus is denoted by  $(Z_C, N_C)$ . As an example, consider the  $n + {}^{56}\text{Fe} \rightarrow {}^{54}\text{Mn} + x$  reaction. The exclusive cross sections that add up to the  ${}^{54}\text{Mn}$  production cross section are  $\sigma_{n,2np}$ ,  $\sigma_{n,nd}$ , and  $\sigma_{n,t}$ , or  $\sigma^{ex}(2, 1, 0, 0, 0, 0)$ ,  $\sigma^{ex}(1, 0, 1, 0, 0, 0)$ , and  $\sigma^{ex}(0, 0, 0, 1, 0, 0)$ , respectively.

Since all exclusive cross sections contribute to the residual production cross section for one nuclide  $(Z, N)$  only, Eq. (4.6) automatically implies

$$\sigma_{non-el} = \sum_Z \sum_N \sigma_{prod}(Z, N).\tag{4.27}$$

Similar to Eq. (4.13), Eq. (4.25) is separated per isomer

$$\sigma_{prod,i}(Z, N) = \sum_{i_n=0}^{\infty} \sum_{i_p=0}^{\infty} \sum_{i_d=0}^{\infty} \sum_{i_t=0}^{\infty} \sum_{i_h=0}^{\infty} \sum_{i_\alpha=0}^{\infty} \sigma_i^{ex}(i_n, i_p, i_d, i_t, i_h, i_\alpha) \delta_N \delta_Z,\tag{4.28}$$

and the equivalent of Eq. (4.13) is

$$\sigma_{prod}(Z, N) = \sum_{i=0, isomers} \sigma_{prod,i}(Z, N).\tag{4.29}$$

Also here, we do not calculate  $\sigma_{prod}$  and  $\sigma_{prod,i}$  using Eqs. (4.25) and (4.28), although optionally TALYS includes it as a numerical check in the output for residual nuclides close to the target. Analogous to the total particle production, we determine the residual production cross section, for both the isomers and the ground state, after the complete decay of each nucleus by means of an inclusive calculation.

#### Sample case 4.4 Residual production cross sections for protons on Fe

In this sample case, we calculate the residual production cross sections for protons on  ${}^{nat}\text{Fe}$  for incident energies up to 100 MeV. A calculation for a natural target is launched, meaning that successive TALYS calculations for each isotope are performed, after which the results are weighted with the natural abundance. We restrict ourselves to a calculation with all nuclear model parameters set to their default values. The following input file is used:

```
#
# p-Fe000-rp
#
# General
#
projectile p
element fe
mass 0
energy energies
#
# Output
```

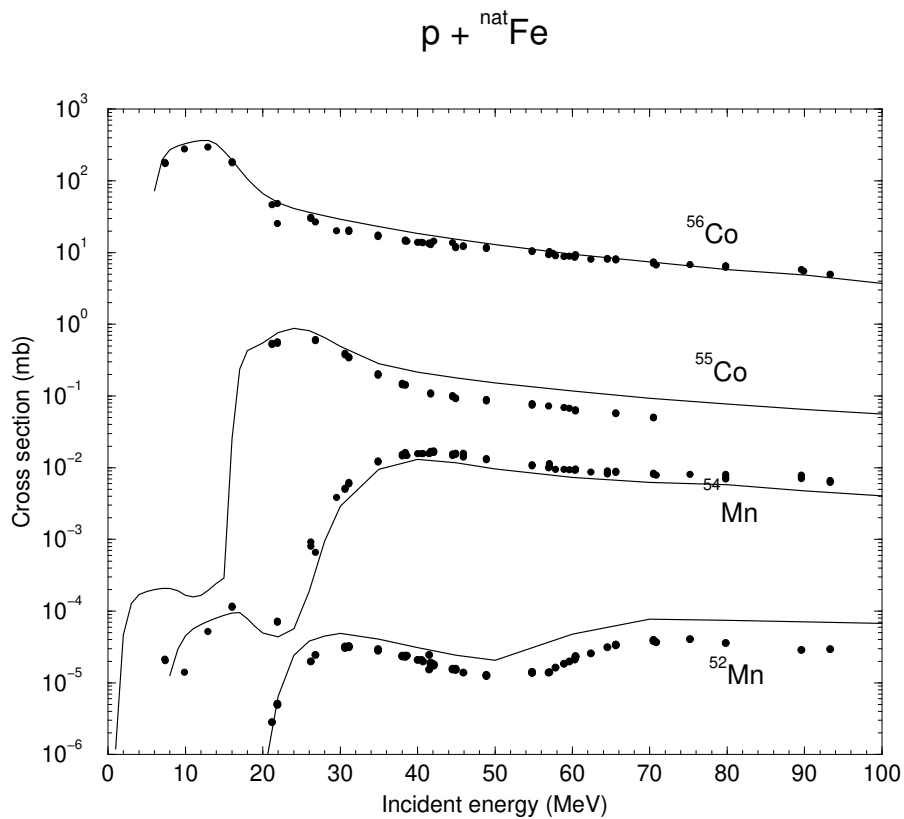


Figure 4.8: Residual production cross sections for protons incident on  ${}^{\text{nat}}\text{Fe}$ . Experimental data are obtained from [7].

```
#
fileresidual y
```

The file *energies* contains 34 incident energies between 1 and 100 MeV. Obviously, this sample case can be extended to more incident energies, e.g. up to 200 MeV, by simply adding numbers to the *energies* file. In that case, we recommend to include more energy bins in the calculation, (e.g. **bins 80**) to avoid numerical fluctuations, although this will inevitably take more computer time. Note that we have enabled the **fileresidual** keyword, so that a separate cross sections file for each final product is produced. The results from the files *rp027056.tot*, *rp027055.tot*, *rp025054.tot* and *rp025052.tot* are presented, together with experimental data, in Fig. 4.8. ■

#### Sample case 4.5 Gamma-ray intensities: ${}^{208}\text{Pb}(n, n\gamma)$ and ${}^{208}\text{Pb}(n, 2n\gamma)$

This feature could simply have been included in the sample case on excitation functions for  ${}^{208}\text{Pb}$ , but in order not to overburden the description of that sample case we include it here. With the input file

```
#
# n-Pb208-nngamma
#
```

```

# General
#
projectile n
element pb
mass 208
energy energies
#
# Parameters
#
isomer 1.e-4
maxZ 0
gnorm 0.35
Rgamma 2.2
egr      82 209 12.0   E1 1
optmodfileN 82 n-Pb208.omp
#
# Output
#
channels y
filechannels y
fileresidual y
outgamdis y

```

all discrete gamma lines are printed and stored in separate files. To avoid the production of too many data files, we have put **maxZ 0** so that only the gamma-ray production files for Pb-chain are created. Also, we include a special OMP with the file *pb.omp* and we set **isomer 1.e-4** to allow for gamma decay of some rather short-lived levels. Experimental data exists for the  $^{208}\text{Pb}(n, n'\gamma)$  cross section for level 1 to level 0 and the  $^{208}\text{Pb}(n, 2n'\gamma)$  cross section for level 2 to level 0 and for level 1 to level 0. These data have been plotted together with the results of the calculated files *gam082208L01L00.tot*, *gam082207L02L00.tot* and *gam082207L01L00.tot*, in Fig. 4.9. ■

### 4.3 Spectra and angular distributions

In addition to cross sections, TALYS also predicts energy spectra, angular distributions and energy-angle distributions.

#### 4.3.1 Discrete angular distributions

The elastic angular distribution  $\frac{d\sigma^{el}}{d\Omega}$  has a direct and a compound component:

$$\frac{d\sigma^{el}}{d\Omega} = \frac{d\sigma^{shape-el}}{d\Omega} + \frac{d\sigma^{comp-el}}{d\Omega}, \quad (4.30)$$

where the shape-elastic part comes directly from the optical model while the compound part comes from compound nucleus theory, namely Eq. (13.8). An analogous relation holds for inelastic scattering to a single discrete state  $i$

$$\frac{d\sigma_{n,n'}^i}{d\Omega} = \frac{d\sigma_{n,n'}^{i,direct}}{d\Omega} + \frac{d\sigma_{n,n'}^{i,compound}}{d\Omega}, \quad (4.31)$$



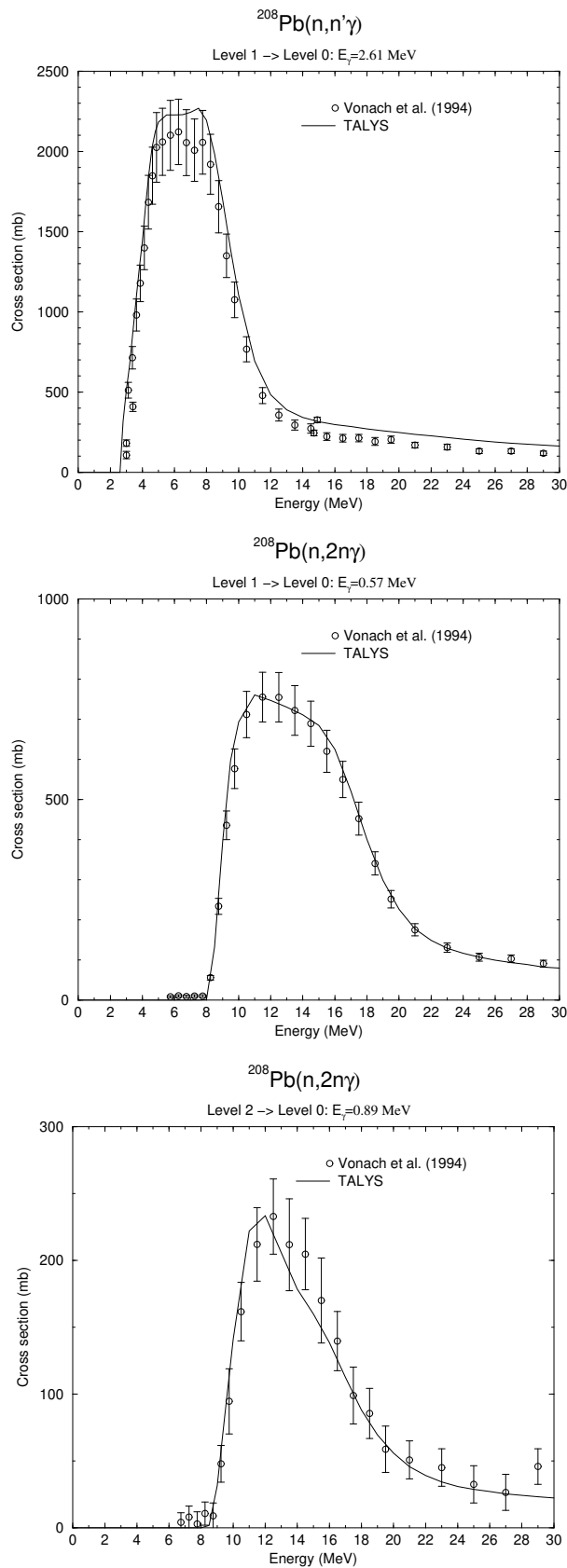


Figure 4.9: Gamma-ray production lines for a few transitions in  $^{208}\text{Pb}(n,n')$  and  $^{208}\text{Pb}(n,2n)$  reactions. The experimental data are from [8].

where the direct component comes from DWBA or coupled-channels calculations. For charge exchange, we can write

$$\frac{d\sigma_{n,p}^i}{d\Omega} = \frac{d\sigma_{n,p}^{i,direct}}{d\Omega} + \frac{d\sigma_{n,p}^{i,compound}}{d\Omega} \quad (4.32)$$

and analogous expressions can be written for the other binary reactions ( $n, d$ ), etc.

Of course, the integration over solid angle of every angular distribution defined here must be equal to the corresponding cross section, e.g.

$$\sigma_{n,n'}^{i,direct} = \int d\Omega \frac{d\sigma_{n,n'}^{i,direct}}{d\Omega}. \quad (4.33)$$

In the output of TALYS, we use a representation in terms of outgoing angle and one in terms of Legendre coefficients, i.e. Eq. (4.30) can also be written as

$$\frac{d\sigma^{el}}{d\Omega} = \sum_L (C_L^{shape-el} + C_L^{comp-el}) P_L(\cos\Theta), \quad (4.34)$$

where  $P_L$  are Legendre polynomials. For inelastic scattering we have

$$\frac{d\sigma_{n,n'}^i}{d\Omega} = \sum_L (C_L^{i,direct} + C_L^{i,comp}) P_L(\cos\Theta), \quad (4.35)$$

and similarly for the other binary channels. The Legendre expansion is required for the storage of the results in ENDF data libraries.

### 4.3.2 Exclusive spectra

An exclusive spectrum is not only specified by the exact number of emitted particles, but also by their outgoing energies.

In TALYS, exclusive spectra are calculated in the same loops that take care of the exclusive cross sections. The inclusive continuum spectra are obtained by taking the derivative of the inclusive cross sections per excitation energy of Eq. (4.10) with respect to the outgoing particle energy  $E_{k'}$ ,

$$E_{k'} = E_x - E_{x'}(i') - S_{k'}, \quad (4.36)$$

where  $S_{k'}$  is the separation energy for outgoing particle  $k'$ . Note that since the inclusive cross section per excitation energy  $S$  depends on  $E_{k'}$  via  $s_k$ , the product rule of differentiation applies to Eq. (4.10). Therefore, the inclusive spectrum per excitation energy for an outgoing particle  $k'$  of a given  $(i_n, i_p, i_d, i_t, i_h, i_\alpha)$  channel is

$$\begin{aligned} \frac{dS}{dE_{k'}}(i_n, i_p, i_d, i_t, i_h, i_\alpha, E_{x'}(i')) &= \sum_{k=\gamma,n,p,d,t,h,\alpha} \sum_i \\ &\left[ \frac{s_k(Z, N, E_x(i), E_{x'}(i'))}{P(Z, N, E_x(i))} \frac{dS}{dE_{k'}}(i_n - \delta_{nk}, i_p - \delta_{pk}, i_d - \delta_{dk}, i_t - \delta_{tk}, i_h - \delta_{hk}, i_\alpha - \delta_{\alpha k}, E_x(i)) \right. \\ &+ \left. \delta_{kk'} \frac{ds_k(Z, N, E_x(i), E_{x'}(i'))}{dE_{k'}} \frac{S(i_n - \delta_{nk}, i_p - \delta_{pk}, i_d - \delta_{dk}, i_t - \delta_{tk}, i_h - \delta_{hk}, i_\alpha - \delta_{\alpha k}, E_x(i))}{P(Z, N, E_x(i))} \right], \end{aligned} \quad (4.37)$$

where, as initial condition, the derivatives of  $s_k(Z_C, N_C, E_x^{max}, E_{x'}(i'))$  are the binary emission spectra. The first term on the right-hand side corresponds to the spectrum of the feeding channel and the

second term denotes the contribution of the last emitted particle. The calculation of Eq. (4.37) can be done simultaneously with the exclusive cross section calculation, i.e. we follow exactly the same recursive procedure. The final exclusive spectrum for outgoing particle  $k'$  is given by

$$\frac{d\sigma^{ex}}{dE_{k'}}(i_n, i_p, i_d, i_t, i_h, i_\alpha) = \sum_{i=0, \text{isomers}} \frac{dS}{dE_{k'}}(i_n, i_p, i_d, i_t, i_h, i_\alpha, E_i), \quad (4.38)$$

The terms on the right hand side are the exclusive spectra per ground state or isomer. The latter naturally result from our method, even though only the total exclusive spectra of the left hand side are of interest.

We stress that for a given  $(i_n, i_p, i_d, i_t, i_h, i_\alpha)$  channel, Eq. (4.37) is calculated for *every* outgoing particle  $k'$  (i.e. n, p, d, t, h and  $\alpha$ ). Hence, e.g. the  $(n, 2np\alpha)$  channel is characterized by only one exclusive cross section,  $\sigma_{n,2np\alpha}$ , but by three spectra, one for outgoing neutrons, protons and alpha's, respectively, whereby all three spectra are constructed from components from the first up to the fourth particle emission (i.e. the  $\alpha$  can have been emitted in each of the four stages). In practice, this means that all spectra have a first order pre-equilibrium component (and for higher energies also higher order pre-equilibrium components), and a compound component from multiple emission. Upon integration over outgoing energy, the exclusive cross sections may be obtained,

$$\sigma^{ex}(i_n, i_p, i_d, i_t, i_h, i_\alpha) = \frac{1}{i_n + i_p + i_d + i_t + i_h + i_\alpha} \sum_{k'=n,p,d,t,h,\alpha} \int dE_{k'} \frac{d\sigma^{ex}}{dE_{k'}}(i_n, i_p, i_d, i_t, i_h, i_\alpha). \quad (4.39)$$

### 4.3.3 Binary spectra

Similar to the cross sections, the exclusive spectra determine various other specific spectra of interest. The exclusive *inelastic spectrum* is a special case of Eq. (4.38)

$$\frac{d\sigma_{n,n'}}{dE_{n'}} = \frac{d\sigma^{ex}}{dE_{n'}}(1, 0, 0, 0, 0, 0). \quad (4.40)$$

Since Eq. (4.37) represents an energy spectrum, it includes by definition only continuum transitions, i.e. it does not include the binary reactions to discrete states. Hence, upon integration, Eq. (4.40) only gives the continuum inelastic cross section of Eq. (4.14):

$$\sigma_{n,n'}^{cont} = \int dE_{n'} \frac{d\sigma_{n,n'}}{dE_{n'}}. \quad (4.41)$$

Similar relations hold for the binary  $(n, p)$ ,  $(n, d)$ ,  $(n, t)$ ,  $(n, h)$  and  $(n, \alpha)$  spectra. The contributions to the binary spectra generally come from pre-equilibrium and continuum compound spectra.

### 4.3.4 Total particle production spectra

Similar to the total particle production cross sections, the *composite* or *total neutron spectrum* can be expressed in terms of exclusive spectra as follows

$$\frac{d\sigma_{n,xn}}{dE_{n'}} = \sum_{i_n=0}^{\infty} \sum_{i_p=0}^{\infty} \sum_{i_d=0}^{\infty} \sum_{i_t=0}^{\infty} \sum_{i_h=0}^{\infty} \sum_{i_\alpha=0}^{\infty} \frac{d\sigma^{ex}}{dE_{n'}}(i_n, i_p, i_d, i_t, i_h, i_\alpha), \quad (4.42)$$

i.e. in the other notation,

$$\frac{d\sigma_{n,xn}}{dE_{n'}} = \frac{d\sigma_{n,n'}}{dE_{n'}} + \frac{d\sigma_{n,2n}}{dE_{n'}} + \frac{d\sigma_{n,np}}{dE_{n'}} + \frac{d\sigma_{n,2np}}{dE_{n'}} + \dots \quad (4.43)$$

Similar relations hold for the  $(n, xp)$ , etc. spectra. Note that, in contrast with Eq. (4.20), the multiplicity is already implicit in the exclusive spectra.

Again, in practice we do not use Eq. (4.42) to calculate the composite spectra but instead add the  $ds_k(Z, N, E_x, E_{x'})/dE_{k'}$  term that appears in Eq. (4.37) to the composite spectra while depleting all nuclides in an inclusive calculation. We do use Eq. (4.42) as a numerical check in the case of a few outgoing particles. Finally, integration of the total neutron spectrum and addition of the binary discrete cross section gives the total particle production cross section

$$\sigma_{n,xn} = \int dE_{n'} \frac{d\sigma_{n,xn}}{dE_{n'}} + \sigma_{n,n'}^{disc}, \quad (4.44)$$

and similarly for the other particles.

#### Sample case 4.6 Inelastic spectra at 20 MeV: Direct + Preeq + GR + Compound

For pre-equilibrium studies, it may be worthwhile to distinguish between the various components of the emission spectrum. This was already mentioned in sample case (1c). As an extra sample case, we compare the calculated  $^{209}\text{Bi}(n,xn)$  spectrum at 20 MeV with experimental data. This is accomplished with the following input file,

```
#
# n-Bi209-spec
#
# General
#
projectile n
element bi
mass 209
energy 20.
#
# Parameters
#
ddxmode 2
filespectrum n
```

The various components of the spectrum, and the total, as present in the file *nspec020.000.tot*, are plotted in Fig. 4.10. ■

### 4.3.5 Double-differential cross sections

The generalization of the exclusive spectra to angular dependent cross sections is done by means of the exclusive double-differential cross sections

$$\frac{d^2\sigma^{ex}}{dE_{k'}d\Omega}(i_n, i_p, i_d, i_t, i_h, i_\alpha), \quad (4.45)$$

which are obtained by either physical models or systematics. Integration over angles yields the exclusive spectrum

$$\frac{d\sigma^{ex}}{dE_{k'}}(i_n, i_p, i_d, i_t, i_h, i_\alpha) = \int d\Omega \frac{d^2\sigma^{ex}}{dE_{k'}d\Omega}(i_n, i_p, i_d, i_t, i_h, i_\alpha). \quad (4.46)$$

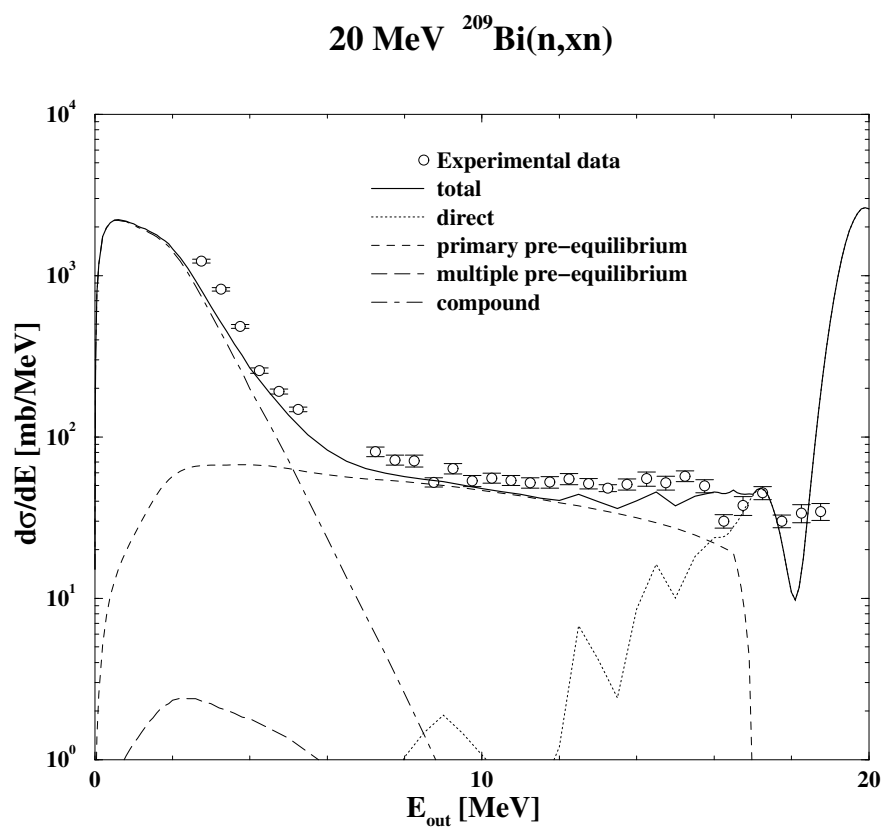


Figure 4.10:  $^{209}\text{Bi}(n,xn)$  spectrum at 20 MeV. Experimental data are obtained from [9].

The other relations are analogous to those of the spectra, e.g. the inelastic double-differential cross section for the continuum is

$$\frac{d^2\sigma_{n,n'}}{dE_{n'}d\Omega} = \frac{d^2\sigma^{ex}}{dE_{n'}d\Omega}(1, 0, 0, 0, 0, 0), \quad (4.47)$$

and the total neutron double-differential cross section can be expressed as

$$\frac{d^2\sigma_{n,xn}}{dE_{n'}d\Omega} = \sum_{i_n=0}^{\infty} \sum_{i_p=0}^{\infty} \sum_{i_d=0}^{\infty} \sum_{i_t=0}^{\infty} \sum_{i_h=0}^{\infty} \sum_{i_\alpha=0}^{\infty} \frac{d^2\sigma^{ex}}{dE_{n'}d\Omega}(i_n, i_p, i_d, i_t, i_h, i_\alpha). \quad (4.48)$$

For the *exclusive* calculation, the angular information is only tracked for the first particle emission. The reason is that for incident energies up to about 20 to 30 MeV, only the first emitted particle deviates from an isotropic angular distribution. Multiple compound emission to the continuum is essentially isotropic. The isotropic contribution to the exclusive double-differential spectrum is then simply determined by the part of the corresponding cross section that comes from Hauser-Feshbach decay. At higher incident energies, where the approximation of only one forward-peaked particle becomes incorrect, the interest in exclusive spectra, or for that matter, the computational check of Eq. (4.48), is no longer there. The presence of multiple pre-equilibrium emission at energies above several tens of MeV requires that we include angular information for every emitted particle in the *total* double-differential cross section, i.e. the left-hand side of Eq. (4.48). Again, this is all tracked correctly in the full inclusive calculation.

#### 4.4 Fission cross sections

For clarity, we have kept the fission channel out of the discussion so far. The generalization to a picture in which fission is possible is however not too difficult. For fissile nuclides, the first expression that needs generalization is that of the non-elastic cross section expressed as a sum of exclusive cross sections, Eq. (4.6). It should read

$$\sigma_{non-el} = \sum_{i_n=0}^{\infty} \sum_{i_p=0}^{\infty} \sum_{i_d=0}^{\infty} \sum_{i_t=0}^{\infty} \sum_{i_h=0}^{\infty} \sum_{i_\alpha=0}^{\infty} \sigma^{ex}(i_n, i_p, i_d, i_t, i_h, i_\alpha) + \sigma_f, \quad (4.49)$$

where the *total fission* cross section  $\sigma_f$  is the sum over exclusive fission cross sections

$$\sigma_f = \sum_{i_n=0}^{\infty} \sum_{i_p=0}^{\infty} \sum_{i_d=0}^{\infty} \sum_{i_t=0}^{\infty} \sum_{i_h=0}^{\infty} \sum_{i_\alpha=0}^{\infty} \sigma_f^{ex}(i_n, i_p, i_d, i_t, i_h, i_\alpha), \quad (4.50)$$

where  $\sigma_f^{ex}(i_n, i_p, i_d, i_t, i_h, i_\alpha)$  represents the cross section for fissioning *after* the emission of  $i_n$  neutrons,  $i_p$  protons, etc. Well-known special cases are  $\sigma_{n,f} = \sigma_f^{ex}(0, 0, 0, 0, 0, 0)$ ,  $\sigma_{n,nf} = \sigma_f^{ex}(1, 0, 0, 0, 0, 0)$  and  $\sigma_{n,2nf} = \sigma_f^{ex}(2, 0, 0, 0, 0, 0)$ , which are also known as first-chance, second-chance and third-chance fission cross section, respectively. Eq. (4.50) is more general in the sense that it also includes cases where particles other than neutrons can be emitted before the residual nucleus fissions, e.g.  $(n, npf)$ , which may occur at higher incident energies.

The generalization of the non-elastic cross section of Eq. (4.19) is

$$\sigma_{non-el} = \sum_{k=\gamma, n', p, d, t, h, \alpha} \sigma_{n,k}^{inc.bin} + \sigma_f^{bin}, \quad (4.51)$$

where  $\sigma_f^{bin}$  represents fission from the initial compound state (i.e. excluding  $(n, \gamma f)$  processes).

Analogous to Eq. (4.25), we can define a cross section for each fissioning residual product

$$\sigma_{prod}^{fis}(Z, N) = \sum_{i_n=0}^{\infty} \sum_{i_p=0}^{\infty} \sum_{i_d=0}^{\infty} \sum_{i_t=0}^{\infty} \sum_{i_h=0}^{\infty} \sum_{i_\alpha=0}^{\infty} \sigma_f^{ex}(i_n, i_p, i_d, i_t, i_h, i_\alpha) \delta_N \delta_Z. \quad (4.52)$$

At higher energies, the meaning of  $\sigma_{prod}^{fis}(Z, N)$  is more relevant than the exclusive fission cross sections. Consequently, for the total fission cross section we have

$$\sigma_f = \sum_Z \sum_N \sigma_{prod}^{fis}(Z, N). \quad (4.53)$$

What remains to be explained is how  $\sigma_f^{ex}$  is computed. First, we need to add to Eq. (4.8) a term we denote by  $s_f(Z, N, E_x(i))$ , which is the part of the population that fissions from the  $(Z, N, E_x(i))$  bin. Hence, for fissile nuclides we have

$$P(Z, N, E_x(i)) = s_f(Z, N, E_x(i)) + \sum_{k=\gamma, n, p, d, t, h, \alpha} \sum_i s_k(Z, N, E_x(i), E_{x'}(i')). \quad (4.54)$$

where in this case the sum over  $i$  runs over discrete levels and continuum bins from 0 to  $E_x(i) - S_k$ . The exclusive fission cross section  $\sigma_f^{ex}$  is

$$\sigma_f^{ex}(i_n, i_p, i_d, i_t, i_h, i_\alpha) = \sum_i \frac{s_f(Z, N, E_x(i))}{P(Z, N, E_x(i))} S(i_n, i_p, i_d, i_t, i_h, i_\alpha, E_x(i)). \quad (4.55)$$

where  $i$  runs from 0 to  $E_x^{max}(Z, N)$ . The rest of the calculation of the exclusive particle cross section proceeds exactly as before. Eq. (4.10) is now automatically depleted from the fission cross section (4.55), in the sense that the  $s_k$  terms alone, summed over  $\gamma$  and particles only, no longer add up to the population  $P$ .

Finally, the exclusive fission cross sections are also accompanied by spectra. For example, the first two neutrons emitted in the  $(n, 2nf)$  channel (third-chance fission) are described by an outgoing neutron spectrum. The exclusive spectrum of outgoing particle  $k'$  in a fission channel is

$$\frac{d\sigma_f^{ex}}{dE_{k'}}(i_n, i_p, i_d, i_t, i_h, i_\alpha) = \sum_i \frac{s_f(Z, N, E_x(i))}{P(Z, N, E_x(i))} \frac{dS}{dE_{k'}}(i_n, i_p, i_d, i_t, i_h, i_\alpha), \quad (4.56)$$

while the exclusive particle spectra are again described by Eq. (4.37). For double-differential spectra, the usual generalization holds. We also repeat here that the total (observable) fission cross section is always calculated by letting reaction population go into the fission channel from each  $(Z, N, E_x, J, \Pi)$  channel until all nuclides have ended up in their ground or isomeric state, irrespective of the user request for an exclusive channel calculation.

#### Sample case 4.7 Fission cross sections: $n + {}^{232}\text{Th}$

It is well known that a systematic approach for fission is difficult to achieve. It is possible to obtain very satisfactory fits to fission data with TALYS, but at the expense of using many adjustable input parameters. We are performing extensive model calculations to bring somewhat more structure in the collection of fitting parameters. In the meantime, we include here a sample case for the description of the fission cross section of  ${}^{232}\text{Th}$ . We use the following input file,

```
#
# n-Th232-fis
#
# General
#
```

```

projectile n
element Th
mass 232
energy energies
partable y
bins 40
channels y
filechannels y
outfission y
best y
ecissave y

```

Note that, although no adjustable parameters are visible above, this is certainly no “default” calculation. Instead, this sample case illustrates the use of so-called “best files”. By using the **best y** keyword, the file *talys/structure/best/Th232/n-Th232.talys* is automatically invoked. The contents of that file are

```

#avadjustF n 0. 2. 10. 1.
#avadjustF n 2. 7. -22. 3.
#avadjustF n 7. 20. 25. 6.
#rvadjustF n 14. 25. 5. 8.
#rvadjustF n 8. 12. 3. 4.
Pshiftadjust 90 232 -1.0
colldamp y
asys y
class2 y
maxrot 5
t          90 233  0.41756  1
e0         90 233 -0.90113  1
exmatch   90 233  5.89461  1
t          90 233  0.56780  2
e0         90 233 -2.52108  2
exmatch   90 233  6.63983  2
fisbar    90 233  6.41136  1
fishw     90 233  0.61550  1
class2width 90 233  0.2      1
fisbar    90 233  5.23377  2
t          90 232  0.46845
e0         90 232 -0.39156
Exmatch   90 232  5.55627  0
a          90 232 31.90038
t          90 232  0.41262  1
e0         90 232 -1.66339  1
exmatch   90 232  6.74958  1
t          90 232  0.44833  2
e0         90 232 -0.51471  2

```



```

exmatch      90 232   6.45816  2
class2width  90 232   0.10013  1
fisbar       90 232   6.31815  2
a            90 231  26.62965
t            90 231   0.45494
e0           90 231  -0.91497
exmatch      90 231   4.92684  0
t            90 231   0.47007  1
e0           90 231  -2.16775  1
exmatch      90 231   5.98780  1
t            90 231   0.68793  2
e0           90 231  -2.79424  2
exmatch      90 231   6.74959  2
a            90 230  26.21683
t            90 230   0.47155
e0           90 230  -0.33802
exmatch      90 230   5.52092  0
t            90 230   0.45039  1
e0           90 230  -1.14073  1
exmatch      90 230   6.80289  1
t            90 230   0.41118  2
e0           90 230  -1.25432  2
exmatch      90 230   6.52706  2
#cwidth 0.06

```

The above parameters are the usual ones to be adjusted to get good agreement with experimental data: level density and fission parameters for the ground state or on top of the barriers. The resulting file *fission.tot* is plotted together with experimental data in Fig. 4.11.

Due to the presence of **outfission y** in the input file, all nuclear structure related to the fission process is given in the output for the target and the compound nucleus

Fission information for Z= 90 N=142 (232Th)

```

Number of fission barriers      : 2
Number of sets of class2 states : 1

```

Parameters for fission barrier 1

```

Type of axially                : 1
Height of fission barrier 1     : 5.800
Width of fission barrier 1      : 0.900
Rtransmom                       : 0.600
Moment of inertia                : 87.657
Number of head band transition states: 8
Start of continuum energy       : 0.800

```

Head band transition states

no.	E	spin	parity
1	0.000	0.0	+
2	0.500	2.0	+
3	0.400	0.0	-
4	0.400	1.0	-
5	0.500	2.0	+
6	0.400	2.0	-
7	0.800	0.0	+
8	0.800	0.0	+

## Rotational bands

no.	E	spin	parity
1	0.000	0.0	+
2	0.034	2.0	+
3	0.114	4.0	+
4	0.240	6.0	+
5	0.400	1.0	-
6	0.400	2.0	-
7	0.411	8.0	+
8	0.411	1.0	-
9	0.423	2.0	-
10	0.434	3.0	-

.....  
Parameters for fission barrier 2

Type of axiality : 2  
 Height of fission barrier 2 : 6.318  
 Width of fission barrier 2 : 0.600  
 Rtransmom : 1.000  
 Moment of inertia : 154.212  
 Number of head band transition states: 4  
 Start of continuum energy : 0.500

## Head band transition states

no.	E	spin	parity
1	0.000	0.0	+
2	0.500	2.0	+
3	0.200	0.0	-
4	0.500	1.0	-

.....

Moreover, the corresponding fission transmission coefficients are printed for all excitation energies encountered in the calculation

Fission transmission coefficients for Z= 90 N=143 (233Th) and an excitation en

J	T(J,-)	T(J,+)
---	--------	--------

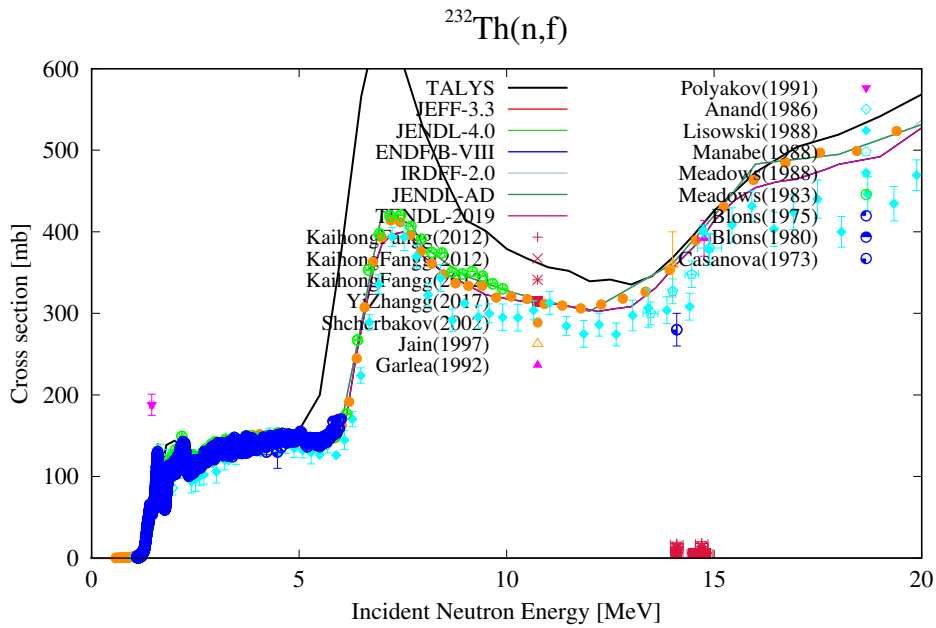


Figure 4.11: Neutron induced fission cross section of  $^{232}\text{Th}$  compared with experimental data.

```

0.5  4.02053E-08  6.51894E-08
1.5  0.00000E+00  0.00000E+00
.....

```

The fission information for all residual nuclides can be obtained in the output file as well by adding *outpopulation y* to the input file.

As an exercise, the user may want to try to adjust the first fission barrier of  $^{232}\text{Th}$  to get a better fit of the second chance fission cross section.

## 4.5 Recoils

### 4.5.1 Qualitative analysis

In a nuclear reaction code, the calculations are usually performed in the center of mass (CM) frame, while the experimental data are obtained in the Laboratory (LAB) frame. It is therefore necessary to perform a transformation either by (i) expressing the experimental data in the CM frame or by (ii) expressing the CM model results in the LAB frame. Of course, the cross sections are the same in both frames, but the spectra are certainly different. The best example is given by the elastic peak in an emission spectrum which is a Dirac delta peak in the CM frame and rather looks like a Gaussian when measured experimentally. The reason for this, apart from the fact that the projectile beam is not perfectly mono-energetic, is that the composite system has a velocity in the LAB frame before decay occurs. Consequently if one considers the emission of an ejectile with a well defined energy in the CM frame, the ejectile energy in the LAB frame will not be unique because of all the CM emission angles. More precisely, a maximum ejectile energy will be obtained when the emission occurs at  $0^\circ$ , and a minimum will be obtained at  $180^\circ$ , together with all the intermediate

situations. Dealing with this situation is simple if only one nucleus decays, but if two particles are sequentially emitted, the first emission probabilities create a velocity distribution of the residual nuclei in the LAB frame. One must first loop over these velocities before one can compute the secondary emission.

### 4.5.2 General method

As mentioned in section 4.2.2, in TALYS each nucleus that can decay is described by an array  $P(Z, N, E_x)$  which gives the population in a bin/level with excitation energy  $E_x$  of the nucleus  $(Z, N)$ . A special case is the initial compound nucleus which contains all the initial reaction population at its total excitation energy  $E_x^{max}$ . For the kinematics of the binary reactions, it is necessary to keep track of the velocities and moving directions of these nuclei in the LAB frame, so that we can reconstruct the LAB spectra from the decays in the CM frame. We therefore have to add in principle three dimensions to the  $P$  array. The first one to keep track of the recoil energy, and the two other ones for the emission angles. However, such book-keeping would become very time consuming, especially for high energies.

Hence, we only take into account the recoil energies and the usual  $\Theta$  angle and define another array  $P_{rec}(Z, N, E_x, E_r, \Theta_r)$  which indicates the fraction of the total population  $P(Z, N, E_x)$  moving with the kinetic energy  $E_r$  in the direction  $\Theta_r$  with respect to the beam direction in the LAB frame. Obviously,

$$P(Z, N, E_x) = \sum_{E_r \text{ bins}} \sum_{\Theta_r \text{ bins}} P_{rec}(Z, N, E_x, E_r, \Theta_r). \quad (4.57)$$

Again, the initial compound nucleus  $(Z_C, N_C)$  is a special one. Its kinetic energy  $E_r^0$  in the LAB frame is unique and is given by

$$E_r^0 = \sqrt{(E_p^2 + 2M_p E_p + M_C^2)} - M_C, \quad (4.58)$$

where  $E_p$  is the projectile kinetic energy in the LAB,  $M_p$  the projectile mass and  $M_C$  the compound nucleus mass, and it moves in the beam direction (i.e.  $0^\circ$ ). Before any emission is calculated, the initial reaction population is stored in the array element  $P_{rec}(Z_c, N_c, E_x^{max}, E_r^0, 0)$ . As explained before, the population of the residual nuclei bins are calculated by looping over all possible ejectiles, emission energies and angles in the CM frame. Therefore, each time we decay from a mother bin to a residual bin, we know exactly what fraction of the total bin population is emitted in a given CM (energy, angle) bin. We then simply couple the CM emission energies and angles with the CM kinetic energy and moving direction in the LAB frame to determine simultaneously the ejectile double-differential spectrum in the LAB and the residual nucleus population in the corresponding LAB (energy, angle) bin. This may seem simple from a qualitative point of view, it is however not trivial to implement numerically and can be time consuming.

### 4.5.3 Quantitative analysis

From now on, for simplicity, we assume that the kinematics of the binary reactions can be considered as a classical process, i.e. we exclude  $\gamma$  decay and relativistic kinematics in the recoil calculation. We here consider the emission of a given ejectile from a given energy bin  $i$  of the decaying nucleus  $(Z, N)$  which moves with a given velocity  $v_{cm}$  (or kinetic energy  $E_{cm}$ ) in the direction  $\Theta_{cm}$  with respect to the beam direction. The total population that is going to decay is  $P(Z, N, E_i)$  and the fraction of this population moving with the velocity  $v_{cm}$  in the direction  $\Theta_{cm}$  is given by

$P_{rec}(Z, N, E_i, E_{cm}, \Theta_{cm})$ . We can determine the total emitted flux for a given emission energy and a given emission angle in the CM frame. In practice, we rather decay from a initial bin to a residual bin in a given angular bin in the CM frame. If recoil effects are neglected we directly derive from such a decay an energy bin  $[E_{low}^{CM}, E_{up}^{CM}]$  and an angular bin  $[\Theta_{low}^{CM}, \Theta_{up}^{CM}]$  in which the total flux  $\Phi_{ej}^{CM}$  is emitted. Accounting for recoil effects requires an intermediate step to share the available energy  $\Delta E$  (difference between the energy bins of the initial nucleus and final nucleus) among the ejectile with mass  $m_{ej}$  and the residual nucleus with mass  $M_R$ .

To do this, we use the classical relation

$$\vec{v}_{ej}^{LAB} = \vec{v}_{cm} + \vec{v}_{ej}^{CM}, \quad (4.59)$$

which connects the LAB velocity  $\vec{v}_{ej}^{LAB}$  of the ejectile with its velocity  $\vec{v}_{ej}^{CM}$  in the CM frame and the CM frame velocity  $\vec{v}_{cm}$ . We need to connect  $\vec{v}_{ej}^{CM}$  with  $\Delta E$ .

This can be done upon writing

$$\Delta E = \frac{1}{2}m_{ej}(\vec{v}_{ej}^{CM})^2 + \frac{1}{2}M_R(\vec{v}_R^{CM})^2, \quad (4.60)$$

where  $\vec{v}_R^{CM}$  is the residual nucleus velocity in the CM frame, and using the relation

$$m_{ej}\vec{v}_{ej}^{CM} + m_R\vec{v}_R^{CM} = \vec{0}. \quad (4.61)$$

Combining (4.60) and (4.61) yields

$$v_{ej}^{CM} = \sqrt{2 \frac{M_R}{m_{ej}(m_{ej} + M_R)} \Delta E}, \quad (4.62)$$

which reduces to the classical relation

$$v_{ej}^{CM} = \sqrt{\frac{2\Delta E}{m_{ej}}}, \quad (4.63)$$

if recoil effects are neglected (i.e. in the limit  $M_R \rightarrow +\infty$ ).

Once this connection is established, Eq. (4.59) is used to determine the velocity and angle of both the emitted light particle and the residual nucleus by simple projections on the LAB axis.

Hence, given a decay situation in the CM frame, we can reconstruct both the energy and angle of emission in the LAB frame. We now have to determine the link between the double-differential decay characteristics in both frames. The solution is well known (see Ref. [10] for instance) and consists of using a Jacobian which accounts for the modification of an elementary solid angle  $d\Omega$  in the CM frame when going into the LAB frame. However, in TALYS we have to employ another method because we do not generally calculate decays for well defined energies and angles but rather for a given energy bin and angular bin. Moreover, since we do not account for the azimuthal angle, we may also encounter some problems when calculating recoil for secondary emission. Indeed, only the first binary process has the azimuthal symmetry with respect to the beam direction.

#### 4.5.4 The recoil treatment in TALYS

The way the double-differential spectra are calculated by TALYS in the LAB frame from those obtained in the CM frame is illustrated in Fig. 4.12. As stressed in Chapter 4, the emission energy grid for the outgoing particles is non-equidistant. Moreover, one has to keep in mind that the total

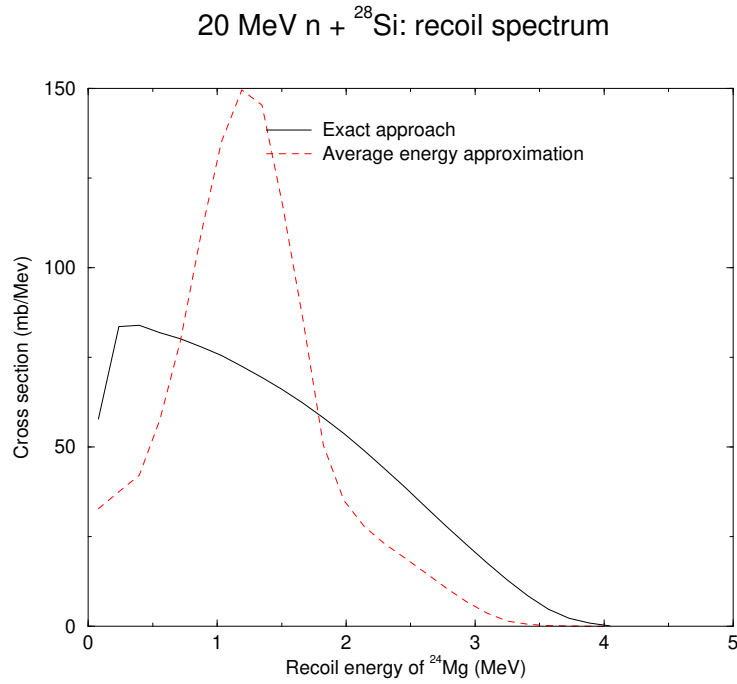


Figure 4.12: CM and LAB double-differential spectra in TALYS.

flux  $\Phi(i, j)$  in an energy-angular bin  $(i, j)$  is connected with the double differential cross section  $xs(i, j)$  by

$$\Phi(i, j) = xs(i, j)\Delta E(i)\Delta\cos\Theta(j). \quad (4.64)$$

Consequently, it is appropriate to locate the grid points using an energy-cosine grid. As an example, in Fig. 4.12, we consider the decay in the CM bin defined by the energy interval  $[2, 3]$  and the cosine interval  $[0.25, 0.5]$  (black region). We assume that the decay occurs from a composite system moving with a kinetic energy of 1 MeV in the direction  $45^\circ$  with respect to the initial beam direction. The mass of the ejectile is assumed to be  $m_e = 1$  arb.unit, and that of the composite system 20 arb.unit. In that case, the region reached by the ejectile is a slightly deformed trapezoid (gray region) which covers several bins. Therefore, if the emitted flux is located in a single bin in the CM, it must be distributed over several bins when going to the LAB. This is the key problem to be solved in TALYS. To solve this problem, we need to make assumptions to be able (i) to calculate the area covered in the LAB frame (gray region) and (ii) the way this global area is distributed over the bins it partially covers. The assumption made in TALYS consists of neglecting the deformation of the boundaries of the grey area and assuming this area to be a trapezoid. In other words, we make the assumption that a triangle in the CM frame is transformed into a triangle when going in the LAB. This is helpful since the area of a triangle is given by a simple analytic expression as function of the coordinates of the summits of the triangle. Therefore, we divide the starting CM bins into two triangles to determine the two triangles obtained in the LAB frame. With such a method, the whole problem can be solved and the decay calculated in the CM frame can be transformed to the LAB frame without any further approximations.

However, in practice, coupling the angular direction (in the LAB) of the nucleus that decays with the ejectile emission angle in the CM frame, while neglecting the azimuthal angle, gives double differential ejectile spectra in the LAB which are generally not correct. In fact, we believe

that it is better not to account for the angular distribution of the decaying nucleus unless both  $\Theta$  and  $\phi$  are explicitly treated. Fortunately, the final angular distribution of the recoiling nucleus is seldom of interest.

#### 4.5.5 Method of average velocity

As mentioned above, we do not loop over the angular distribution of the decaying nucleus. This is equivalent to replacing the array  $P_{rec}(Z, N, E_i, E_r, \Theta_r)$  by  $P_{rec}(Z, N, E_i, E_r)$ . Then, we only have to keep track of the velocities of the nucleus that is going to decay, i.e. we have to loop over the  $E_r$  bins to reconstruct both the ejectile and residual nuclei spectra. Another approximation that we have implemented as an option with **recoilaverage y** (p. 313), consists of using an average velocity before this reconstruction, a method first applied by Chadwick et al. [11, 12]. This approach avoids the loop over the  $E_r$  bins altogether and reduces the calculation time. However, for high energies, this might be too crude an approximation.

#### 4.5.6 Approximative recoil correction for binary ejectile spectra

Maybe you are not interested in a full recoil calculation, but merely want to correct the outgoing particle spectra for the recoil of the nucleus. In that case you may use the following method which is implemented in TALYS for just that purpose and which is outlined below.

The assumptions are made that (i) only binary emission takes place, and that (ii) emission only occurs under  $0^\circ$ . Hence, this approximation is basically expected to be valid for angle-integrated spectra only. The CM to LAB conversion of the ejectile spectra takes under these conditions the following simple form:

$$E_{ej}^{lab} = \frac{M_R}{M_C} \Delta E + \frac{m_{ej} M_p}{M_C^2} E_p + 2 \sqrt{\frac{m_{ej} M_R M_p}{M_C^3} E_p \Delta E}, \quad (4.65)$$

in which  $E_{ej}^{lab}$  is the LAB ejectile energy. This correction is applied to the *full* ejectile spectrum including the multiple emission contributions. Hence, the approximation is rather crude. It saves, however, a lot of computer time. Since the high-energy tail originates completely from binary emission, this tail is correctly converted to the LAB system. Furthermore, the correction is small at low energies where the largest contributions from multiple emission reside.

#### Sample case 4.8 Recoils: 20 MeV n + $^{28}\text{Si}$

In this sample case, we calculate the recoils of the residual nuclides produced by 20 MeV neutrons incident on  $^{28}\text{Si}$  reaction. Two methods are compared.

##### Case a: "Exact" approach

In the exact approach, each excitation energy bin of the population of each residual nucleus is described by a full distribution of kinetic recoil energies. The following input file is used

```
#
# n-Si028-recoil-exact
#
# General
#
projectile n
element si
```

```

mass 28
energy 20.
#
# Parameters
#
m2constant 0.70
sysreaction p d t h a
spherical y
#
# Output
#
recoil y
filerecoil y

#
# General
#
projectile n
element si
mass 28
energy 20.
#
# Parameters
#
m2constant 0.70
sysreaction p d t h a
spherical y
#
# Output
#
recoil y
filerecoil y

```

For increasing incident energies, this calculation becomes quickly time-expensive. The recoil calculation yields separate files with the recoil spectrum per residual nucleus, starting with *rec*, followed by the Z, A and incident energy, e.g. *rec012024spec020.000.tot*. Also following additional output block is printed:

#### 8. Recoil spectra

Recoil Spectrum for 29Si

Energy	Cross section
--------	---------------

0.018	0.00000E+00
-------	-------------

0.053	0.00000E+00
-------	-------------

0.088	0.00000E+00
-------	-------------

.....



```
0.676 3.21368E+00
0.720 3.22112E+00
0.763 3.21215E+00
0.807 3.10619E+00
0.851 2.60156E+00
0.894 2.30919E-01
```

```
Integrated recoil spectrum      : 1.04358E+00
Residual production cross section: 9.71110E-01
```

**Case b: Approximative approach**

As an approximation, each excitation energy bin of the population of each residual nucleus is described by a an average kinetic recoil energy. For this, we add one line to the input file above,

```
recoilaverage y
```

The results, together with those of case (a), are compared in Fig. 4.12. ■



## 5. Nuclear structure and model parameters

### 5.1 General setup of the database

We have aimed to unify the nuclear structure and model parameter database of TALYS as much as possible. In the *talys/structure/* directory you can find the *masses/*, *abundance/*, *levels/*, *fission/*, *resonances/*, *deformation/*, *optical/*, *thermal/*, *gamma/*, *density/*, *decay/* *integral/* and *best/* subdirectories. Most subdirectories contain about 100 files, where each individual file points to a nuclear element and has the name *zZZZ* where *ZZZ* is the charge number of the element. One file contains the info for all the isotopes of one element. For example, the file *talys/structure/levels/final/Fe.lev* contains the discrete levels of all Fe ( $Z=26$ ) isotopes. As you will see below, the chemical symbol is always present in the file itself, allowing an easy search. Every directory has this same substructure and also the formats in which the data are stored have been kept uniform as much as possible.

The nuclear structure database has been created from a collection of “raw” data files, which for a large part come from the Reference Input Parameter Library RIPL [6], that we used as a basis for TALYS.

### 5.2 Integral validation library

Since all excitation functions are available in TALYS after a run with many incident energies, it is relatively easy to calculate production or reaction rates for various applications. This is for example done for astrophysics and medical isotope production. Another possibility is to calculate the so called effective cross section for integral activation measurements, by folding the excitation functions by an experimental flux. In *talys/structure/flux*, we have stored more than 40 spectra, coming from the EASY package [13, 14], which have been used in past activation benchmarks. Also the effective cross sections are stored, allowing direct comparison with experimental data.

### 5.3 Decay data library

For medical isotope production, the decay paths to the produced isotopes need to be known. Therefore, in *talys/structure/decay* we have stored the JEFF-3.1.1 Radioactive Decay Data File in the usual element by element files.

### 5.4 Best TALYS input parameters

We have created the possibility to store the “best” set of model input parameters per nuclide. This is a helpful feature to ensure reproducibility of earlier obtained results, i.e. a nuclear reaction analysis with TALYS is stored in the best possible way (rather than scattered over input files in several working directories). Theoretically, a complete collection of best TALYS input parameter files would enable a high-quality description of nuclear reactions on all nuclides simultaneously (when used e.g. in a script to produce a complete nuclear data library), especially if the work is divided among more than a few persons. One would thus be able to get the best possible results with a seemingly default input file. The only requirement is that the keyword **best y** (p. 294) is set in the input file. All adjusted optical model, level density etc. parameters are stored in the *talys/structure/best* directory. As subdirectories, full isotope names should be used in the format (a1,i3.3) or (a2,i3.3), depending on the element symbol. Furthermore, the first character of the Symbol should be in upper case. In other words, valid examples of subdirectories are *Yb174/*, *F019/*, *Be009/*, and *U235*. For the sets of best parameters in these subdirectories, we use filenames *zZZZaAAAS.best* where ZZZ and AAA are the Z and A of the nuclide in i3.3 format, and S is the particle symbol (g, n, p, d, t, h, or a). This strict naming procedure is required for software that uses TALYS for nuclear data evaluation purposes. As an example, we show the best parameters for neutrons incident on  $^{80}\text{Se}$  as present in the file *talys/structure/best/Se080/n-Se080.talys*,

```
#
# General
#
ldmodel 2
m2constant 0.9
#
# (n,tot), (n,e1), (n,in1)
#
rvadjust n 1. 0.01 1. 0.5 0.99
rvadjust n 1. 10. 5.5 1.02
#
# (n,p), (n,2n), (n,np)
#
rvadjust p 1.04
avadjust p 1.04
gnadjust 34 81 0.90
gpadjust 34 81 0.90
tadjust 34 80 1.15
#
# (n,a)
#
rvadjust a 1.05
avadjust a 1.05
Cstrip a 1.00
Cknock a 1.00
aadjust      32 77 1.10
#
```

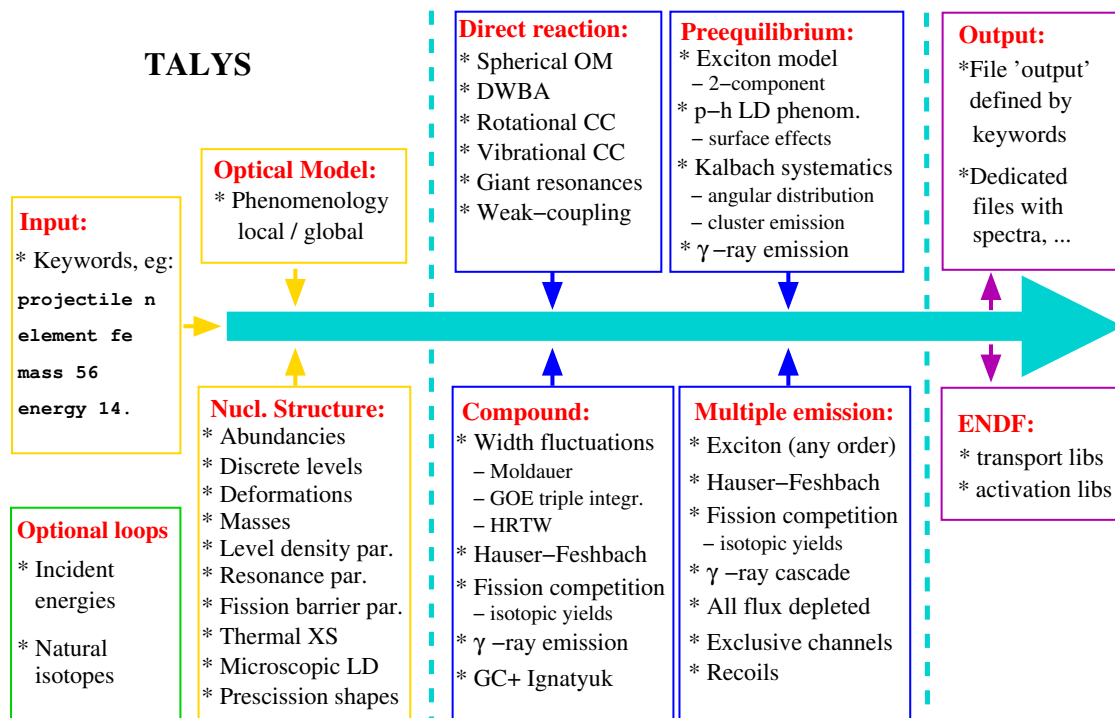


Figure 5.1: Nuclear models in TALYS

```
# (n,g)
ldmodelCN 1
#
#
# Other: Isomers, (n,d), (n,t), (n,h) etc.
#
```

If we use **best y**, the above set of keywords will automatically be added to the TALYS input file. This directory is also the best place to store other TALYS input files such as files with tabular optical model parameters. They need to be copied to the working directory when the **best y** option is used. It is also possible to create alternative “best” database under *talys/structure* with a different collection of best input files. These can then be invoked with **bestpath** (p. 324).

Fig. 5.1 gives an overview of the nuclear models that are included in TALYS. They can generally be categorized into optical, direct, pre-equilibrium, compound and fission models, all driven by a comprehensive database of nuclear structure and model parameters. We will first describe the optical model and the various models for direct reactions that are used. Next, we give an outline of the various pre-equilibrium models that are included. Then, we describe compound nucleus models for both binary and multiple emission, and level densities, which are important ingredients of pre-equilibrium and compound models. Finally, we give an outline of the fission models.





# Part Two: Nuclear Models

6	Nuclear masses and deformations ..	107
7	Discrete levels .....	111
8	Level densities .....	117
9	Reactions involving photons .....	139
10	Optical model .....	149
11	Direct reactions .....	169
12	Pre-equilibrium reactions .....	179
13	Compound reactions .....	205
14	Multiple emission .....	217
15	Fission .....	221
16	Thermal reactions .....	247
17	Populated initial nucleus .....	253
18	Astrophysical reaction rates .....	255
19	Medical isotope production .....	261
20	Verification and validation, sample cases and output .....	269





In this part of the tutorial, all nuclear models which are implemented in TALYS are described. This includes nuclear structure databases, which contains properties such as masses, discrete levels, level densities, fission barriers, etc., to nuclear reaction models such as the optical model and models for direct, compound, pre-equilibrium and fission reactions.

The set up in each Chapter follows the same consistent structure:

- **Outline of the model.** The nuclear models are explained to a reasonable depth. In general we restrict ourselves to the physical aspects and equations which are actually implemented in TALYS. We try to give as many relevant references as possible for more detailed information.
- **Options.** All TALYS keywords relevant to the Chapter under consideration are briefly mentioned. A full description of the keywords is given in Part III, the reference guide.
- **Implementation.** We give a short description on the implementation in terms of databases and also the subroutines which take care of the modelling.
- **Sample cases.** We aim to give you practical advice on how to use TALYS while we explain the nuclear models. That means that sample cases are scattered throughout this Part, when they matter most.



## 6. Nuclear masses and deformations

### 6.1 Nuclear masses

On the level of nuclear physics that is relevant to TALYS, the mass can be considered as the most basic property of the nucleus, and directly affects all the outcomes of a nuclear reaction calculation. For a start, separation energies and  $Q$ -values are directly dependent on the masses of the involved nuclides, and determine whether certain reactions can take place or not. The nuclear community has invested a lot of effort to compile experimental nuclear masses in data tables, and simultaneously theoretical nuclear structure models have been developed to predict masses and other properties of the nucleus. TALYS uses various different sources, and thus options, for nuclear masses, in order of priority:

- Experimental masses:
  - The Atomic Mass Evaluation, AME2020 [15], the successor of the well-known Audi-Wapstra table. Obviously, around the valley of stability we need to make use of experimentally determined masses and can not afford to deviate from this, especially if we want to reproduce experimental reaction data and be credible for technological applications. AME2020 is the default choice for our mass table.
- Theoretical masses are needed outside the range where measurements have been performed and sometimes exhibit an interesting dispersion, especially when applied in nuclear reaction models. The options we have included are:
  - The latest version of the Skyrme-Hartree-Fock-Bogoliubov nuclear mass model by Goriely and Pearson [16],
  - The Gogny-Hartree-Fock-Bogoliubov nuclear mass model by Goriely et al. [17],
  - The Finite-Range Droplet Macroscopic (FRDM) model by Möller et al. [18],
- The Duflou-Zuker mass formula [19], included as a subroutine in TALYS. This is used if a nuclide is even outside the mass tables, or simply as an interesting alternative,

and as you will read below you can also add your own mass tables.

In TALYS, we use both the real mass  $M$  in atomic mass units ( $amu = 931.49386$  MeV) and the mass excess  $\Delta M = (M - A) * amu$  in MeV from the mass tables. The latter are retrieved for a more precise calculation of separation energies. For example, the neutron separation energy of a nucleus  $(Z,N)$  is

$$S_n(Z,N) = M(Z,N-1) - M(Z,N) + M(n) \quad (6.1)$$

where  $\Delta M(n)$  is the neutron mass excess and similarly for the other light particles, but we have actually implemented

$$S_n(Z,N) = \Delta M(Z,N-1) - \Delta M(Z,N) + \Delta M(n) \quad (6.2)$$

For reaction Q-values one needs the masses of two nuclides. The Q-value is the difference between two separation energies,

$$Q(a,b) = S_a(Z,N) - S_b(Z,N) \quad (6.3)$$

If for only one of the two masses the experimental mass excess is known, we take the two *theoretical* mass excesses to calculate the Q-value, for consistency. This only occurs for nuclides far from the valley of stability.

**Options 6.1.1** Enough flexibility has been built in for varying the nuclear masses. First of all, **massmodel** (p. 330) can be used to change the theoretical mass model, **expmass** (p. 331) to disable the experimental mass table altogether (i.e. discard AME2020), while the keywords **massnucleus** (p. 334) and **massexcess** (p. 335) can be used to directly change individual masses in the user input.

It is possible that we find future mass models so important that we add them to the database of the official TALYS version. Until that time, you can add your own mass table by adding a subdirectory to *talys/structure/masses/* and we will extend the range of values for **massmodel**. It is then your responsibility that the files have exactly the same structure as those of the standard four mass models. Hence, if you create e.g. a new mass table called *talys/structure/masses/moller2020/* with the correct files and file structure inside (i.e. that of **massmodel 1, 2** or **3**, you can invoke that in the input with **massdir moller2020** (p. 333).

**Implementation 6.1.1** All the mass models have been processed into our database and are stored in the *talys/structure/masses/* directory. We have provided 4 subdirectories, *audi/* for AME2020, *hfb/* for Skyrme-Hartree-Fock-Bogoliubov, *hfbdlm/* for Gogny-Hartree-Fock-Bogoliubov and *frdm/* for the FRDM model.

As an example, below are masses of Fe-isotopes as given in file *masses/audi/Fe.mass*. We have stored  $Z$ ,  $A$ , mass ( $amu$ ), mass excess (MeV), nuclear symbol with the format (2i4,2f12.6,42x,i4,a2).

26	45	45.014419	13.431000	45Fe
26	46	46.000630	0.587000	46Fe
26	47	46.991850	-7.592000	47Fe
26	48	47.980230	-18.416000	48Fe
26	49	48.973429	-24.750727	49Fe
26	50	49.962974	-34.488631	50Fe

All nuclear masses are read in *masses.f90*. First, it is checked whether an experimental value

for the mass exists. If not, it reads the masses from one of the theoretical mass tables or resorts to the Duflo-Zuker formula, which is programmed in *duflo.f90*. Once all masses have been read, *separation.f* calculates the separation energies for all nuclides and particles of interest. Next, the program returns to *nuclides.f90* where the Q-values are calculated.

Fundamental constants such as amu, masses and mass excesses of light particles, e.g.  $\Delta M(n)$ , are all stored in *constants.f90*.

### Sample case 6.1 Mass models: $n + {}^{132}\text{Sn}$ .

This sample case illustrates the use of different mass models. We calculate neutron-induced reactions on  ${}^{132}\text{Sn}$  as a function of incident energy for the default mass model (AME2020), which for this reaction provides experimental masses, the 3 different theoretical mass tables, and the Duflo-Zuker formula. The following input file is used for the default run,

```
#
# General
#
projectile n
element Sn
mass 132
energy 8. 20. 1.

    while for the other cases we use

#
# General
#
projectile n
element Sn
mass 132
energy 8. 20. 1.
expmass n
massmodel 0
```

where **massmodel 0** activates the Duflo-Zuker formula, and another 3 runs with only the last line changed into **massmodel 1, 2 and 3**, respectively.

Fig. 9.1 displays the different (n,p) cross sections, as obtained in files *rp049132.tot*. ■

## 6.2 Deformation parameters

Deformation parameters are used at various places in TALYS. First of all, as input to OMP calculations for deformed nuclides, by means of the coupled-channels formalism, which require values for (at least)  $\beta_2$  and  $\beta_4$  for rotational nuclides. Other applications are the estimation of moments of inertia.

**Options 6.2.1** The **beta2** (p. 332) is available to manually change the deformation parameter  $\beta_2$ .

**Implementation 6.2.1** For the 3 directories with theoretical masses, the ground state deformation parameters  $\beta_2$  and  $\beta_4$  are given in the data files. The parameters in the *hfb/* directory are

used by default for deformed nuclides. Moreover, the ground state spin and parity (for *hfb/* and *hfbdlm/*, not for *frdm/*) as estimated by the HFB model are given. As an example, below are the masses of some of the Fe-isotopes as given in file *masses/hfb/Fe.mass*. We have stored *Z*, *A*, mass (*amu*), mass excess (MeV),  $\beta_2$ ,  $\beta_4$ , ground state spin and parity, nuclear symbol with the format (2i4,2f12.6,2f8.4,20x,f4.1,i2,i4,a2).

26	42	42.055503	51.701000	0.3200	0.0400	0.0	1	42Fe
26	43	43.041632	38.780000	0.2700	0.0300	0.5	1	43Fe
26	44	44.026225	24.428000	-0.2100	0.0400	0.0	1	44Fe
26	45	45.014192	13.220000	-0.1800	0.0300	1.5	1	45Fe
26	46	45.999931	-0.064000	-0.1100	0.0200	0.0	1	46Fe
26	47	46.992187	-7.278000	-0.0900	0.0200	3.5	-1	47Fe
26	48	47.979358	-19.228000	0.1800	0.0700	0.0	1	48Fe

### 6.3 Isotopic abundances

We have included the possibility to evaluate nuclear reactions for natural elements. The abundances are taken from RIPL[6] (which are equal to those of the Nuclear Wallet Cards from Brookhaven National Laboratory). If **mass 0** (p. 280), a TALYS calculation is performed for each isotope, after which the results are averaged with the isotopic abundance as weight.

**Options 6.3.1** By default, the abundances are taken from the tables mentioned above. However, it may be useful to do TALYS calculations for enriched targets. For that purpose, the **abundance** (p. 320) keyword is available to create the isotopic target composition of your own choice.

**Implementation 6.3.1** In *abundance.f90* the abundances are read. Note that for natural targets, there is a loop over all core subroutines of TALYS, in *talysloops.f90*, for each isotope, after which all results are properly averaged with the abundances as weight in *natural.f90*.

The isotopic abundances are stored in the *talys/structure/abundance/* directory. As an example, below are the isotopic abundances for Fe from the file *abundance/Fe.abun*. For each isotope, we have stored *Z*, *A*, its abundance, its uncertainty (not used in TALYS), nuclear symbol with the format (2i4,f11.6,f10.6,45x,i4,a2).

26	54	5.845000	0.035000					54Fe
26	56	91.754000	0.036000					56Fe
26	57	2.119000	0.010000					57Fe
26	58	0.282000	0.004000					58Fe

## 7. Discrete levels

### 7.1 Discrete level file

Besides global properties like the mass and deformation of the nucleus, the discrete level scheme is essential for nuclear reaction calculations, as it is important for both direct and compound nuclear reactions. For the current capabilities of TALYS, we require for each level the

- energy
- spin
- parity
- half life, especially when large (isomers)
- number of branchings for gamma decay
- branching ratios
- electron conversion coefficients

Ideally, this information would all be available from measurements, and a good example of a complete experimental discrete level compilation can be found in the ENSDF and NUBASE databases. These are the sources of the RIPL database which aims to give a more complete compilation of discrete level information. Unfortunately, only measured discrete level data is not enough for TALYS. Each level needs to have all properties in the above list, so that Hauser-Feshbach decay and gamma-ray cascade can be properly calculated. For example, omission of branching ratios for a certain level, because they have not been measured, would mean that the level can not decay and necessarily, and falsely, be regarded a long-lived isomer.

Whenever information was missing in RIPL, it has been added to the TALYS discrete level files with educational guesses. For nuclides far away from the valley of stability, the discrete level database has been completed with the HFB ground state estimates for spin and parity from the mass database, so that it spans the same range of nuclides.

Each excited level should always have a spin and parity. If a spin is not given by the RIPL

database for a certain level, a histogram is made of all spins up to that level. Next, that histogram is compared with a Wigner spin distribution and the level gets the spin of the bin in that distribution which most underestimates the Wigner distribution. For missing parity, we insert the parity which leads to a more equal parity distribution for all levels up to that level.

Often, for a level the branching ratios for gamma decay to other levels are missing. In that case, we take the spin of the level as basis and let it decay to two levels, which have a spin close to the mother level, with 50% branching ratio to each level.

**Options 7.1.1** There are three versions of the discrete level database, and the **disctable** (p. 342) keyword allows to choose between them. The default choice is the RIPL-3 database, but with using theoretical level densities to extend any discrete level sequence to at least 100 levels. Moreover, unknown spins, parities and branching ratios are always assigned a value, based on the simple statistical rules mentioned above. We know discrete levels exist, measured or not, and including them should at least give a more realistic prediction of the discrete lines in a gamma-ray spectrum. Another options is the "pure" RIPL database, based on measurements only, which means that the last known discrete level may have a rather low sequence number, and in Hauser-Feshbach calculations is then directly followed by the continuum described by the level density. The third option is a purely theoretical database, i.e all discrete levels come from level density estimates, which may be used for interesting studies on the predictive power of Hauser-Feshbach calculations. With **branch** (p. 344) the branching ratios of the discrete level database can be overruled.

**Implementation 7.1.1** The discrete level schemes are in the *talys/structure/levels/* directory. We have transformed it to a format that corresponds with that of the other parameter files in our database. Also, we filled in some omissions, repaired several errors and added some information necessary for Hauser-Feshbach calculations as explained above. For consistency, we have included discrete levels for all nuclides that are present in the (theoretical) mass database.

As an example, below are the first discrete levels of  $^{93}\text{Nb}$  as given in file *final/Nb.lev*. We have stored *Z*, *A*, total number of lines for this isotope, number of levels for this isotope, nuclear symbol with the format (2i4,2i5,56x,i4,a2). There is a loop over the levels in which we first read the level number, level energy in MeV, spin, parity, number of gamma-ray branchings, lifetime, spin assignment character, parity assignment character, original ENSDF string with the format (i4,f11.6,f6.1,3x,i2,i3,19x,e9.3,1x,2a1,a18). If the spin of a level does not come from the original file but instead is assigned in RIPL or by us, we place a 'J' in column 59. Analogously, an assigned parity is denoted by a 'P' in column 60. Note also that we have retained the original ENSDF string in columns 61-78. This string indicates the possible choices for spins and parities. If TALYS produces odd results for a nuclear reaction, a possible cause could be the wrong assignment of spin or parity. Columns 59 and 60 and the ENSDF string will reveal whether there are other possibilities that can be (and maybe should have been) adopted. For each level, there may be an inner loop over the number of gamma-ray branchings, and for each branching we read the number of the level to which the gamma-ray decay takes place and the corresponding branching ratio and electron-conversion factor. If a branching ratio has been assigned by us, a 'B' is placed in column 58. The format for this line is (29x,i3,f10.6,e10.3,5x,a1).

41	93	394	126									93Nb
0		0.000000	4.5	1	0							9/2+
1		0.030770	0.5	-1	1				5.090E+08			1/2-



					0	1.000000	1.690E+05		
2	0.686790	1.5	-1	1			2.800E-13		3/2-
					1	1.000000	1.870E-03		
3	0.743950	3.5	1	1			5.100E-13		7/2+
					0	1.000000	1.410E-03		
4	0.808820	2.5	1	2			6.160E-12		5/2+
					3	0.021580	7.670E-01		
					0	0.978420	1.200E-03		
5	0.810320	2.5	-1	2			1.000E-12		5/2-
					2	0.000000	1.273E-01		
					1	1.000000	1.310E-03		
6	0.949800	6.5	1	1			4.360E-12		13/2+
					0	1.000000	8.120E-04		
7	0.970000	1.5	-1	2				J	1/2-, 3/2-
					4	0.500000	0.000E+00	B	
					2	0.500000	0.000E+00	B	
8	0.978910	5.5	1	1			2.580E-13		11/2+
					0	1.000000	7.690E-04		
9	1.082680	4.5	1	3			2.800E-12		9/2+
					8	0.073863	2.044E-01		
					3	0.687527	9.110E-03		
					0	0.238610	6.080E-04		
10	1.127090	1.5	1	1				J	3/2, 5/2, 7/2
					4	1.000000	1.050E-02		

For nuclides for which no experimental information was available, the string 'HFB' has been added as a comment to denote that the information comes from structure calculations.

There is a directory *levels/spectn* with spectroscopic factors for the direct capture process.

## 7.2 Deformation parameters

**Options 7.2.1** The complexity of the coupling scheme can be steered by the **maxrot** (p. 525) for the number of included levels from the rotational band and **maxband** (p. 526) for the number of included vibrational levels.

**Implementation 7.2.1** Deformation parameters, lengths and coupling schemes for coupled channels calculations are stored in directory *talys/structure/deformation/*. They are strongly linked to the discrete level file. For each isotope, we first write *Z, A*, number of levels, type of collectivity, the type of parameter, nuclear symbol with the format (3i4,3x,a1,3x,a1,54x,i4,a2). The type of parameter can be D (deformation length  $\delta_L$ ) or B (deformation parameter  $\beta_L$ ). The type of collectivity can be S (spherical), V (vibrational), R (rotational) and A (asymmetric rotational). Next, we read for each level the number of the level corresponding to that of the discrete level database, the type of collectivity per level, the number of the vibrational band, multipolarity, magnetic quantum number, phonon number of the level, deformation parameter(s) with the format (i4,3x,a1,4i4,4f9.4). The type of collectivity per level can be either D (DWBA), V (vibrational) or R (rotational). In the case of levels that belong to a rotational band, only the level number and an 'R' are given. For the first level of the rotational band, the deformation parameter  $\beta_2$  (and, if present,  $\beta_4$  and  $\beta_6$ ) can be given. If these  $\beta$  parameters are not given they are retrieved from the *talys/structure/mass/* directory. In certain cases, the deformation parameters have been

adjusted to fit data. In those cases, they are added to the coupling schemes. Also, for the first level of a vibrational band the deformation parameter is given. The vibration-rotational model is thus invoked if within the rotational model also states belonging to a vibrational band can be specified. The level of complexity of rotational or vibrational-rotational calculations can be specified with the **maxrot** (p. 525) and **maxband** (p. 526) keywords. For weakly coupled levels that can be treated with DWBA, the level number, a 'D' and the deformation parameter is given.

As an example, below are the deformation parameters for some even Ca isotopes, taken from *Fe.def*. This file ensures that for a reaction on  $^{40}\text{Ca}$ , a coupled-channels calculation with a vibrational model will automatically be invoked. Levels 2 ( $3^-$  at 3.737 MeV, with  $\delta_3 = 1.34$ ), 3 ( $2^+$  at 3.904 MeV, with  $\delta_2 = 0.36$ ), 4 ( $5^-$  at 4.491 MeV, with  $\delta_5 = 0.93$ ), will all be coupled individually as one-phonon states. There is an option to enforce a spherical OMP calculation through **spherical y** (p. 518) in the input, In that case all levels will be treated with DWBA. The table below reveals that for the other Ca-isotopes the direct calculation will always be done with DWBA, but with deformation parameters  $\beta_L$  instead of deformation lengths  $\delta_L$ .

20	40	4	V	D					40Ca
0	V	0							
2	V	1	3		1	1.34000			
3	V	2	2		1	0.36000			
4	V	3	5		1	0.93000			
20	42	3	S	B					42Ca
0	V	0							
1	D					0.24700			
9	D					0.30264			
20	44	3	S	B					44Ca
0	V	0							
1	D					0.25300			
8	D					0.24012			
20	46	3	S	B					46Ca
0	V	0							
1	D					0.15300			
6	D					0.20408			
20	48	3	S	B					48Ca
0	V	0							
1	D					0.10600			
4	D					0.23003			

As a second example, below is the info for  $^{238}\text{U}$  from the *U.def* file. The basis for the coupling scheme is a rotational model with deformation lengths  $\delta_2 = 1.546$  and  $\delta_4 = 0.445$ , in which levels 1, 2, 3, 4, 7 and 22 can be coupled. In practice, we would include at least levels 1 and 2 (and if the results are important enough also levels 3 and 4) as rotational levels. There are 5 vibrational bands which can be included. By default we include no vibrational bands in a rotational model, but if e.g. **maxband 1** in the input, the levels 5, 6, 8 and 13 would be included with deformation length  $\delta_3 = 0.9$ .

92	238	23	R	D					238U
0	R	0				1.54606	0.44508		
1	R	0							
2	R	0							

```
3 R 0
4 R 0
5 V 1 3 0 0.90000
6 V 1
7 R 0
8 V 1
9 V 2 4 0 0.20000
10 V 3 3 1 0.10000
11 V 3
12 V 2
13 V 1
14 V 4 2 0 0.10000
15 V 3
16 V 4
17 V 2
21 V 5 2 2 0.10000
22 R 0
23 V 5
25 V 4
31 V 5
```

Finally, we note that in principle all level numbers in the *deformation/* database need to be re-checked if a new discrete level database (see the previous Section) is installed. It is however unlikely that new low-lying levels will be discovered for nuclides for which the coupling scheme is given.



## 8. Level densities

In statistical models for predicting cross sections, nuclear level densities are used at excitation energies where discrete level information is not available or incomplete. We use several models for the level density in TALYS, which range from phenomenological analytical expressions to tabulated level densities derived from microscopic models. The complete details for phenomenological models can be found in Koning et al. [20].

To set the notation, we first give some general definitions. The *level density*  $\rho(E_x, J, \Pi)$  corresponds to the number of nuclear levels per MeV around an excitation energy  $E_x$ , for a certain spin  $J$  and parity  $\Pi$ . The *total level density*  $\rho^{\text{tot}}(E_x)$  corresponds to the total number of levels per MeV around  $E_x$ , and is obtained by summing the level density over spin and parity:

$$\rho^{\text{tot}}(E_x) = \sum_J \sum_{\Pi} \rho(E_x, J, \Pi). \quad (8.1)$$

The nuclear levels are degenerate in  $M$ , the magnetic quantum number, which gives rise to the *total state density*  $\omega^{\text{tot}}(E_x)$  which includes the  $2J + 1$  states for each level, i.e.

$$\omega^{\text{tot}}(E_x) = \sum_J \sum_{\Pi} (2J + 1) \rho(E_x, J, \Pi). \quad (8.2)$$

When level densities are given by analytical expressions they are usually factorized as follows

$$\rho(E_x, J, \Pi) = P(E_x, J, \Pi) R(E_x, J) \rho^{\text{tot}}(E_x), \quad (8.3)$$

where  $P(E_x, J, \Pi)$  is the parity distribution and  $R(E_x, J)$  the spin distribution. In all but two level density models in TALYS (**ldmodel 5,6**) (p. 352), the parity equipartition is assumed, i.e.

$$P(E_x, J, \Pi) = \frac{1}{2}, \quad (8.4)$$

However, in our programming, we have accounted for the possibility to adopt non-equal parities, such as e.g. in the case of microscopic level density tables.

## 8.1 Effective level density

We first describe the simplest expressions that are included in TALYS for level densities. We here use the term "effective" to denote that the nuclear collective effects are not explicitly considered, but instead are effectively included in the level density expression.

### 8.1.1 The Fermi Gas Model

Arguably the best known analytical level density expression is that of the Fermi Gas model (FGM). It is based on the assumption that the single particle states which construct the excited levels of the nucleus are equally spaced, and that collective levels are absent. For a two-fermion system, i.e. distinguishing between excited neutrons and protons, the total Fermi gas state density reads

$$\omega_F^{\text{tot}}(E_x) = \frac{\sqrt{\pi} \exp[2\sqrt{aU}]}{12 a^{1/4} U^{5/4}}, \quad (8.5)$$

with  $U$  defined by

$$U = E_x - \Delta, \quad (8.6)$$

where the energy shift  $\Delta$  is an empirical parameter which is equal to, or for some models closely related to, the pairing energy which is included to simulate the known odd-even effects in nuclei. The underlying idea is that  $\Delta$  accounts for the fact that pairs of nucleons must be separated before each component can be excited individually. In practice,  $\Delta$  plays an important role as adjustable parameter to reproduce observables, and its definition can be different for the various models we discuss here. Eq. (8.5) indicates that throughout this manual we will use both the *true* excitation energy  $E_x$ , as basic running variable and for expressions related to discrete levels, and the *effective* excitation energy  $U$ , mostly for expressions related to the continuum.

Eq. (8.5) also contains the level density parameter  $a$ , which theoretically is given by  $a = \frac{\pi^2}{6}(g_\pi + g_\nu)$ , with  $g_\pi$  ( $g_\nu$ ) denoting the spacing of the proton (neutron) single particle states near the Fermi energy. In practice  $a$  is determined, through Eq. (8.5), from experimental information of the specific nucleus under consideration or from global systematics. In contemporary analytical models, it is energy-dependent. This will be discussed in more detail below.

Under the assumption that the projections of the total angular momentum are randomly coupled, it was derived by Ericson [21] that the Fermi gas level density is

$$\rho_F(E_x, J, \Pi) = \frac{1}{2} \frac{2J+1}{2\sqrt{2\pi}\sigma^3} \exp\left[-\frac{(J+\frac{1}{2})^2}{2\sigma^2}\right] \frac{\sqrt{\pi} \exp[2\sqrt{aU}]}{12 a^{1/4} U^{5/4}}, \quad (8.7)$$

where the first factor  $\frac{1}{2}$  represents the aforementioned equiparity distribution and  $\sigma^2$  is the spin cut-off parameter, which represents the width of the angular momentum distribution. It depends on excitation energy and will also be discussed in more detail below. Eq. (8.7) is a special case of the factorization of Eq. (8.3), with the Fermi gas spin distribution given by,

$$R_F(E_x, J) = \frac{2J+1}{2\sigma^2} \exp\left[-\frac{(J+\frac{1}{2})^2}{2\sigma^2}\right]. \quad (8.8)$$

Summing  $\rho_F(E_x, J, \Pi)$  over all spins and parities yields for the total Fermi gas level density

$$\rho_F^{\text{tot}}(E_x) = \frac{1}{\sqrt{2\pi}\sigma} \frac{\sqrt{\pi} \exp[2\sqrt{aU}]}{12 a^{1/4} U^{5/4}}, \quad (8.9)$$

which is, through Eq. (8.5), related to the total Fermi gas state density as

$$\rho_F^{\text{tot}}(E_x) = \frac{\omega_F^{\text{tot}}(E_x)}{\sqrt{2\pi\sigma}}. \quad (8.10)$$

These equations show that  $\rho_F^{\text{tot}}$  and  $\rho_F$  are determined by three parameters,  $a$ ,  $\sigma$  and  $\Delta$ . The first two of these have specific energy dependencies that will now be discussed separately, while we postpone the discussion of  $\Delta$  to the Section on the various specific level density models.

In the Fermi gas model, the level density parameter  $a$  can be derived from  $D_0$ , the average s-wave level spacing at the neutron separation energy  $S_n$ , which is usually obtained from the available experimental set of s-wave resonances. The following equation can be used:

$$\frac{1}{D_0} = \sum_{J=|I-\frac{1}{2}|}^{J=I+\frac{1}{2}} \rho_F(S_n, J, \Pi), \quad (8.11)$$

where  $I$  is the spin of the target nucleus. From this equation, the level density parameter  $a$  can be extracted by an iterative search procedure.

In the TALYS-output, all quantities of interest are printed, if requested.

### 8.1.2 The level density parameter $a$

The formulae described above may suggest a nuclide-specific constant value for the level density parameter  $a$ , and the first level density analyses spanning an entire range of nuclide, see Gilbert et al, Dilg et al, and Baba et al [22, 23, 24] indeed treated  $a$  as a parameter independent of energy. Later, Ignatyuk et al. [25] recognized the correlation between the parameter  $a$  and the shell correction term of the liquid-drop component of the mass formula. They argued that a more realistic level density is obtained by assuming that the Fermi gas formulae outlined above are still valid, but that energy-dependent shell effects should be effectively included through an energy dependent expression for  $a$ . This expression takes into account the presence of shell effects at low energy and their disappearance at high energy in a phenomenological manner. It reads,

$$a = a(E_x) = \tilde{a} \left( 1 + \delta W \frac{1 - \exp[-\gamma U]}{U} \right). \quad (8.12)$$

Here,  $\tilde{a}$  is the asymptotic level density value one would obtain in the absence of any shell effects, i.e.  $\tilde{a} = a(E_x \rightarrow \infty)$  in general, but also  $\tilde{a} = a(E_x)$  for all  $E_x$  if  $\delta W = 0$ . The damping parameter  $\gamma$  determines how rapidly  $a(E_x)$  approaches  $\tilde{a}$ . Finally,  $\delta W$  is the shell correction energy. The absolute magnitude of  $\delta W$  determines how different  $a(E_x)$  is from  $\tilde{a}$  at low energies, while the sign of  $\delta W$  determines whether  $a(E_x)$  decreases or increases as a function of  $E_x$ .

The asymptotic value  $\tilde{a}$  is given by the smooth form

$$\tilde{a} = \alpha A + \beta A^{2/3}, \quad (8.13)$$

where  $A$  is the mass number, while the following systematical formula for the damping parameter is used,

$$\gamma = \frac{\gamma_1}{A^{1/3}} + \gamma_2. \quad (8.14)$$

In Eqs. (8.13)-(8.14),  $\alpha$ ,  $\beta$  and  $\gamma_{1,2}$  are global parameters that have been determined to give the best average level density description over a whole range of nuclides. They are given in Table 8.1, where also the average pairing correction  $\delta_{\text{global}}$  is given. Also,  $\gamma_2 = 0$  by default.

Model	$\alpha$	$\beta$	$\gamma_1$	$\delta_{\text{global}}$
BFM effective	0.0722396	0.195267	0.410289	0.173015
BFM collective	0.0381563	0.105378	0.546474	0.743229
CTM effective	0.0692559	0.282769	0.433090	0.
CTM collective	0.0207305	0.229537	0.473625	0.
GSM effective	0.110575	0.0313662	0.648723	1.13208
GSM collective	0.0357750	0.135307	0.699663	-0.149106

Table 8.1: Global level density parameters for the phenomenological models

We define  $\delta W$ , expressed in MeV, as the difference between the experimental mass of the nucleus  $M_{\text{exp}}$  and its mass according to the spherical liquid-drop model mass  $M_{\text{LDM}}$  (both expressed in MeV),

$$\delta W = M_{\text{exp}} - M_{\text{LDM}}. \quad (8.15)$$

For the real mass we take the value from the AME2020 experimental mass compilation [15]. Following Mengoni and Nakajima [26], for  $M_{\text{LDM}}$  we take the formula by Myers and Swiatecki [27]:

$$\begin{aligned}
M_{\text{LDM}} &= M_n N + M_H Z + E_{\text{vol}} + E_{\text{sur}} + E_{\text{coul}} + \delta \\
M_n &= 8.07144 \text{ MeV} \\
M_H &= 7.28899 \text{ MeV} \\
E_{\text{vol}} &= -c_1 A \\
E_{\text{sur}} &= c_2 A^{2/3} \\
E_{\text{coul}} &= c_3 \frac{Z^2}{A^{1/3}} - c_4 \frac{Z^2}{A} \\
c_i &= a_i \left[ 1 - \kappa \left( \frac{N-Z}{A} \right)^2 \right], \quad i = 1, 2 \\
a_1 &= 15.677 \text{ MeV} \\
a_2 &= 18.56 \text{ MeV} \\
\kappa &= 1.79 \\
c_3 &= 0.717 \text{ MeV} \\
c_4 &= 1.21129 \text{ MeV} \\
\delta &= -\frac{11}{\sqrt{A}} \text{ even - even} \\
&= 0 \quad \text{odd} \\
&= \frac{11}{\sqrt{A}} \text{ odd - odd.}
\end{aligned} \quad (8.16)$$

Eq. (8.12) should in principle be applied at all excitation energies, unless a different level density prescription is used at low energies, as e.g. for the CTM. Therefore, a helpful extra note for practical calculations is that for small excitation energies, i.e.  $E_x \leq \Delta$ , the limiting value of Eq. (8.12) is given by its first order Taylor expansion

$$\lim_{E_x \rightarrow \Delta \rightarrow 0} a(E_x) = \tilde{a} [1 + \gamma \delta W]. \quad (8.17)$$

From now on, wherever the level density parameter  $a$  appears in the formalism, we implicitly assume the form (8.12) for  $a(E_x)$ .



It is important to define the order in which the various parameters of Eq. (8.12) are calculated in TALYS, because they can be given as an adjustable parameter in the input file, they can be known from experiment or they can be determined from systematics.

If the level density parameter at the neutron separation energy  $a(S_n)$  is *not* known from an experimental  $D_0$  value, we use the above systematical formulae for the global level density parameters. In this case, all parameters in Eq. (8.12) are defined and  $a(E_x)$  can be completely computed at any excitation energy. However, for several nuclei  $a(S_n)$  can be derived from an experimental  $D_0$  value through Eq. (8.11), and one may want to use this information. In TALYS, this occurs when an input value for  $a(S_n)$  is given and when `asys n` is set, meaning that instead of using the systematical formulae (8.13)-(8.15) the resonance parameter database is used to determine level density parameters. If we want to use this “experimental”  $a(S_n)$  we are immediately facing a constraint: Eq. (8.12) gives the following condition that must be obeyed

$$a(S_n) = \tilde{a} \left[ 1 + \delta W \frac{1 - \exp(-\gamma(S_n - \Delta))}{S_n - \Delta} \right] \quad (8.18)$$

This means also that  $a(S_n)$ ,  $\tilde{a}$ ,  $\delta W$  and  $\gamma$  cannot *all* be given simultaneously in the input, since it would lead to inconsistency. In the case of an experimental  $a(S_n)$ , at least another parameter must be re-adjusted. The asymptotic level density parameter  $\tilde{a}$  is the first choice. This is not as strange as it seems at first sight. Even though the values for  $\delta W$  depend strongly on the particular theoretical mass model and are merely adopted to reproduce the global trend of shell effects for various regions of the nuclide chart, we do not alter them when going from a global to a local model. Likewise, we feel that it is dangerous to adjust  $\gamma_1$  merely on the basis of discrete level information and the mean resonance spacing at the neutron separation energy, since its presence in the exponent of Eq. (8.12) may lead to level density values that deviate strongly from the global average at high energy.

Hence, if  $a(S_n)$  is given, and also  $\gamma$  and  $\delta W$  are given in the input or from tables then the asymptotic level density parameter  $\tilde{a}$  is automatically obtained from

$$\tilde{a} = a(S_n) / \left[ 1 + \delta W \frac{1 - \exp(-\gamma(S_n - \Delta))}{S_n - \Delta} \right] \quad (8.19)$$

Other choices can however be forced with the input file. If  $a(S_n)$ ,  $\tilde{a}$  and  $\delta W$  are given while  $\gamma$  is not given in the input,  $\gamma$  is eliminated as a free parameter and is obtained by inverting Eq. (8.18),

$$\gamma = -\frac{1}{S_n - \Delta} \ln \left[ 1 - \frac{S_n - \Delta}{\delta W} \left( \frac{\tilde{a}}{a(S_n)} - 1 \right) \right] \quad (8.20)$$

If  $\delta W$  is the only parameter not given in the input, it is automatically determined by inverting Eq. (8.18),

$$\delta W = \frac{(S_n - \Delta) \left( \frac{a(S_n)}{\tilde{a}} - 1 \right)}{1 - \exp(-\gamma(S_n - \Delta))} \quad (8.21)$$

All this flexibility is not completely without danger. Since both  $\delta W$  and  $a(S_n)$  are independently derived from experimental values, it may occur that Eq. (8.20) poses problems. In particular,  $\delta W$  may have a sign opposite to  $[a(S_n) - \tilde{a}]$ . In other words, if the argument of the natural logarithm is not between 0 and 1, our escape route is to return to Eq. (8.14) for  $\gamma$  and to readjust  $\delta W$  through Eq. (8.21).

The recipe outlined above represents a full-proof method to deal with all the parameters of Eq. (8.12), i.e. it always gives a reasonable answer since we are able to invert the Ignatyuk formula

in all possible ways. We emphasize that in general, consistent calculations are obtained with Eqs. (8.15), (8.13), (8.14), and when available, a specific  $a(S_n)$  value from the tables. The full range of possibilities of parameter specification for the Ignatyuk formula is summarized in Table 8.2. The reason to include all these parameter possibilities is simple: fitting experiments. Moreover, these variations are not as unphysical as they may seem: Regardless of whether they are derived from experimental data or from microscopic nuclear structure models, the parameters  $a(S_n)$ ,  $\tilde{a}$ ,  $\delta W$  and  $\gamma$  always have an uncertainty. Hence, as long as the deviation from their default values is kept within bounds, they can be helpful fitting parameters.

Table 8.2: Specification of parameter handling for Ignatyuk formula

Input	Calculation
- (Default)	$\gamma$ : Eq. (8.14), $\tilde{a}$ : Eq. (8.13) or Eq. (8.19), $\delta W$ : Eq. (8.15), $a(S_n)$ : Eq. (8.18)
$a(S_n), \gamma, \tilde{a}, \delta W$	TALYS-Error: Conflict
$a(S_n)$ (table), $\gamma, \tilde{a}, \delta W$	$a(S_n)$ : Eq. (8.18) (Input overruled)
$\gamma, \tilde{a}, \delta W$	$a(S_n)$ : Eq. (8.18)
$\gamma, \tilde{a}$	$\delta W$ : Eq. (8.15), $a(S_n)$ : Eq. (8.18)
$\gamma, \delta W$	$\tilde{a}$ : Eq. (8.13), $a(S_n)$ : Eq. (8.18)
$\tilde{a}, \delta W$	$\gamma$ : Eq. (8.14), $a(S_n)$ : Eq. (8.18)
$\gamma$	$\tilde{a}$ : Eq. (8.13), $\delta W$ : Eq. (8.15), $a(S_n)$ : Eq. (8.18)
$\tilde{a}$	$\gamma$ : Eq. (8.14), $\delta W$ : Eq. (8.15), $a(S_n)$ : Eq. (8.18)
$\delta W$	$\gamma$ : Eq. (8.14), $\tilde{a}$ : Eq. (8.13), $a(S_n)$ : Eq. (8.18)
$a(S_n)$	$\gamma$ : Eq. (8.20) or (8.14), $\tilde{a}$ : Eq. (8.19), $\delta W$ : Eq. (8.15) or (8.21)
$a(S_n), \tilde{a}, \delta W$	$\gamma$ : Eq. (8.20)
$a(S_n), \delta W$	$\tilde{a}$ : Eq. (8.19) $\gamma$ : Eq. (8.20)
$a(S_n), \tilde{a}$	$\delta W$ : Eq. (8.15), $\gamma$ : Eq. (8.20) or $\gamma$ : Eq. (8.14), $\delta W$ : Eq. (8.21)
$a(S_n), \gamma$	$\delta W$ : Eq. (8.15), $\tilde{a}$ : Eq. (8.19)
$a(S_n), \gamma, \delta W$	$\tilde{a}$ : Eq. (8.19)
$a(S_n), \gamma, \tilde{a}$	$\delta W$ : Eq. (8.21)

### 8.1.3 The spin cut-off parameter

The spin cut-off parameter  $\sigma^2$  represents the width of the angular momentum distribution of the level density. The general expression for the continuum is based on the observation that a nucleus possesses collective rotational energy that can not be used to excite the individual nucleons. In this picture, one can relate  $\sigma^2$  to the (undeformed) moment of inertia of the nucleus  $I_0$  and the thermodynamic temperature  $t$ ,

$$t = \sqrt{\frac{U}{a}}. \quad (8.22)$$

Indeed, an often used expression is  $\sigma^2 = \sigma_{\parallel}^2 = I_0 t$ , where the symbol  $\sigma_{\parallel}^2$  designates the parallel spin cut-off parameter, obtained from the projection of the angular momentum of the single-particle states on the symmetry axis. However, it has been observed from microscopic level density studies that the quantity  $\sigma^2/t$  is not constant as argued by Goriely and Hilaire [28, 29], but instead

shows marked shell effects, similar to the level density parameter  $a$ . To take that effect into account we follow Goriely and Mughabghab [28, 30] and adopt the following expression,

$$\sigma^2 = \sigma_{\parallel}^2 = \sigma_F^2(E_x) = I_0 \frac{a}{\tilde{a}}, \quad (8.23)$$

with  $\tilde{a}$  from Eq. (8.13) and

$$I_0 = \frac{\frac{2}{3}m_0R^2A}{(\hbar c)^2}, \quad (8.24)$$

where  $R = 1.2A^{1/3}$  is the radius, and  $m_0$  the neutron mass in amu. This gives

$$\sigma_F^2(E_x) = 0.01389 \frac{A^{5/3}}{\tilde{a}} \sqrt{aU}. \quad (8.25)$$

On average, the  $\sqrt{aU}/\tilde{a}$  has the same energy- and mass-dependent behaviour as the temperature  $\sqrt{U/a}$ . The differences occur in the regions with large shell effects. Eq. (8.25) can be altered by means of **Rspincut** (p. 375).

Analogous to the level density parameter, we have to account for low excitation energies for which Eq. (8.25) is not defined ( $E_x \leq \Delta$ ) or less appropriate. This leads us to an alternative method to determine the spin cut-off parameter, namely from the spins of the low-lying discrete levels. Suppose we want to determine this discrete spin cut-off parameter  $\sigma_d^2$  in the energy range where the total level density agrees well with the discrete level sequence, i.e. from a lower discrete level  $N_L$  with energy  $E_L$  to an upper level  $N_U$  with energy  $E_U$ . It can be derived that

$$\sigma_d^2 = \frac{1}{3 \sum_{i=N_L}^{N_U} (2J_i + 1)} \sum_{i=N_L}^{N_U} J_i(J_i + 1)(2J_i + 1). \quad (8.26)$$

where  $J_i$  is the spin of discrete level  $i$ . Reading these spins from the discrete level file readily gives the value for  $\sigma_d^2$ . In TALYS,  $\sigma_d^2$  can be used on a nucleus-by-nucleus basis, when discrete levels are known. For cases where either Eqs. (8.25) or (8.26) are not applicable, e.g. because there are no discrete levels and  $U = E_x - P$  is negative, we take the systematical formula

$$\sigma^2 = (0.83A^{0.26})^2 \quad (8.27)$$

which gives a reasonable estimate for energies in the order of 1-2 MeV.

The final functional form for  $\sigma^2(E_x)$  is a combination of Eqs. (8.25) and (8.26). Defining  $E_d = \frac{1}{2}(E_L + E_U)$  as the energy in the middle of the  $N_L - N_U$  region, we assume  $\sigma_d^2$  is constant up to this energy and can then be linearly interpolated to the expression given by Eq. (8.25). We choose the matching point to be the neutron separation energy  $S_n$  of the nucleus under consideration, i.e.

$$\begin{aligned} \sigma^2(E_x) &= \sigma_d^2 && \text{for } 0 \leq E_x \leq E_d \\ &= \sigma_d^2 + \frac{E_x - E_d}{S_n - E_d} (\sigma_F^2(E_x) - \sigma_d^2) && \text{for } E_d \leq E_x \leq S_n \\ &= \sigma_F^2(E_x) && \text{for } E_x \geq S_n. \end{aligned} \quad (8.28)$$

Analogous to the level density parameter  $a$ , from now on we implicitly assume the energy dependence for  $\sigma^2(E_x)$  whenever  $\sigma^2$  appears in the formalism.

### 8.1.4 Constant Temperature Model

In the Constant Temperature Model (CTM), as introduced by Gilbert and Cameron [22], the excitation energy range is divided into a low energy part from 0 MeV up to a matching energy  $E_M$ , where the so-called constant temperature law applies and a high energy part above  $E_M$ , where the Fermi gas model applies. Hence, for the total level density we have

$$\begin{aligned}\rho^{\text{tot}}(E_x) &= \rho_T^{\text{tot}}(E_x), \quad \text{if } E_x \leq E_M, \\ &= \rho_F^{\text{tot}}(E_x), \quad \text{if } E_x \geq E_M,\end{aligned}\quad (8.29)$$

and similarly for the level density

$$\begin{aligned}\rho(E_x, J, \Pi) &= \frac{1}{2} R_F(E_x, J) \rho_T^{\text{tot}}(E_x), \quad \text{if } E_x \leq E_M, \\ &= \rho_F(E_x, J, \Pi), \quad \text{if } E_x \geq E_M.\end{aligned}\quad (8.30)$$

Note that the spin distribution of Eq. (8.8) is also used in the constant temperature region, including the low-energy behaviour for the spin cut-off parameter as expressed by Eq. (8.28).

For the Fermi gas expression, we use the effective excitation energy  $U = E_x - \Delta^{\text{CTM}}$ , where the energy shift is given by

$$\Delta^{\text{CTM}} = \chi \frac{12}{\sqrt{A}}, \quad (8.31)$$

with

$$\begin{aligned}\chi &= 0, \quad \text{for odd - odd,} \\ &= 1, \quad \text{for odd - even,} \\ &= 2, \quad \text{for even - even.}\end{aligned}\quad (8.32)$$

Note that by default no adjustable pairing shift parameter is used in the CTM. In TALYS, the number 12 in the numerator of Eq. (8.31) can be altered using **pairconstant** (p. 392). This also applies to the other level density models.

For low excitation energy, the CTM is based on the experimental evidence that the cumulated histogram  $N(E_x)$  of the first discrete levels can be well reproduced by an exponential law of the type

$$N(E_x) = \exp\left(\frac{E_x - E_0}{T}\right), \quad (8.33)$$

which is called the constant temperature law. The nuclear temperature  $T$  and  $E_0$  are parameters that serve to adjust the formula to the experimental discrete levels. Accordingly, the constant temperature part of the total level density reads

$$\rho_T^{\text{tot}}(E_x) = \frac{dN(E_x)}{dE_x} = \frac{1}{T} \exp\left(\frac{E_x - E_0}{T}\right). \quad (8.34)$$

For higher energies, the Fermi gas model is more suitable and the total level density is given by Eq. (8.9). The expressions for  $\rho_T^{\text{tot}}$  and  $\rho_F^{\text{tot}}$  have to be matched at a matching energy  $E_M$  where they, and their derivatives, are identical. First, continuity requires that

$$\rho_T^{\text{tot}}(E_M) = \rho_F^{\text{tot}}(E_M). \quad (8.35)$$

Inserting Eq. (8.34) in this equation directly leads to the condition

$$E_0 = E_M - T \ln [T \rho_F^{\text{tot}}(E_M)]. \quad (8.36)$$

Second, continuity of the derivatives requires that

$$\frac{d\rho_T^{\text{tot}}}{dE_x}(E_M) = \frac{d\rho_F^{\text{tot}}}{dE_x}(E_M). \quad (8.37)$$

Inserting Eq. (8.34) in this equation directly leads to the condition

$$\frac{\rho_T^{\text{tot}}(E_M)}{T} = \frac{d\rho_F^{\text{tot}}}{dE_x}(E_M), \quad (8.38)$$

or

$$\frac{1}{T} = \frac{d \ln \rho_F^{\text{tot}}}{dE_x}(E_M). \quad (8.39)$$

In principle, for all Fermi gas type expressions, including the energy dependent expressions for  $a$ ,  $\sigma^2$ ,  $K_{\text{rot}}$  etc., Eq. (8.39) could be elaborated analytically, but in practice we use a numerical approach to allow any level density model to be used in the matching problem. For this, we determine the inverse temperature of Eq. (8.39) numerically by calculating  $\rho_F^{\text{tot}}$  on a sufficiently dense energy grid.

The matching problem gives us two conditions, given by Eqs. (8.36) and (8.39), with three unknowns:  $T$ ,  $E_0$  and  $E_M$ . Hence, we need another constraint. This is obtained by demanding that in the discrete level region the constant temperature law reproduces the experimental discrete levels, i.e.  $\rho_T^{\text{tot}}$  needs to obey

$$N_U = N_L + \int_{E_L}^{E_U} dE_x \rho^{\text{tot}}(E_x), \quad (8.40)$$

or, after inserting Eq. (8.34),

$$N_U = N_L + \left( \exp\left[\frac{E_U}{T}\right] - \exp\left[\frac{E_L}{T}\right] \right) \exp\left[\frac{-E_0}{T}\right]. \quad (8.41)$$

The combination of Eqs. (8.36), (8.39) and (8.41) determines  $T$ ,  $E_0$  and  $E_M$ . Inserting Eq. (8.36) in Eq. (8.41) yields:

$$T \rho_F^{\text{tot}}(E_M) \exp\left[\frac{-E_M}{T}\right] \left( \exp\left[\frac{E_U}{T}\right] - \exp\left[\frac{E_L}{T}\right] \right) + N_L - N_U = 0, \quad (8.42)$$

from which  $E_M$  can be solved by an iterative procedure with the simultaneous use of the tabulated values given by Eq. (8.39). The levels  $N_L$  and  $N_U$  are chosen such that  $\rho_T(E_x)$  gives the best description of the observed discrete states and are stored in the nuclear structure database. For nuclides for which no, or not enough, discrete levels are given we rely on empirical formula for the temperature. For the effective model,

$$T = -0.22 + \frac{9.4}{\sqrt{A(1 + \gamma\delta W)}} \quad (8.43)$$

and for the collective model

$$T = -0.25 + \frac{10.2}{\sqrt{A(1 + \gamma\delta W)}} \quad (8.44)$$

where  $\gamma$  is taken from Eq. (8.14) and Table 8.1. Next, we directly obtain  $E_M$  from Eq. (8.39) and  $E_0$  from Eq. (8.36). Again, Eqs. (8.43) and (8.44) were obtained by fitting all individual values of the nuclides for which sufficient discrete level information exists. In a few cases, the global expression for  $T$  leads to a value for  $E_M$  which is clearly off scale. In that case, we resort to empirical expressions for the matching energy. For the effective model

$$E_M = 2.33 + 253/A + \Delta^{CTM}, \quad (8.45)$$

and for the collective model

$$E_M = 2.67 + 253/A + \Delta^{CTM}, \quad (8.46)$$

after which we obtain  $T$  from Eq. (8.39).

### 8.1.5 The Back-shifted Fermi gas Model

In the Back-shifted Fermi gas Model (BFM) of Dilg et al [23], the pairing energy is treated as an adjustable parameter and the Fermi gas expression is used all the way down to 0 MeV. Hence for the total level density we have

$$\rho_F^{\text{tot}}(E_x) = \frac{1}{\sqrt{2\pi\sigma}} \frac{\sqrt{\pi} \exp[2\sqrt{aU}]}{12 a^{1/4} U^{5/4}}, \quad (8.47)$$

and for the level density,

$$\rho_F(E_x, J, \Pi) = \frac{1}{2} \frac{2J+1}{2\sqrt{2\pi\sigma^3}} \exp\left[-\frac{(J+\frac{1}{2})^2}{2\sigma^2}\right] \frac{\sqrt{\pi} \exp[2\sqrt{aU}]}{12 a^{1/4} U^{5/4}}, \quad (8.48)$$

respectively. These expressions, as well as the energy-dependent expressions for  $a$  and  $\sigma^2$ , contain the effective excitation energy  $U = E_x - \Delta^{\text{BFM}}$ , where the energy shift is given by

$$\Delta^{\text{BFM}} = \chi \frac{12}{\sqrt{A}} + \delta, \quad (8.49)$$

with

$$\begin{aligned} \chi &= -1, \text{ for odd - odd,} \\ &= 0, \text{ for odd - even,} \\ &= 1, \text{ for even - even,} \end{aligned} \quad (8.50)$$

and  $\delta$  an adjustable parameter to fit experimental data per nucleus.

A problem of the original BFM, which may have hampered its use as the default level density option in nuclear model analyses, is the divergence of Eqs. (8.47)-(8.48) when  $U$  goes to zero. A solution to this problem has been provided by Grossjean and Feldmeier [31], has been put into a practical form by Demetriou and Goriely [32], and is adopted in TALYS. The expression for the total BFM level density is

$$\rho_{\text{BFM}}^{\text{tot}}(E_x) = \left[ \frac{1}{\rho_F^{\text{tot}}(E_x)} + \frac{1}{\rho_0(t)} \right]^{-1}, \quad (8.51)$$

where  $\rho_0$  is given by

$$\rho_0(t) = \frac{\exp(1)}{24\sigma} \frac{(a_n + a_p)^2}{\sqrt{a_n a_p}} \exp(4a_n a_p t^2), \quad (8.52)$$

where  $a_n = a_p = a/2$  and  $t$  is given by Eq. (8.22).

With the usual spin distribution, the level density reads

$$\rho_{\text{BFM}}(E_x, J, \Pi) = \frac{1}{2} \frac{2J+1}{2\sigma^2} \exp\left[-\frac{(J+\frac{1}{2})^2}{2\sigma^2}\right] \rho_{\text{BFM}}^{\text{tot}}(E_x). \quad (8.53)$$

In sum, there are two adjustable parameters for the BFM,  $a$  and  $\delta$ .

### 8.1.6 The Generalized Superfluid Model

The Generalized Superfluid Model (GSM) takes superconductive pairing correlations into account according to the Bardeen-Cooper-Schrieffer theory. The phenomenological version of the model, by Ignatyuk et al. [33, 34] is characterized by a phase transition from a superfluid behaviour at low energy, where pairing correlations strongly influence the level density, to a high energy region which is described by the FGM. The GSM thus resembles the CTM to the extent that it distinguishes between a low energy and a high energy region, although for the GSM this distinction follows naturally from the theory and does not depend on specific discrete levels that determine a matching energy. Instead, the model automatically provides a constant temperature-like behaviour at low energies. For the level density expressions, it is useful to recall the general formula for the total level density,

$$\rho^{\text{tot}}(E_x) = \frac{1}{\sqrt{2\pi\sigma}} \frac{e^S}{\sqrt{D}}, \quad (8.54)$$

where  $S$  is the entropy and  $D$  is the determinant related to the saddle-point approximation. For the GSM this expression has two forms: one below and one above the so called critical energy  $U_c$ .

For energies below  $U_c$ , the level density is described in terms of thermodynamical functions defined at  $U_c$ , which is given by

$$U_c = a_c T_c^2 + E_{\text{cond}}. \quad (8.55)$$

Here, the critical temperature  $T_c$  is

$$T_c = 0.567\Delta_0, \quad (8.56)$$

where the pairing correlation function is given by

$$\Delta_0 = \frac{12}{\sqrt{A}}. \quad (8.57)$$

This correlation function also determines the condensation energy  $E_{\text{cond}}$ , which characterizes the decrease of the superfluid phase relative to the Fermi gas phase. It is given by the expression

$$E_{\text{cond}} = \frac{3}{2\pi^2} a_c \Delta_0^2, \quad (8.58)$$

where the critical level density parameter  $a_c$  is given by the iterative equation

$$a_c = \tilde{a} \left[ 1 + \delta W \frac{1 - \exp(-\gamma a_c T_c^2)}{a_c T_c^2} \right], \quad (8.59)$$

which is easily obtained once  $\tilde{a}$ ,  $\delta W$  and  $\gamma$  are known. Eq. (8.59) indicates that shell effects are again appropriately taken into account. For the determination of the level density we also invoke the expression for the critical entropy  $S_c$ ,

$$S_c = 2a_c T_c, \quad (8.60)$$

the critical determinant  $D_c$ ,

$$D_c = \frac{144}{\pi} a_c^3 T_c^5, \quad (8.61)$$

and the critical spin cut-off parameter  $\sigma_c^2$ ,

$$\sigma_c^2 = 0.01389 A^{5/3} \frac{a_c}{\tilde{a}} T_c. \quad (8.62)$$

Now that everything is specified at  $U_c$ , we can use the superfluid Equation Of State (EOS) to define the level density below  $U_c$ . For this, we define an effective excitation energy

$$U' = E_x + \chi \Delta_0 + \delta, \quad (8.63)$$

where

$$\begin{aligned} \chi &= 2, \text{ for odd - odd,} \\ &= 1, \text{ for odd - even,} \\ &= 0, \text{ for even - even,} \end{aligned} \quad (8.64)$$

and  $\delta$  is an adjustable shift parameter to obtain the best description of experimental data per nucleus. Note that the convention for  $\chi$  is again different from that of the BFM or CTM. Defining

$$\varphi^2 = 1 - \frac{U'}{U_c}, \quad (8.65)$$

then for  $U' \leq U_c$  the quantities  $\varphi$  and  $T$  obey the superfluid EOS, according to Ignatyuk et al [33],

$$\varphi = \tanh\left(\frac{T_c}{T} \varphi\right), \quad (8.66)$$

which is equivalent to

$$T = 2T_c \varphi \left[ \ln \frac{1+\varphi}{1-\varphi} \right]^{-1}. \quad (8.67)$$

The other required functions for  $U' \leq U_c$  are the entropy  $S$ ,

$$S = S_c \frac{T_c}{T} (1 - \varphi^2) = S_c \frac{T_c U'}{T U_c}, \quad (8.68)$$

the determinant  $D$ ,

$$D = D_c (1 - \varphi^2) (1 + \varphi^2)^2 = D_c \frac{U'}{U_c} \left(2 - \frac{U'}{U_c}\right)^2, \quad (8.69)$$

and the spin cut-off parameter

$$\sigma^2 = \sigma_c^2 (1 - \varphi^2) = \sigma_c^2 \frac{U'}{U_c}. \quad (8.70)$$

In sum, the level density can now be specified for the entire energy range. For  $U' \leq U_c$ , the total level density is given by

$$\rho_{\text{GSM}}^{\text{tot}}(E_x) = \frac{1}{\sqrt{2\pi\sigma}} \frac{e^S}{\sqrt{D}}, \quad (8.71)$$



using Eqs. (8.68)-(8.70). Similarly, the level density is

$$\rho_{\text{GSM}}(E_x, J, \Pi) = \frac{1}{2} R_F(E_x, J) \rho_{\text{GSM}}^{\text{tot}}(E_x). \quad (8.72)$$

For  $U' \geq U_c$  the FGM applies, though with an energy shift that is different from the pairing correction of the CTM and BFM. The total level density is

$$\rho_{\text{GSM}}^{\text{tot}}(E_x) = \frac{1}{\sqrt{2\pi\sigma}} \frac{\sqrt{\pi} \exp[2\sqrt{aU}]}{12 a^{1/4} U^{5/4}}, \quad (8.73)$$

where the effective excitation energy is defined by  $U = E_x - \Delta^{\text{GSM}}$ , with

$$\Delta^{\text{GSM}} = E_{\text{cond}} - \chi\Delta_0 - \delta. \quad (8.74)$$

The spin cut-off parameter in the high-energy region reads

$$\sigma^2 = I_0 \frac{a}{\bar{a}} \sqrt{\frac{U}{a}}, \quad (8.75)$$

and  $I_0$  is given by Eq. (8.24). The level density is given by

$$\rho_{\text{GSM}}(E_x, J, \Pi) = \frac{1}{2} R_F(E_x, J) \rho_{\text{GSM}}^{\text{tot}}(E_x). \quad (8.76)$$

At the matching energy, i.e., for  $E'_x = U_c - \chi\Delta_0 - \delta$ , it is easy to verify that Eqs. (8.71) and (8.73) match so that the total level density is perfectly continuous. In sum, there are two adjustable parameters for the GSM,  $a$  and  $\delta$ .

## 8.2 Collective effects in the level density

All the previously described models do not explicitly account for collective effects. However, it is well known that generally the first excited levels of nuclei result from coherent excitations of the fermions it contains. The Fermi gas model is not appropriate to describe such levels. Nevertheless, the models presented so far can still be applied successfully in most cases since they incorporate collectivity in the level density in an effective way through a proper choice of the energy-dependent level density parameter values.

In some calculations, especially if the disappearance of collective effects with excitation energy plays a role (e.g. in the case of fission), one would like to model the collective effects in more detail. It can be shown that the collective effects may be accounted for explicitly by introducing collective enhancement factors on top of an intrinsic level density  $\rho_{F,\text{int}}(E_x, J, \Pi)$ . Then the deformed Fermi gas level density  $\rho_{F,\text{def}}(E_x, J, \Pi)$  reads

$$\rho_{F,\text{def}}(E_x, J, \Pi) = K_{\text{rot}}(E_x) K_{\text{vib}}(E_x) \rho_{F,\text{int}}(E_x, J, \Pi), \quad (8.77)$$

while the total level densities  $\rho_{F,\text{def}}^{\text{tot}}$  and  $\rho_{F,\text{int}}^{\text{tot}}$  are related in the same way.  $K_{\text{rot}}$  and  $K_{\text{vib}}$  are called the rotational and vibrational enhancement factors, respectively. If  $K_{\text{rot}}$  and  $K_{\text{vib}}$  are explicitly accounted for,  $\rho_{F,\text{int}}(E_x, J, \Pi)$  should now describe purely single-particle excitations, and can be determined again by using the Fermi gas formula. Obviously, the level density parameter  $a$  of  $\rho_{F,\text{int}}$  will be different from that of the effective level density described before.

The vibrational enhancement of the level density is approximated, see Capote et al [6], by

$$K_{\text{vib}} = \exp[\delta S - (\delta U/t)], \quad (8.78)$$

where  $\delta S$  and  $\delta U$  are changes in the entropy and excitation energy, respectively, resulting from the vibrational modes and  $T$  is the nuclear temperature given by Eq. (8.22). These changes are described by the Bose gas relationships, i.e

$$\begin{aligned}\delta S &= \sum_i (2\lambda_i + 1) \left[ (1 + n_i) \ln(1 + n_i) - n_i \ln n_i \right], \\ \delta U &= \sum_i (2\lambda_i + 1) \omega_i n_i,\end{aligned}\quad (8.79)$$

where  $\omega_i$  are the energies,  $\lambda_i$  the multiplicities, and  $n_i$  the occupation numbers for vibrational excitations at a given temperature. The disappearance of collective enhancement of the level density at high temperatures can be taken into account by defining the occupation numbers in terms of the equation

$$n_i = \frac{\exp(-\gamma_i/2\omega_i)}{\exp(\omega_i/T) - 1}, \quad (8.80)$$

where  $\gamma_i$  are the spreading widths of the vibrational excitations. This spreading of collective excitations in nuclei should be similar to the zero-sound damping in a Fermi liquid, and the corresponding width can be written as

$$\gamma_i = C(\omega_i^2 + 4\pi^2 T^2). \quad (8.81)$$

The value of  $C = 0.0075 A^{1/3}$  was obtained from the systematics of the neutron resonance densities of medium-weight nuclei by Grudzevich et al [35]. We use a modified systematics, see Capote et al [6] which includes shell effects to estimate the phonon energies (in MeV), namely

$$\omega_2 = 65A^{-5/6}/(1 + 0.05\delta W), \quad (8.82)$$

for the quadrupole vibrations and

$$\omega_3 = 100A^{-5/6}/(1 + 0.05\delta W), \quad (8.83)$$

for the octupole excitations.

An alternative, liquid drop model, estimation of the vibrational collective enhancement factor is given by Iljinov et al. [36]

$$K_{\text{vib}}(E_x) = \exp\left(0.0555A^{\frac{2}{3}}t^{\frac{4}{3}}\right). \quad (8.84)$$

The **kvibmodel** keyword can be used to choose between these models.

A more important contribution to the collective enhancement of the level density originates from rotational excitations. Its effect is not only much stronger ( $K_{\text{rot}} \sim 10 - 100$  whereas  $K_{\text{vib}} \sim 3$ ), but the form for the rotational enhancement depends on the nuclear shape as well. This makes it crucial, among others, for the description of fission cross sections.

The expression for the rotational enhancement factor depends on the deformation, see Capote et al and Junghans et al [6, 37]. Basically,  $K_{\text{rot}}$  is equal to the perpendicular spin cut-off parameter  $\sigma_{\perp}^2$ ,

$$\sigma_{\perp}^2 = I_{\perp} t, \quad (8.85)$$

with the rigid-body moment of inertia perpendicular to the symmetry axis given by

$$I_{\perp} = I_0 \left(1 + \frac{\beta_2}{3}\right) = 0.01389A^{5/3} \left(1 + \frac{\beta_2}{3}\right), \quad (8.86)$$

where  $\beta_2$  is the ground-state quadrupole deformation, which we take from the nuclear structure database. Hence,

$$\sigma_{\perp}^2 = 0.01389A^{5/3} \left(1 + \frac{\beta_2}{3}\right) \sqrt{\frac{U}{a}}. \quad (8.87)$$

For high excitation energies, it is known that the rotational behavior vanishes. To take this into account, it is customary to introduce a phenomenological damping function  $f(E_x)$  which is equal to 1 in the purely deformed case and 0 in the spherical case. The expression for the level density is then

$$\begin{aligned} \rho(E_x, J, \Pi) &= [1 - f(E_x)]K_{\text{vib}}(E_x)\rho_{F,\text{int}}(E_x, J, \Pi) + f(E_x)\rho_{F,\text{def}}(E_x, J, \Pi) \\ &= K_{\text{rot}}(E_x)K_{\text{vib}}(E_x)\rho_{F,\text{int}}(E_x, J, \Pi) \end{aligned} \quad (8.88)$$

where

$$K_{\text{rot}}(E_x) = \max([\sigma_{\perp}^2 - 1]f(E_x) + 1, 1). \quad (8.89)$$

The function  $f(E_x)$  is taken as a combination of Fermi functions,

$$f(E_x) = \frac{1}{1 + \exp\left(\frac{E_x - E_{\text{col}}^{\text{g.s.}}}{d_{\text{col}}^{\text{g.s.}}}\right)}, \quad (8.90)$$

which yields the desired property of  $K_{\text{rot}}$  going to 1 for high excitation energy. Little is known about the parameters that govern this damping, although attempts have been made (see e.g. Hansen et al[38]). We arbitrarily take  $E_{\text{col}}^{\text{g.s.}} = 30$  MeV,  $d_{\text{col}}^{\text{g.s.}} = 5$  MeV.

Finally, these collective enhancement expressions can be applied to the various phenomenological level density models. The CTM formalism can be extended with explicit collective enhancement, i.e. the total level density reads

$$\begin{aligned} \rho^{\text{tot}}(E_x) &= \rho_T^{\text{tot}}(E_x), \quad \text{if } E_x \leq E_M, \\ &= K_{\text{rot}}(E_x)K_{\text{vib}}(E_x)\rho_{F,\text{int}}^{\text{tot}}(E_x), \quad \text{if } E_x \geq E_M, \end{aligned} \quad (8.91)$$

and similarly for the level density  $\rho(E_x, J, \Pi)$ . Note that the collective enhancement is not applied to the constant temperature region, since collectivity is assumed to be already implicitly included in the discrete levels. The matching problem is completely analogous to that described before, although the resulting parameters  $E_M$ ,  $E_0$  and  $T$  will of course be different.

The BFM can also be extended with explicit collective enhancement, i.e.

$$\rho_{\text{BFM}}^{\text{tot}}(E_x) = K_{\text{rot}}(E_x)K_{\text{vib}}(E_x) \left[ \frac{1}{\rho_{F,\text{int}}^{\text{tot}}(E_x)} + \frac{1}{\rho_0(t)} \right]^{-1}, \quad (8.92)$$

and similarly for the level density  $\rho(E_x, J, \Pi)$ . Finally, the GSM can be extended as follows

$$\rho_{\text{GSM}}^{\text{tot}}(E_x) = K_{\text{rot}}(E_x)K_{\text{vib}}(E_x)\rho_{\text{GSM},\text{int}}^{\text{tot}}(E_x). \quad (8.93)$$

(In fact, the term ‘‘general’’ in the GSM was originally meant for the collective enhancement).

**Options 8.2.1** Since phenomenological level densities need to be generated from analytical expressions for the total level density, spin distribution, collective enhancement, shell models, etc. they come with a large number of models and adjustable parameters. First of all, different level density models can be chosen with **ldmodel** (p. 352) The global level density parameters of Eq. (8.13), see table 8.1, can be adjusted with the keywords **alphald** (p. 387), and **betald** (p. 388), and the global shell damping parameter of Eq. (8.14) with **gammashell1** (p. 390)

and **gammashell2** (p. 391). The global pairing correction can be changed with **Pshiftconstant** (p. 374). For a particular nucleus, the level density parameter at the binding energy can be adjusted with **a** (p. 354), the asymptotic level density parameter at infinite excitation energy with **alimit** (p. 356), and the shell damping parameter with **gammald** (p. 25). In addition, we can impose a relative adjustment to the level density parameter  $a$  (so we don't need to know its actual value) with **aadjust** (p. 355). One can also choose to use a global model for the level density parameter with **asys** (p. 357). For the CTM, the keywords **T** (p. 368), **Tadjust** (p. 369), **E0** (p. 370), **E0adjust** (p. 371), **Exmatchadjust** (p. 367), **Exmatch** (p. 366), **Nlow** (p. 364), **Ntop** (p. 365), **CTMglobal** (p. 385), are available. The pairing energy can be adjusted with the keywords **pairconstant** (p. 392), **pair** (p. 359), **Pshift** (p. 372), **Pshiftadjust** (p. 373). The user can change the model and parameters for the spin cutoff parameter with **spincutmodel** (p. 382), **Rspincut** (p. 375), **s2adjust** (p. 377), the shell model with **shellmodel** (p. 383), and **deltaW** (p. 358), and vibrational enhancement with **Kvibmodel** (p. 384). Collective enhancement can be enabled or disabled with **colenhance** (p. 380), and further adjustment of collective enhancement can be attained with **colldamp** (p. 381), **Krotconstant** (p. 393), **Ufermi** (p. 396), **Ufermibf** (p. 397), **cfermi** (p. 394), **cfermibf** (p. 395). For fission level densities, **Rtransmom** (p. 398), **Rclass2mom** (p. 399), are available. The amount of level densities in output files can be steered with **filedensity** (p. 378), **outdensity** (p. 379).

#### Implementation 8.2.1 Level density parameters are determined in *densitypar.f*.

The level density parameters are stored in the *talys/structure/density/* directory. First, there are 4 subdirectories, *ground/*, *fission/*, *ph/*, *phjp* for ground state, fission barrier, particle-hole level densities, and spin-parity dependent particle-hole level densities, respectively. In *ground/ctm/*, *ground/bfm/*, and *ground/gsm/*, phenomenological level density parameters are stored. When the level density parameter **a** is given, it is derived from a fit to the  $D_0$  resonance spacing of the RIPL database. As an example, below are the parameters for the Zr-isotopes from *ground/bfm/Zr.ld*. For each isotope, we have stored  $Z$ ,  $A$ , and then for both the effective and explicit collective model  $N_L$ ,  $N_U$ , level density parameter  $a$  in  $\text{MeV}^{-1}$ , and pairing correction in MeV, nuclear symbol. The format is (2i4,2(2i4,2f12.5)4x,i4,a2).

40	86	8	17	0.00000	0.51730	8	17	0.00000	1.04574	86Zr
40	87	8	26	0.00000	0.15936	8	26	0.00000	0.65313	87Zr
40	88	5	17	0.00000	-0.04425	5	17	0.00000	0.46704	88Zr
40	89	2	15	0.00000	0.39230	2	15	0.00000	0.96543	89Zr
40	90	8	16	0.00000	0.99793	8	16	0.00000	1.80236	90Zr
40	91	3	17	9.57730	0.40553	3	17	5.13903	1.15428	91Zr
40	92	3	16	9.94736	-0.03571	3	16	5.00387	0.39829	92Zr
40	93	8	18	11.07918	0.50929	8	18	5.92169	1.07202	93Zr
40	94	4	16	12.51538	0.28941	4	16	6.66540	0.60420	94Zr
40	95	3	18	11.98832	0.83535	3	18	5.97084	1.28803	95Zr
40	96	3	16	0.00000	0.64799	3	16	0.00000	1.06393	96Zr
40	98	8	16	0.00000	0.32758	8	16	0.00000	0.67594	98Zr
40	99	2	21	0.00000	-0.01325	2	21	0.00000	0.26100	99Zr
40	100	8	17	0.00000	-0.25519	8	17	0.00000	0.00789	100Zr
40	101	8	15	0.00000	-0.13926	8	15	0.00000	0.17608	101Zr
40	102	8	17	0.00000	-0.37952	8	17	0.00000	-0.12167	102Zr

### 8.3 Microscopic level densities

Besides the phenomenological models that are used in TALYS, there is also an option to employ more microscopic approaches. For the RIPL database, S. Goriely has calculated level densities from drip line to drip line on the basis of Hartree-Fock calculations [39] for excitation energies up to 150 MeV and for spin values up to  $I = 30$ . If **ldmodel 4** (p. 352), these tables with microscopic level densities can be read. Moreover, new energy-, spin- and parity-dependent nuclear level densities based on the microscopic combinatorial model have been proposed by Hilaire and Goriely [40]. The combinatorial model includes a detailed microscopic calculation of the intrinsic state density and collective enhancement. The only phenomenological aspect of the model is a simple damping function for the transition from spherical to deformed, see also Eq. (8.88). The calculations make coherent use of nuclear structure properties determined within the deformed Skyrme-Hartree-Fock-Bogoliubov framework. Level densities for more than 8500 nuclei are made available in tabular format, for excitation energies up to 200 MeV and for spin values up to  $J = 49$ . These level densities are used with **ldmodel 5**.

The most recent option, invoked with **ldmodel 6**, is based on temperature-dependent Hartree-Fock-Bogoliubov calculations using the Gogny force [41].

Since these microscopical level densities, which we will call  $\rho_{\text{HFM}}$ , have not been adjusted to experimental data, we add adjustment flexibility through a scaling function, i.e.

$$\rho(E_x, J, \pi) = \exp(c\sqrt{E_x - \delta})\rho_{\text{HFM}}(E_x - \delta, J, \pi) \quad (8.94)$$

where by default  $c = 0$  and  $\delta = 0$  (i.e. unaltered values from the tables). The “pairing shift”  $\delta$  simply implies obtaining the level density from the table at a different energy. The constant  $c$  plays a role similar to that of the level density parameter  $a$  of phenomenological models. Adjusting  $c$  and  $\delta$  together gives adjustment flexibility at both low and higher energies.

For both microscopic level density models, tables for level densities on top of the fission barriers are automatically invoked for **ldmodel 4**, **5** or **6**, when available in the structure database. For nuclides outside the tabulated microscopic database, the default Fermi gas model is used.

**Options 8.3.1** For microscopic level density models all characteristics like spin distributions, collective enhancement etc are all intrinsic in the estimation of the level density tables from the theory. Therefore, only a few parameters are available to adjust these tabulated levels densities: the keywords **ctable** (p. 360), **ptable** (p. 362), **cglobal** (p. 400), **pglobal** (p. 401). To study its effect, parity dependence can be enabled and disabled with **parity** (p. 386).

**Implementation 8.3.1** The two subdirectories *ground/hilaire/* and *ground/goriely/*, contain the tabulated microscopic level densities of Hilaire [40] and Goriely [39], respectively, also present in RIPL. For each isotope, there are first 4 comment lines, indicating the nucleus under consideration. Next, the excitation energy, temperature, number of cumulative levels, total level density, total state density and level density per spin are read in the format (f7.2,7.3,e10.2,31e9.2). Below is an example for the first energies of  $^{42}\text{Fe}$  from *ground/goriely/Fe.tab*

```
*****
* Z= 26 A= 42: Total and Spin-dependent Level Density [MeV-1] for Fe 42 *
*****
U[MeV]  T[MeV]  NCUMUL  RHO0BS  RHOTOT   J=0     J=1     J=2
  0.25   0.224   1.09E+00 7.33E-01 2.07E+00 2.44E-01 3.39E-01 1.26E-01
  0.50   0.302   1.33E+00 1.17E+00 4.14E+00 2.74E-01 4.83E-01 2.88E-01
```

```

0.75 0.365 1.71E+00 1.86E+00 7.60E+00 3.45E-01 6.76E-01 4.95E-01
1.00 0.419 2.29E+00 2.82E+00 1.31E+01 4.33E-01 9.05E-01 7.55E-01

```

Level densities for fission barriers are tabulated in *fission/hilaire/* and *fission/goriely/*. There is an *inner/* and *outer/* subdirectory, for the inner and outer barrier, respectively. Here is an example for the first few energy/spin points of  $^{230}\text{U}$  from *fission/goriely/inner/U.ld*. The format is the same as for the ground state, see above.

```

*****
* Z= 92 A=230: Total and Spin-dependent Level Density [MeV-1] at the inner
  saddle point c= 1.24 h= 0.00 a= 0.00 B= 3.80 MeV *
*****
U[MeV]  T[MeV]  NCUMUL  RHOBS  RHOTOT  J=0  J=1  J=2
0.25 0.162 1.11E+00 8.86E-01 5.92E+00 6.44E-02 1.64E-01 1.98E-01
0.50 0.190 1.51E+00 2.32E+00 1.81E+01 1.19E-01 3.20E-01 4.31E-01
0.75 0.209 2.56E+00 6.10E+00 5.25E+01 2.54E-01 7.01E-01 9.85E-01

```

In *density/ph/* microscopic particle-hole state densities are stored for 72 two-component ph combinations and 14 one-component ph combinations. The same energy grid and format as for total level densities is used. The particle-hole combinations are denoted as  $p_{\pi}h_{\pi}p_{\nu}h_{\nu}$ .

Here is an example for the first few energy/ph points of  $^{42}\text{Fe}$  from *density/ph/Fe.ph*.

```

*****
* Z= 26 A= 42 Particle-hole density [MeV-1] *
* Proton Fermi energy = -49.556 MeV *
* Neutron Fermi energy = -31.217 MeV *
*****
U[MeV]  0010  1000  0011  0020  0110  1001  1010
0.25 0.00E+00 0.00E+00 0.00E+00 0.00E+00 0.00E+00 0.00E+00 0.00E+00
0.50 0.00E+00 2.00E+00 0.00E+00 0.00E+00 0.00E+00 4.00E+00 0.00E+00
0.75 0.00E+00 2.00E+00 0.00E+00 0.00E+00 0.00E+00 4.00E+00 0.00E+00
1.00 0.00E+00 2.00E+00 0.00E+00 0.00E+00 0.00E+00 8.00E+00 0.00E+00
1.25 0.00E+00 2.00E+00 0.00E+00 0.00E+00 0.00E+00 8.00E+00 0.00E+00
1.50 0.00E+00 0.00E+00 0.00E+00 0.00E+00 0.00E+00 4.00E+00 0.00E+00

```

Finally, in *density/phjp/* microscopic particle-hole state densities, dependent on spin and parity, are tabulated for direct capture calculations.

## 8.4 Average resonance parameters

Neutron average resonance parameters are provided in the *talys/structure/resonances/* directory and stem from the the RIPL-2 database. As an example, below are the parameters for the Fe-isotopes from *resonances/Fe.res*. For each isotope, we have stored  $Z$ ,  $A$ , the experimental s-wave resonance spacing  $D_0$  in keV, its uncertainty, the experimental s-wave strength function  $S_0$  ( $\times 10^{-4}$ ), its uncertainty, the experimental total radiative width in eV and its uncertainty, nuclear symbol. The format is (2i4,2e9.2,2f5.2,2f9.5,20x,i4,a2).

```

26 55 1.80E+01 2.40E+00 6.90 1.80 1.80000 0.50000 55Fe
26 57 2.54E+01 2.20E+00 2.30 0.60 0.92000 0.41000 57Fe
26 58 6.50E+00 1.00E+00 4.70 1.10 1.90000 0.60000 58Fe
26 59 2.54E+01 4.90E+00 4.40 1.30 3.00000 0.90000 59Fe

```

**Sample case 8.1 Different level density models :  $n + {}^{99}\text{Tc}$** 

To demonstrate the variety of level density models that we have added recently to TALYS, we include a sample case in which 3 different models are compared. The results are given in Fig. 8.1 for the cumulative number of discrete levels and in Fig. 8.2 for the (n,p) cross section.

**Case a: Constant Temperature Model**

The input file is

```
#
# n-Tc099-ld1
#
# General
#
projectile n
element tc
mass 99
energy energies
#
# Parameters
#
outdensity y
filedensity y
ldmodel 1
```

This is the default calculation: TALYS use the local CTM level density for its calculations

**Case b: Back-shifted Fermi gas Model**

The input file is

```
#
# n-Tc099-ld2
#
# General
#
projectile n
element tc
mass 99
energy energies
#
# Parameters
#
outdensity y
filedensity y
ldmodel 2
```

**Case c: Hartree-Fock Model**

The input file is

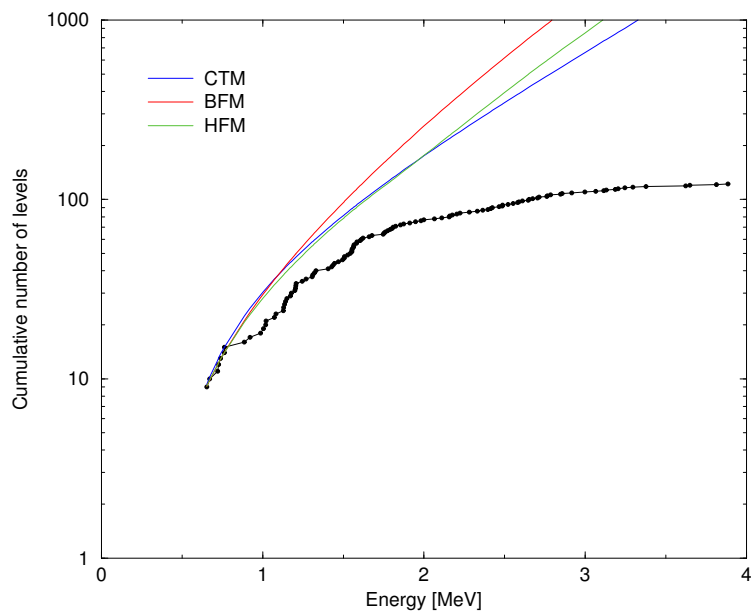


Figure 8.1: Cumulative number of discrete levels of  $^{99}\text{Tc}$  for different level density models.

```
#
# n-Tc099-ld5
#
# General
#
projectile n
element tc
mass 99
energy energies
#
# Parameters
#
outdensity y
filedensity y
ldmodel 5
```



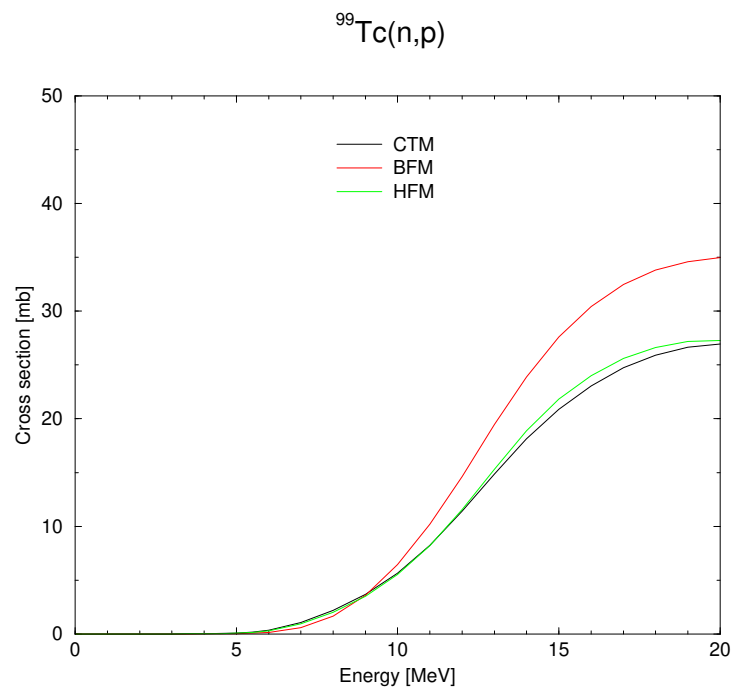


Figure 8.2:  $^{99}\text{Tc}(n,p)$  cross section for different level density models.



## 9. Reactions involving photons

### 9.1 Photon strength functions and transmission coefficients

Photon strength functions (PSF) are important for the description of the gamma transitions in nuclear reactions. The best known reaction channels which are governed by the PSF are the  $(n, \gamma)$  and  $(\gamma, n)$  reaction channels but the PSF can be regarded as an important universal entity in reaction modeling since gamma rays, in general, may accompany emission of any other emitted particle, and gamma transitions occur for both the continuum and discrete levels.

Before we describe the various PSF models implemented in TALYS, we first give the basic relations which establish their parameters, constraints and use in reaction models.

The transmission of a gamma ray at an energy  $E_\gamma$  is described by the PSF  $f_{X\ell}(E_\gamma)$ . Here  $X$  denotes either electric ( $E$ ) or magnetic ( $M$ ) radiation; and  $\ell$  is the radiation multipolarity. In general, dipole ( $\ell = 1$ ) radiation dominates over radiation of higher multipolarity for a given  $E_\gamma$ , e.g.  $f_{E1}(E_\gamma) \gg f_{E2}(E_\gamma)$ . Also  $E$  transitions are generally stronger than  $M$  transitions for a given multipolarity, e.g.  $f_{E1}(E_\gamma) \gg f_{M1}(E_\gamma)$

One distinguishes between

- upward PSF,  $\overrightarrow{f_{X\ell}}(E_\gamma)$  related to the cross section for gamma absorption such as for  $(\gamma, n)$  reactions, and is determined for  $E_\gamma > S_n$  (the neutron separation energy).
- downward PSF,  $\overleftarrow{f_{X\ell}}(E_\gamma)$  which determines the average width of the gamma decay, can be related to  $(n, \gamma)$  reactions, and is determined for  $E_\gamma < S_n$ .

The essential relation for the upward PSF is

$$\overrightarrow{f_{X\ell}}(E_\gamma) = K_{X\ell} \frac{\sigma_{X\ell}}{E_\gamma^{2\ell-1}}, \quad (9.1)$$

where,

$$K_{X\ell} = \frac{1}{(2\ell + 1)\pi^2 \hbar^2 c^2}, \quad (9.2)$$

Eq. (9.1) means that the shape, and parameters, of the PSF is determined by the (experimental) photoabsorption cross section  $\sigma_{X\ell}$  summed over all spins and parities. In practice, this means that the high-energy part of  $f_{E1}$  is determined by the  $(\gamma, n)$  cross section.

The downward PSF is related to the de-excitation of the nucleus. Like the particle transmission coefficients that emerge from the optical model, photon transmission coefficients enter the Hauser-Feshbach model for the determination of the competition of photon emission with emission of other particles.

The relation between the photon transmission coefficient and downward PSF is given by

$$T_{X\ell}(E_\gamma) = 2\pi \overleftarrow{f}_{X\ell}(E_\gamma) E_\gamma^{2\ell+1}. \quad (9.3)$$

The  $s$ -wave radiation width may be obtained by integrating the gamma-ray transmission coefficients over the density of final states that may be reached in the first step of the gamma-ray cascade. This leads to the essential relation for the downward PSF,

$$\frac{2\pi\Gamma_\gamma}{D_0} = \sum_J \sum_\Pi \sum_{X\ell} \sum_{I'=\lvert J-\ell \rvert}^{J+\ell} \sum_{\Pi'} \int_0^{S_n} dE_\gamma T_{X\ell}(E_\gamma) \rho(S_n - E_\gamma, I', \Pi') F(X, \Pi', \ell), \quad (9.4)$$

where  $D_0$  is the average resonance spacing and  $\rho$  is the level density. The  $J, \Pi$  sum is over the compound nucleus states with spin  $J$  and parity  $\Pi$  that can be formed with  $s$ -wave incident particles, and  $I', \Pi'$  denote the spin and parity of the final states. The multipole selection rules are  $F(E, \Pi', \ell) = 1$  if  $\Pi = \Pi'(-1)^\ell$ ,  $F(M, \Pi', \ell) = 1$  if  $\Pi = \Pi'(-1)^{\ell+1}$ , and 0 otherwise. It is understood that the integral over  $dE_\gamma$  includes a summation over discrete states.

The Brink hypothesis assumes the approximate equality of the two and from now on we will use  $f_{X\ell} = \overleftarrow{f}_{X\ell} = \overrightarrow{f}_{X\ell}$  in the notation. Also in TALYS no distinction is made. Hence, the ideal PSF for the whole energy range is thus characterized by a good description of the  $(\gamma, n)$  cross section for  $E_\gamma > S_n$  and simultaneously a good description of the  $(n, \gamma)$  cross section governed by  $E_\gamma > S_n$ , obeying Eq. (9.4). For the latter low-energy part, a better description of the PSF can be obtained by using experimental information from e.g.  $(p, p')$  reactions that feed gamma decay just below  $S_n$ , also known as the Oslo-method, and also from  $(\gamma, \gamma')$  reactions.

After the description of the various PSF models, we will outline how a proper normalization of PSF's according to the equations above can be obtained.

## 9.2 Models for photon strength functions

We have implemented various phenomenological and microscopic models for the photon strength function. The phenomenological models that we will discuss below are generally parameterized in terms of Lorentzian forms with giant resonance (GR) parameters:

- $\sigma_{X\ell}$ : strength of the giant resonance,
- $E_{X\ell}$ : energy of the giant resonance,
- $\Gamma_{X\ell}$ : width of the giant resonance.

For  $E1$ -transitions, GDR parameters for various individual nuclides exist. These are stored in the nuclear structure database of TALYS, as we will discuss below. Certain nuclides have a split GDR, i.e. a second set of Lorentzian parameters. For these cases, the incoherent sum of two strength functions is taken. For all  $X\ell$  transitions, systematic formulae compiled by Kopecky [6] for the resonance parameters are used. For  $E1$  transitions for which no nuclide-specific data exist, we use

$$\sigma_{E1} = 1.2 \times 120NZ / (A\pi\Gamma_{E1}) \text{ mb}, \quad E_{E1} = 31.2A^{-1/3} + 20.6A^{-1/6} \text{ MeV}, \quad \Gamma_{E1} = 0.026E_{E1}^{1.91} \text{ MeV}.$$

(9.5)

We also note that the PSF sometimes, for deformed nuclei, exhibits two peaks. In that case one often uses a second Lorentzian, and for the phenomenological models the total PSF for  $E1$  is then given by

$$f_{X\ell}(E_\gamma) = \sum_{i=1}^2 f_{X\ell}^i(E_\gamma, \sigma_{E1}^i, E_{E1}^i, \Gamma_{E1}^i) \quad (9.6)$$

For  $E2$  transitions we use

$$\sigma_{E2} = 0.00014Z^2 E_{E2} / (A^{1/3} \Gamma_{E2}) \text{ mb}, \quad E_{E2} = 63.A^{-1/3} \text{ MeV}, \quad \Gamma_{E2} = 6.11 - 0.012A \text{ MeV}. \quad (9.7)$$

For multipole radiation higher than  $E2$ , we use

$$\sigma_{E\ell} = 8.10^{-4} \sigma_{E(\ell-1)}, \quad E_{E\ell} = E_{E(\ell-1)} \quad \Gamma_{E\ell} = \Gamma_{E(\ell-1)}, \quad (9.8)$$

For  $M1$  transitions we use

$$f_{M1} = 1.58e - 9A^{0.47} \text{ at } 7 \text{ MeV}, \quad E_{M1} = 41.A^{-1/3} \text{ MeV}, \quad \Gamma_{M1} = 4 \text{ MeV}, \quad (9.9)$$

where Eq. (9.11) thus needs to be applied at 7 MeV to obtain the  $\sigma_{M1}$  value. For multipole radiation higher than  $M1$ , we use

$$\sigma_{M\ell} = 8.10^{-4} \sigma_{M(\ell-1)}, \quad E_{M\ell} = E_{M(\ell-1)} \quad \Gamma_{M\ell} = \Gamma_{M(\ell-1)}, \quad (9.10)$$

The microscopic PSF models are represented by tables, only for  $E1$  and  $M1$  radiation.

### 9.2.1 Brink-Axel model

The first, and oldest, model is the so-called Brink-Axel option [42, 43], in which a standard Lorentzian form describes the giant dipole resonance shape, i.e.

$$f_{X\ell}(E_\gamma) = K_{X\ell} \frac{\sigma_{X\ell} E_\gamma \Gamma_{X\ell}^2}{(E_\gamma^2 - E_{X\ell}^2)^2 + E_\gamma^2 \Gamma_{X\ell}^2}. \quad (9.11)$$

At present, we use the Brink-Axel option for all multipoles higher than 1. In TALYS, this model can be forced with **strength 2**.

### 9.2.2 Kopecky-Uhl model

For  $E1$  radiation, the default option used in TALYS is the generalized Lorentzian form of Kopecky and Uhl [44],

$$f_{E1}(E_\gamma, T) = K_{E1} \left[ \frac{E_\gamma \tilde{\Gamma}_{E1}(E_\gamma)}{(E_\gamma^2 - E_{E1}^2)^2 + E_\gamma^2 \tilde{\Gamma}_{E1}(E_\gamma)^2} + \frac{0.7 \Gamma_{E1} 4\pi^2 T^2}{E_{E1}^3} \right] \sigma_{E1} \Gamma_{E1}, \quad (9.12)$$

where the energy-dependent damping width  $\tilde{\Gamma}(E_\gamma)$  is given by

$$\tilde{\Gamma}_{E1}(E_\gamma) = \Gamma_{E1} \frac{E_\gamma^2 + 4\pi^2 T^2}{E_{E1}^2}, \quad (9.13)$$

and  $T$  is the nuclear temperature given by Kopecky et al [45]

$$T = \sqrt{\frac{E_n + S_n - \Delta - E_\gamma}{a(S_n)}}, \quad (9.14)$$

where  $S_n$  is the neutron separation energy,  $E_n$  the incident neutron energy,  $\Delta$  the pairing correction (see the chapter on level densities) and  $a$  the level density parameter at  $S_n$ . This model is used with **strength 1**.

### Pygmy resonances

In addition to the GDR contribution above, we can include Pygmy resonances by adding another contribution of the form (9.11) to the strength function. The Pygmy resonance parameters do not have a default.

### 9.2.3 Skyrme-Hartree-Fock BCS model

There are various microscopic options for  $E1$  radiation and the PSF's for these models have been stored as tables. Goriely and Khan [46] calculated gamma-ray strength functions according to the Skyrme-Hartree-Fock BCS model with QRPA (**strength 3**) and these are invoked with **strength 3**.

To bring these microscopical strength functions, which we will call  $f_{table}$ , at the same level of applicability as the phenomenological models, we add adjustment flexibility through scaling parameters as follows,

$$f_{E1}(E_\gamma) = f^{nor} f_{table}(E_\gamma + E_{shift}) \quad (9.15)$$

where by default  $f^{nor} = 1$  and  $E_{shift} = 0$  (i.e. unaltered values from the tables). As the name of these parameters imply,  $f^{nor}$  scales the entire PSF up or down, while the energy shift  $E_{shift}$  moves the PSF to the left or right and simply implies obtaining the PSF from the table at a somewhat different energy. Adjusting  $f^{nor}$  and  $E_{shift}$  together gives often enough adjustment flexibility. In TALYS, the associated keywords are **etable** (p. 420) and **ftable** (p. 422). Note that the adjustment flexibility of the (less physical) phenomenological models is somewhat higher, since in addition one may adjust the width of the PSF, as well as a possible second peak. Microscopic PSF's certainly predict such double peaks, but we have to accept the relative position of the two peaks as they come from the tables.

### 9.2.4 Skyrme-Hartree-Fock-Bogoliubov model

The Skyrme-Hartree-Fock-Bogoliubov model with QRPA by Goriely et al. [47] is implemented as (**strength 4**), see also RIPL [6]. Also here, Eq. (9.15) can be used to adjust the tables for fitting purposes.

### 9.2.5 Hybrid model

The hybrid model of Goriely [48] (**strength 5**), see also RIPL [6], is a Lorentzian model with energy and temperature dependent width which results in a different type of functional form at low energy than that of Kopecky-Uhl.

### 9.2.6 Temperature-dependent Skyrme-Hartree-Fock-Bogoliubov model

The Skyrme-Hartree-Fock-Bogoliubov model with QRPA by Goriely et al. [47] can be extended with temperature dependence and this is implemented as (**strength 6**). Also here, Eq. (9.15) can be used to adjust the tables for fitting purposes.

### 9.2.7 Temperature-dependent Relativistic Mean Field model

The Temperature-dependent RMF model by Daoutidis and Goriely [49] is implemented as (**strength 7**). Also here, Eq. (9.15) can be used to adjust the tables for fitting purposes.

### 9.2.8 Gogny-Hartree-Fock-Bogoliubov model

The Gogny-Hartree-Fock-Bogoliubov model with QRPA by Goriely et al. [50], based on the D1M version of the Gogny force is implemented as (**strength 8**). Also here, Eq. (9.15) can be used to adjust the tables for fitting purposes.

**Options 9.2.1** The keywords **strength** (p. 406) and **strengthM1** (p. 407) can be used to vary between different PSF models. The strength, width and energy parameters for the phenomenological GR PSF's for each radiation type and multipolarity can be explicitly provided by **sgr** (p. 409), **ggr** (p. 410), **egr** (p. 408), and for Pygmy resonances by **spr** (p. 437), **gpr** (p. 438), **epr** (p. 436). In addition to these nominal values, one may use **sgradjust** (p. 412), **ggradjust** (p. 413), **egradjust** (p. 411), **spradjust** (p. 440), **gpradjust** (p. 441), **epradjust** (p. 439), which all have a default value of 1. to deviate relative to the built-in default value for the GR and PR parameters.

To adjust the PSF normalization according to Eq. (9.4), the **gnorm** (p. 405) and **gamgamadjust** (p. 415) are available. For microscopic PSF models, **etable** (p. 420), and **ftable** (p. 422) are available to adjust tabulated values

Direct capture can be invoked with **racap** (p. 432).

**Implementation 9.2.1** GDR parameters are read in *gammapar.f* For E1 radiation, the Giant Dipole Resonance (GDR) parameters, stored in the *talys/structure/gamma/gdr/* directory, originate from the Beijing GDR compilation, as present in the RIPL database. As an example, below are the GDR parameters for the U-isotopes from *U.gdr*. For each isotope, we have stored Z, A, energy  $E_0$  in MeV, strength  $\sigma_0$  in mb, width of the GDR  $\Gamma_0$  in MeV and, if present, another energy, strength and width for the second peak, nuclear symbol. The format is (2i4,6f8.2,18x,i4,a2).

92	233	11.08	221.00	1.94	13.86	433.00	5.47	233U
92	234	11.13	371.00	2.26	13.94	401.00	4.46	234U
92	235	10.90	328.00	2.30	13.96	459.00	4.75	235U
92	236	10.92	271.00	2.55	13.78	415.00	4.88	236U
92	238	10.77	311.00	2.37	13.80	459.00	5.13	238U

The other subdirectories contain tabulated PSF's from various microscopic approaches. For example, the second subdirectory *gamma/hbfc/*, contains the tabulated microscopic gamma ray strength functions of Goriely et al [46], calculated according to Hartree-Fock BCS theory. For each isotope, there is first a line indicating the nucleus under consideration, read in the format (2(4x,i3)). Next, one line with units is given after which comes a table of excitation energies and strength functions, in the format (f9.3,e12.3). Below is an example for the first energies of  $^{110}\text{Ba}$  from *gamma/hbfc/Ba.psf*

```
Z= 56 A= 121 Ba
U[MeV] fE1 [mb/MeV]
0.100 5.581E-05
0.200 2.233E-04
```

```

0.300  5.028E-04
0.400  8.947E-04
0.500  1.400E-03
0.600  2.018E-03
0.700  2.752E-03
0.800  3.601E-03

```

Similarly *gamma/hbf/*, contains the tabulated microscopic gamma ray strength functions of Goriely et al [47], calculated according to Hartree-Fock QRPA theory. For each isotope, there is first a line indicating the nucleus under consideration, read in the format (2(4x,i3)). Next, one line with units is given after which comes a table of excitation energies and strength functions, in the format (f9.3,e12.3). Below is an example for the first energies of  $^{110}\text{Ba}$  from *gamma/hfb/Ba.psf*

```

Z= 56 A= 110
U[MeV]  fE1 [mb/MeV]
0.100  8.463E-03
0.200  9.116E-03
0.300  9.822E-03
0.400  1.058E-02
0.500  1.139E-02
0.600  1.226E-02
0.700  1.318E-02
0.800  1.416E-02

```

### Sample case 9.1 Photonuclear reactions: $g + {}^{90}\text{Zr}$

This sample case illustrates the capabilities of TALYS to simulate photonuclear reactions. We calculate the  $(\gamma, n)$  reaction on  ${}^{90}\text{Zr}$  as a function of incident energy, with default model parameters, and compare the result to experimental data. The following input file is used

```

#
# g-Zr090-xs
#
# General
#
projectile g
element zr
mass 90
energy energies

```

Fig. 9.1 displays the resulting production cross section of  ${}^{89}\text{Zr}$ , as obtained in file *rp040089.tot*.

### Sample case 9.2 Direct neutron capture for $n + {}^{89}\text{Y}$

For this sample case we use the input file

```

#

```



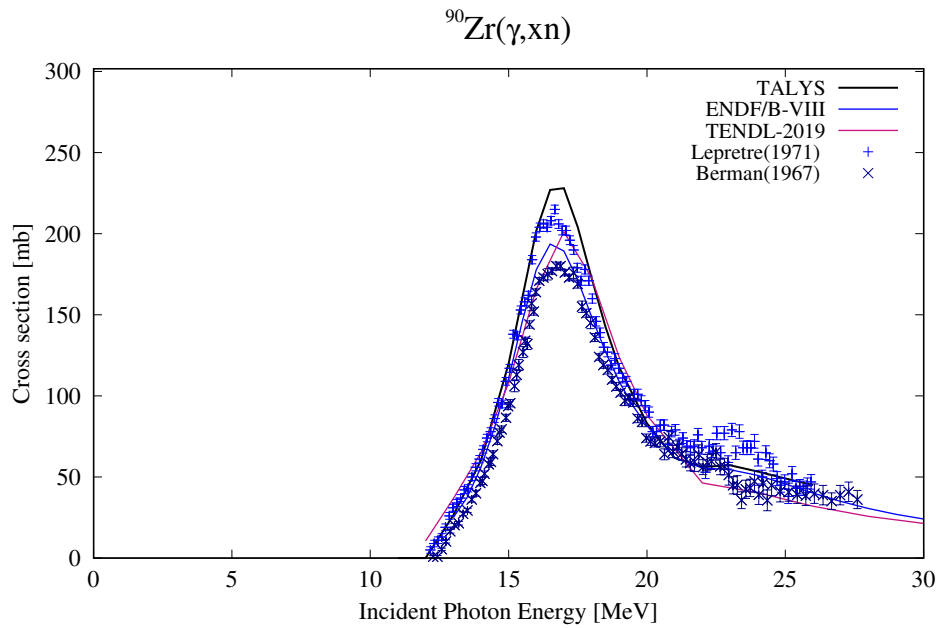


Figure 9.1: Photonuclear reaction on  $^{90}\text{Zr}$ . Experimental data are obtained from Berman et al. [51].

```
# n-Y089-dircap
#
# General
#
projectile n
element y
mass 89
energy n0-5.grid
#
# Parameters
#
gnorm 0.35
racap y
```

The contribution of the direct capture process can be found in the last column of *total.tot*. ■

### 9.3 Renormalization of gamma-ray strength functions

The  $\Gamma_\gamma$  values are, when available, read from our nuclear structure database. For nuclides for which no experimental value is available, we use an interpolation table by Kopecky [52] for  $40 < A < 250$ , the simple form

$$\Gamma_\gamma = 1593/A^2 \text{ eV}, \quad (9.16)$$

for  $A > 250$ , while we apply no gamma normalization for  $A < 40$ .

In the introduction we mentioned the essential relation Eq. (9.4). At sufficiently low incident neutron energies, the average radiative capture width  $\Gamma_\gamma$  is due entirely to the  $s$ -wave interaction, and

it is  $\Gamma_\gamma$  at the neutron separation energy  $S_n$  that is often used to normalize gamma-ray transmission coefficients, see Gardner [53]. Referring to Eq. (9.4), the normalization is then carried out as follows

$$\frac{2\pi\Gamma_\gamma}{D_0} = G_{norm} \sum_J \sum_\Pi \sum_{X\ell} \sum_{\ell'=|J-\ell|}^{J+\ell} \sum_{\Pi'} \int_0^{S_n} dE_\gamma T_{X\ell}(E_\gamma) \rho(S_n - E_\gamma, I', \Pi') f(X, \Pi', \ell), \quad (9.17)$$

i.e.  $G_{norm}$  is the normalization factor that enforces the equality (9.4). In practice, the transmission coefficients (9.3) are thus multiplied by  $G_{norm}$  before they enter the nuclear reaction calculation.  $G_{norm}$  can be specified by the user. The default is the value returned by Eq. (9.17). If  $G_{norm} = 1$  is specified, no normalization is carried out and strength functions purely determined from giant resonance parameters are taken. Other values can be entered for  $G_{norm}$ , e.g. for fitting of the neutron capture cross section. Normalisation per multipolarity can be performed by adjusting the  $\sigma_{X\ell}$  values in the input, see Part III.

## 9.4 Photoabsorption cross section

TALYS requires an estimate of the photo-absorption cross sections for models for photo-nuclear reactions and for pre-equilibrium gamma-ray emission. Obviously, for low energies this can be directly related to the GR PSF we have discussed above. However, at higher energies this needs to be complemented by an additional contribution. Following Chadwick et al. [54], a more general expression for the photo-absorption cross section is

$$\sigma_{abs}(E_\gamma) = \sigma_{GDR}(E_\gamma) + \sigma_{QD}(E_\gamma). \quad (9.18)$$

The GDR component is related to the strength functions outlined above. It is given by

$$\sigma_{GDR}(E_\gamma) = \sum_i \sigma_{E1} \frac{(E_\gamma \Gamma_{E1,i})^2}{(E_\gamma^2 - E_{E1,i}^2)^2 + E_\gamma^2 \Gamma_{E1,i}^2}, \quad (9.19)$$

where the parameters were specified in the previous Section. The sum over  $i$  is over the number of parts into which the GDR is split.

The quasi-deuteron component  $\sigma_{QD}$  is given by

$$\sigma_{QD}(E_\gamma) = L \frac{NZ}{A} \sigma_d(E_\gamma) f(E_\gamma). \quad (9.20)$$

Here,  $\sigma_d(E_\gamma)$  is the experimental deuteron photo-disintegration cross section, parameterized as

$$\sigma_d(E_\gamma) = 61.2 \frac{(E_\gamma - 2.224)^{3/2}}{E_\gamma^3}, \quad (9.21)$$

for  $E_\gamma > 2.224$  MeV and zero otherwise. The so-called Levinger parameter is  $L = 6.5$  and the Pauli-blocking function is approximated by the polynomial expression

$$f(E_\gamma) = 8.3714 \cdot 10^{-2} - 9.8343 \cdot 10^{-3} E_\gamma + 4.1222 \cdot 10^{-4} E_\gamma^2 - 3.4762 \cdot 10^{-6} E_\gamma^3 + 9.3537 \cdot 10^{-9} E_\gamma^4 \quad (9.22)$$

for  $20 < E_\gamma < 140$  MeV,

$$f(E_\gamma) = \exp(-73.3/E_\gamma) \quad (9.23)$$

for  $E_\gamma < 20$  MeV, and

$$f(E_\gamma) = \exp(-24.2348/E_\gamma) \quad (9.24)$$

for  $E_\gamma > 140$  MeV.

## 9.5 Direct capture

For light nuclides and (very) neutron-rich nuclides, the direct capture process may have a significant contribution. Therefore, a direct capture model was implemented, to be activated with the **racap** keyword (p. 432). More details may become available in future versions of this manual. For the moment we refer to Xu and Goriely [55].



## 10. Optical model

The central assumption underlying the optical model is that the complicated interaction between an incident particle and a nucleus can be represented by a complex mean-field potential, which divides the reaction flux into a part covering shape elastic scattering and a part describing all competing non-elastic channels. Solving the Schrödinger equation numerically with this complex potential yields a wealth of valuable information. First, it returns a prediction for the basic observables, namely the elastic angular distribution and polarisation, the reaction and total cross section and, for low energies, the  $s$ ,  $p$ -wave strength functions and the potential scattering radius  $R'$ . The essential value of a good optical model is that it can reliably predict these quantities for energies and nuclides for which no measurements exist. Also, the quality of the not directly observable quantities that are provided by the optical model has an equally important impact on the evaluation of the various reaction channels. Well-known examples are transmission coefficients, for compound nucleus and multi-step compound decay, and the distorted wave functions that are used for direct inelastic reactions and for transitions to the continuum that describe statistical multi-step direct reactions. Also, the reaction cross sections that are calculated with the optical model are crucial for the semi-classical pre-equilibrium models.

All optical model calculations are performed by ECIS-06, written by Raynal [5], which is used as a subroutine in TALYS.

### 10.1 Spherical OMP: Neutrons and protons

The default optical model potentials (OMP) used in TALYS are the local and global parameterisations of Koning and Delaroche [56].

The phenomenological OMP for nucleon-nucleus scattering,  $\mathcal{U}$ , is defined as:

$$\begin{aligned} \mathcal{U}(r, E) = & -\mathcal{V}_V(r, E) - i\mathcal{W}_V(r, E) - i\mathcal{W}_D(r, E) \\ & + \mathcal{V}_{SO}(r, E) \cdot \mathbf{l} \cdot \boldsymbol{\sigma} + i\mathcal{W}_{SO}(r, E) \cdot \mathbf{l} \cdot \boldsymbol{\sigma} + \mathcal{V}_C(r), \end{aligned} \quad (10.1)$$

where  $\mathcal{V}_{V,SO}$  and  $\mathcal{W}_{V,D,SO}$  are the real and imaginary components of the volume-central ( $V$ ), surface-central ( $D$ ) and spin-orbit ( $SO$ ) potentials, respectively.  $E$  is the LAB energy of the incident particle in MeV. All components are separated in energy-dependent well depths,  $V_V, W_V, W_D, V_{SO}$ , and  $W_{SO}$ , and energy-independent radial parts  $f$ , namely

$$\begin{aligned}
 \mathcal{V}_V(r, E) &= V_V(E)f(r, R_V, a_V), \\
 \mathcal{W}_V(r, E) &= W_V(E)f(r, R_V, a_V), \\
 \mathcal{W}_D(r, E) &= -4a_D W_D(E) \frac{d}{dr} f(r, R_D, a_D), \\
 \mathcal{V}_{SO}(r, E) &= V_{SO}(E) \left( \frac{\hbar}{m_\pi c} \right)^2 \frac{1}{r} \frac{d}{dr} f(r, R_{SO}, a_{SO}), \\
 \mathcal{W}_{SO}(r, E) &= W_{SO}(E) \left( \frac{\hbar}{m_\pi c} \right)^2 \frac{1}{r} \frac{d}{dr} f(r, R_{SO}, a_{SO}).
 \end{aligned} \tag{10.2}$$

The form factor  $f(r, R_i, a_i)$  is a Woods-Saxon shape

$$f(r, R_i, a_i) = (1 + \exp[(r - R_i)/a_i])^{-1}, \tag{10.3}$$

where the geometry parameters are the radius  $R_i = r_i A^{1/3}$ , with  $A$  being the atomic mass number, and the diffuseness parameters  $a_i$ . For charged projectiles, the Coulomb term  $\mathcal{V}_C$ , as usual, is given by that of a uniformly charged sphere

$$\begin{aligned}
 \mathcal{V}_C(r) &= \frac{Zze^2}{2R_C} \left( 3 - \frac{r^2}{R_C^2} \right), \quad \text{for } r \leq R_C \\
 &= \frac{Zze^2}{r}, \quad \text{for } r \geq R_C,
 \end{aligned} \tag{10.4}$$

with  $Z(z)$  the charge of the target (projectile), and  $R_C = r_C A^{1/3}$  the Coulomb radius.

The functional forms for the potential depths depend on  $(E - E_f)$ , where  $E_f$ , the Fermi energy in MeV, is defined as the energy halfway between the last occupied and the first unoccupied shell of the nucleus. For incident neutrons,

$$E_f^n = -\frac{1}{2}[S_n(Z, N) + S_n(Z, N + 1)], \tag{10.5}$$

with  $S_n$  the neutron separation energy for a nucleus with proton number  $Z$  and neutron number  $N$ , while for incident protons

$$E_f^p = -\frac{1}{2}[S_p(Z, N) + S_p(Z + 1, N)], \tag{10.6}$$

with  $S_p$  the proton separation energy. We use the mass table of the nuclear structure database to obtain the values of the separation energies.

Our OMP parameterisation for either incident neutrons or protons is

$$\begin{aligned}
V_V(E) &= v_1[1 - v_2(E - E_f) + v_3(E - E_f)^2 - v_4(E - E_f)^3] \\
W_V(E) &= w_1 \frac{(E - E_f)^2}{(E - E_f)^2 + (w_2)^2} \\
r_V &= \text{constant} \\
a_V &= \text{constant} \\
W_D(E) &= d_1 \frac{(E - E_f)^2}{(E - E_f)^2 + (d_3)^2} \exp[-d_2(E - E_f)] \\
r_D &= \text{constant} \\
a_D &= \text{constant} \\
V_{SO}(E) &= v_{so1} \exp[-v_{so2}(E - E_f)] \\
W_{SO}(E) &= w_{so1} \frac{(E - E_f)^2}{(E - E_f)^2 + (w_{so2})^2} \\
r_{SO} &= \text{constant} \\
a_{SO} &= \text{constant} \\
r_C &= \text{constant},
\end{aligned} \tag{10.7}$$

where  $E_f = E_f^n$  for incident neutrons and  $E_f = E_f^p$  for incident protons. Here  $E$  is the incident energy in the LAB system. This representation is valid for incident energies from 1 keV up to 200 MeV. Note that  $V_V$  and  $W_V$  share the same geometry parameters  $r_V$  and  $a_V$ , and likewise for the spin-orbit terms.

In general, all parameters appearing in Eq. (10.7) differ from nucleus to nucleus. When enough experimental scattering data of a certain nucleus is available, a so called local OMP can be constructed. TALYS retrieves all the parameters  $v_1, v_2$ , etc. of these local OMPs automatically from the nuclear structure and model parameter database, see the next Chapter, which contains the same information as the various tables of Koning and Delaroche [56]. If a local OMP parameterisation is not available in the database, the built-in global optical models are automatically used, which can be applied for any  $Z, A$  combination. A flag exists, **localomp** (p. 477) to overrule the local OMP by the global OMP. The global neutron OMP, validated for  $0.001 \leq E \leq 200$  MeV and  $24 \leq A \leq 209$ ,

is given by

$$\begin{aligned}
V_V(E) &= v_1^n [1 - v_2^n (E - E_f^n) + v_3^n (E - E_f^n)^2 - v_4^n (E - E_f^n)^3] \\
W_V(E) &= w_1^n \frac{(E - E_f^n)^2}{(E - E_f^n)^2 + (w_2^n)^2} \\
r_V &= 1.3039 - 0.4054A^{-1/3} \\
a_V &= 0.6778 - 1.487 \cdot 10^{-4} A \\
W_D(E) &= d_1^n \frac{(E - E_f^n)^2}{(E - E_f^n)^2 + (d_3^n)^2} \exp[-d_2^n (E - E_f^n)] \\
r_D &= 1.3424 - 0.01585A^{1/3} \\
a_D &= 0.5446 - 1.656 \cdot 10^{-4} A \\
V_{SO}(E) &= v_{so1}^n \exp[-v_{so2}^n (E - E_f^n)] \\
W_{SO}(E) &= w_{so1}^n \frac{(E - E_f^n)^2}{(E - E_f^n)^2 + (w_{so2}^n)^2} \\
r_{SO} &= 1.1854 - 0.647A^{-1/3} \\
a_{SO} &= 0.59,
\end{aligned} \tag{10.8}$$

where the units are in fm and MeV and the parameters for the potential depths and  $E_f^n$  are given in Table 10.1.

The global proton OMP is given by

$$\begin{aligned}
V_V(E) &= v_1^p [1 - v_2^p (E - E_f^p) + v_3^p (E - E_f^p)^2 - v_4^p (E - E_f^p)^3] \\
&\quad + \bar{V}_C \cdot v_1^p [v_2^p - 2v_3^p (E - E_f^p) + 3v_4^p (E - E_f^p)^2] \\
W_V(E) &= w_1^p \frac{(E - E_f^p)^2}{(E - E_f^p)^2 + (w_2^p)^2} \\
r_V &= 1.3039 - 0.4054A^{-1/3} \\
a_V &= 0.6778 - 1.487 \cdot 10^{-4} A \\
W_D(E) &= d_1^p \frac{(E - E_f^p)^2}{(E - E_f^p)^2 + (d_3^p)^2} \exp[-d_2^p (E - E_f^p)] \\
r_D &= 1.3424 - 0.01585A^{1/3} \\
a_D &= 0.5187 + 5.205 \cdot 10^{-4} A \\
V_{SO}(E) &= v_{so1}^p \exp[-v_{so2}^p (E - E_f^p)] \\
W_{SO}(E) &= w_{so1}^p \frac{(E - E_f^p)^2}{(E - E_f^p)^2 + (w_{so2}^p)^2} \\
r_{SO} &= 1.1854 - 0.647A^{-1/3} \\
a_{SO} &= 0.59 \\
r_C &= 1.198 + 0.697A^{-2/3} + 12.994A^{-5/3},
\end{aligned} \tag{10.9}$$

where the parameters for the potential depths,  $\bar{V}_C$  and  $E_f^p$  are given in Table 10.2. The functional form of the proton global OMP differs from the neutron global OMP only by the Coulomb correction term in  $V_V(E)$ .



$v_1^n$	$= 59.30 - 21.0(N - Z)/A - 0.024A$	MeV
$v_2^n$	$= 0.007228 - 1.48 \cdot 10^{-6}A$	MeV <sup>-1</sup>
$v_3^n$	$= 1.994 \cdot 10^{-5} - 2.0 \cdot 10^{-8}A$	MeV <sup>-2</sup>
$v_4^n$	$= 7 \cdot 10^{-9}$	MeV <sup>-3</sup>
$w_1^n$	$= 12.195 + 0.0167A$	MeV
$w_2^n$	$= 73.55 + 0.0795A$	MeV
$d_1^n$	$= 16.0 - 16.0(N - Z)/A$	MeV
$d_2^n$	$= 0.0180 + 0.003802/(1 + \exp[(A - 156.)/8.])$	MeV <sup>-1</sup>
$d_3^n$	$= 11.5$	MeV
$v_{so1}^n$	$= 5.922 + 0.0030A$	MeV
$v_{so2}^n$	$= 0.0040$	MeV <sup>-1</sup>
$w_{so1}^n$	$= -3.1$	MeV
$w_{so2}^n$	$= 160.$	MeV
$E_f^n$	$= -11.2814 + 0.02646A$	MeV

Table 10.1: Potential depth parameters and Fermi energy for the neutron global OMP of Eq. (10.8).

$v_1^p$	$= 59.30 + 21.0(N - Z)/A - 0.024A$	MeV
$v_2^p$	$= 0.007067 + 4.23 \cdot 10^{-6}A$	MeV <sup>-1</sup>
$v_3^p$	$= 1.729 \cdot 10^{-5} + 1.136 \cdot 10^{-8}A$	MeV <sup>-2</sup>
$v_4^p$	$= v_4^n$	MeV <sup>-3</sup>
$w_1^p$	$= 14.667 + 0.009629A$	MeV
$w_2^p$	$= w_2^n$	MeV
$d_1^p$	$= 16.0 + 16.0(N - Z)/A$	MeV
$d_2^p$	$= d_2^n$	MeV <sup>-1</sup>
$d_3^p$	$= d_3^n$	MeV
$v_{so1}^p$	$= v_{so1}^n$	MeV
$v_{so2}^p$	$= v_{so2}^n$	MeV <sup>-1</sup>
$w_{so1}^p$	$= w_{so1}^n$	MeV
$w_{so2}^p$	$= w_{so2}^n$	MeV
$E_f^p$	$= -8.4075 + 0.01378A$	MeV
$\bar{V}_C$	$= 1.73 \cdot Z \cdot A^{-1/3} / r_C$	MeV

Table 10.2: Potential depth parameters and Fermi energy for the proton global OMP of Eq. (10.9). The parameter values for neutrons are given in Table 10.1.  $\bar{V}_C$  appears in the Coulomb correction term  $\Delta V_C(E)$ , of the real central potential.

The spherical optical model described above provides the transmission coefficients, DWBA cross sections, total and elastic cross sections, etc., mentioned in the beginning of this section. For deformed nuclides, strongly coupled collective levels need to be included. This will be explained in Chapter 11 on direct reactions.

### 10.1.1 Extension to 1 GeV

To be able to predict the total, elastic and non-elastic cross sections up to 1 GeV, the OMP described above has been extended by Koning [57]. It is emphasized here that this was just done to test at which energy the validity of TALYS in predicting other (residual) cross section will fail. We are well aware of the fact that the usual Schrödinger picture of the OMP is valid up to about 180 MeV, and should then be taken over by a Dirac approach. Nevertheless, a functional form was constructed which leaves all KD03 parameter values below a joining energy  $E_J$ , at or around 200 MeV, unaltered while smoothly extending the energy dependence above  $E_J$ . This was only applied to the real,  $V_V$ , and imaginary,  $W_V$ , volume parts of the potential. For that, the KD03 OMP for neutrons below  $E_J$  reads [56],

$$\begin{aligned} V_V(E) &= v_1^n [1 - v_2^n (E - E_f^n) + v_3^n (E - E_f^n)^2 - v_4^n (E - E_f^n)^3] \\ W_V(E) &= w_1^n \frac{(E - E_f^n)^2}{(E - E_f^n)^2 + (w_2^n)^2}, \end{aligned} \quad (10.10)$$

where  $E_f^n$  is the Fermi energy. For  $V_V$ , we assume that the exponential decrease should continue beyond  $E_J$ . After all, the KD03 form of Eq. (10.10) for  $V_V$  is just a Taylor expansion of the exponential function, in which we gave ourselves the freedom to alter the individual coefficients  $v_1$ , etc. Also, following studies like those of Typel et al. [58] and Chiba et al. [59], we assume that it converges to a negative value  $V_\infty$ . Hence, the form chosen for  $E > E_J$  is

$$V_V(E) = V_\infty + b \cdot \exp(-c(E - E_f^n)). \quad (10.11)$$

We determine the new parameters  $b$  and  $c$  by calculating the value at  $E = E_f^n$ , giving

$$b = V_V(E_f^n) - V_\infty. \quad (10.12)$$

Hence,

$$V_V(E) = V_\infty + (V_V(E_f^n) - V_\infty) \exp(-c(E - E_f^n)). \quad (10.13)$$

Next,  $c$  can be determined by requiring that the high-energy potential is equal to that of the low energy expression Eq. (10.10) at the joining energy:

$$V_V(E_J) = V_\infty + (V_V(E_f^n) - V_\infty) \exp(-c(E_J - E_f^n)), \quad (10.14)$$

giving

$$c = -\frac{1}{(E_J - E_f^n)} \log\left(\frac{V_V(E_J) - V_\infty}{V_V(E_f^n) - V_\infty}\right). \quad (10.15)$$

For  $W_V$  it is expected that at high energies new absorption channels, such as pion production, emerge and that  $W_V$  will show another smooth increase as function of energy. Hence, the form of  $W_V$  for  $E > E_J$  is

$$W_V(E) = w_3^n \frac{(E - E_f^n)^4}{(E - E_f^n)^4 + (w_4^n)^4} + d, \quad (10.16)$$

where we find that a power of 4, instead of the usual 2, gives a better description of experimental data. Also here, a parameter  $d$  was added to ensure a value exactly equal to KD03 at  $E_J$ , i.e. at  $E = E_J$  we have

$$d = W_V(E_J) - w_3^n \frac{(E_J - E_f^n)^4}{(E_J - E_f^n)^4 + (w_4^n)^4} \quad (10.17)$$

In sum, we have the following simple extension of the KD03 OMP for  $E > E_J$ :

$$\begin{aligned} V_V(E) &= V_\infty + (V_V(E_f^n) - V_\infty) \\ &\quad * \exp\left(\frac{E - E_f^n}{E_J - E_f^n} * \log\left(\frac{V_V(E_J) - V_\infty}{V_V(E_f^n) - V_\infty}\right)\right) \\ W_V(E) &= W_V(E_J) - w_3^n \frac{(E_J - E_f^n)^4}{(E_J - E_f^n)^4 + (w_4^n)^4} \\ &\quad + w_3^n \frac{(E - E_f^n)^4}{(E - E_f^n)^4 + (w_4^n)^4} \end{aligned} \quad (10.18)$$

which joins smoothly with the KD03 expression of Eq. (10.10) for  $E < E_J$ . The following, preliminary, values were obtained from a fit to neutron total and proton non-elastic cross sections up to 1 GeV:

$$\begin{aligned} E_J &= 200. \\ V_\infty &= -30. \\ w_3^n &= 25. - 0.0417A \\ w_4^n &= 250. \end{aligned} \quad (10.19)$$

All parameters can be adjusted with keywords. In Eq. (10.18),  $V_V(E_f)$ ,  $V_V(E_J)$  and  $W_V(E_J)$  are obtained from Eq. (10.10). The above extension, and parameters, also hold for incident protons.

## 10.2 Spherical dispersive OMP: Neutrons

The theory of the nuclear optical model can be reformulated in terms of dispersion relations that connect the real and imaginary parts of the optical potential, and we have added an option in TALYS to take them into account. These dispersion relations are a natural result of the causality principle that a scattered wave cannot be emitted before the arrival of the incident wave. The dispersion component stems directly from the absorptive part of the potential,

$$\Delta\mathcal{V}(r, E) = \frac{\mathcal{P}}{\pi} \int_{-\infty}^{\infty} \frac{\mathcal{W}(r, E')}{E' - E} dE', \quad (10.20)$$

where  $\mathcal{P}$  denotes the principal value. The total real central potential can be written as the sum of a Hartree-Fock term  $\mathcal{V}_{HF}(r, E)$  and the total dispersion potential  $\Delta\mathcal{V}(r, E)$

$$\mathcal{V}(r, E) = \mathcal{V}_{HF}(r, E) + \Delta\mathcal{V}(r, E). \quad (10.21)$$

Since  $\mathcal{W}(r, E)$  has a volume and a surface component, the dispersive addition is,

$$\begin{aligned}\Delta\mathcal{V}(r,E) &= \Delta\mathcal{V}_V(r,E) + \Delta\mathcal{V}_D(r,E) \\ &= \Delta V_V(E)f(r,R_V,a_V) - 4a_D\Delta V_D(E)\frac{d}{dr}f(r,R_D,a_D)\end{aligned}\quad (10.22)$$

where the volume dispersion term is given by

$$\Delta V_V(E) = \frac{\mathcal{P}}{\pi} \int_{-\infty}^{\infty} \frac{W_V(E')}{E' - E} dE', \quad (10.23)$$

and the surface dispersion term is given by

$$\Delta V_D(E) = \frac{\mathcal{P}}{\pi} \int_{-\infty}^{\infty} \frac{W_D(E')}{E' - E} dE'. \quad (10.24)$$

Hence, the real volume well depth of Eq. (10.2) becomes

$$V_V(E) = V_{HF}(E) + \Delta V_V(E), \quad (10.25)$$

and the real surface well depth is

$$V_D(E) = \Delta V_D(E). \quad (10.26)$$

In general, Eqs. (10.23)-(10.24) cannot be solved analytically. However, under certain plausible conditions, analytical solutions exist. Under the assumption that the imaginary potential is symmetric with respect to the Fermi energy  $E_F$ ,

$$W(E_F - E) = W(E_F + E), \quad (10.27)$$

where  $W$  denotes either the volume or surface term, we can rewrite the dispersion relation as,

$$\Delta V(E) = \frac{2}{\pi} (E - E_F) \mathcal{P} \int_{E_F}^{\infty} \frac{W(E')}{(E' - E)^2 - (E - E_F)^2} dE', \quad (10.28)$$

from which it easily follows that  $\Delta V(E)$  is skew-symmetric around  $E_F$ ,

$$\Delta V(E + E_F) = -\Delta V(E - E_F), \quad (10.29)$$

and hence  $\Delta V(E_F) = 0$ . This can then be used to rewrite Eq. (10.20) as

$$\begin{aligned}\Delta V(E) &= \Delta V(E) - \Delta V(E_F) \\ &= \frac{\mathcal{P}}{\pi} \int_{-\infty}^{\infty} W(E') \left( \frac{1}{E' - E} - \frac{1}{E' - E_F} \right) dE' \\ &= \frac{E - E_F}{\pi} \int_{-\infty}^{\infty} \frac{W(E')}{(E' - E)(E' - E_F)} dE'.\end{aligned}\quad (10.30)$$

For the Hartree-Fock term we adopt the usual form for  $V_V(E)$  given in Eq. (10.7). The dispersion integrals for the functions for absorption can be calculated analytically and are included as options in ECIS-06. This makes the use of a dispersive optical model parameterization completely equivalent to that of a non-dispersive OMP: the dispersive contributions are calculated automatically once the OMP parameters are given. Upon comparison with a nondispersive parameterization, we find that  $v_1$  is rather different (as expected) and that  $r_V$ ,  $a_V$ ,  $v_2$ ,  $v_3$ ,  $w_1$  and  $w_2$  are slightly different. We have included dispersive spherical neutron OMP parameterization for about 70 nuclides (unpublished). They can be used with the keyword **dispersion y** (p. 476).

**Sample case 10.1 Different optical models : n +  $^{120}\text{Sn}$** 

To demonstrate the variety of optical models that we have added to TALYS, we include a sample case in which 4 OMP's for neutrons on  $^{120}\text{Sn}$  are compared. The results are given in Fig. 10.1 for the total cross section and in Fig.10.2 for the total inelastic cross section.

**Case a: Koning-Delaroche local potential**

The input file is

```
#
# n-Sn120-omp-KD03
#
# General
#
projectile n
element    sn
mass      120
energy    energies
outbasic  y
```

This is the default calculation: TALYS will find a local OMP in the structure database and will use it.

**Case b: Koning-Delaroche global potential**

The input file is

```
#
# n-Sn120-omp-KD03global
#
# General
#
projectile n
element    sn
mass      120
energy    energies
#
# Parameters
#
localomp  n
```

**Case c: Koning-Delaroche local dispersive potential**

The input file is

```
#
# n-Sn120-ompKD03disp
#
# General
#
```

```

projectile n
element    sn
mass      120
energy    energies
#
# Parameters
#
dispersion y

```

#### Case d: Bauge-Delaroche JLM potential

The input file is

```

#
# n-Sn120-omp-JLM
#
# General
#
projectile n
element    sn
mass      120
energy    energies
#
# Parameters
#
jlmomp y

```

### 10.3 Spherical OMP: Complex particles

For deuterons, tritons, Helium-3 and alpha particles, we use a simplification of the folding approach of Watanabe [60], see Madland [61]. We take the nucleon OMPs described in the previous section, either local or global, as the basis for these complex particle potentials.

#### 10.3.1 Deuterons

For deuterons, the real central potential depth at incident energy  $E$  is

$$V_V^{deuteron}(E) = V_V^{neutron}(E/2) + V_V^{proton}(E/2), \quad (10.31)$$

and similarly for  $W_V$  and  $W_D$ . For the spin-orbit potential depth we have

$$V_{SO}^{deuteron}(E) = (V_{SO}^{neutron}(E) + V_{SO}^{proton}(E))/2, \quad (10.32)$$

and similarly for  $W_{SO}$ . For the radius and diffuseness parameter of the real central potential we have

$$\begin{aligned} r_V^{deuteron} &= (r_V^{neutron} + r_V^{proton})/2, \\ a_V^{deuteron} &= (a_V^{neutron} + a_V^{proton})/2, \end{aligned} \quad (10.33)$$

and similarly for the geometry parameters of the other potentials.

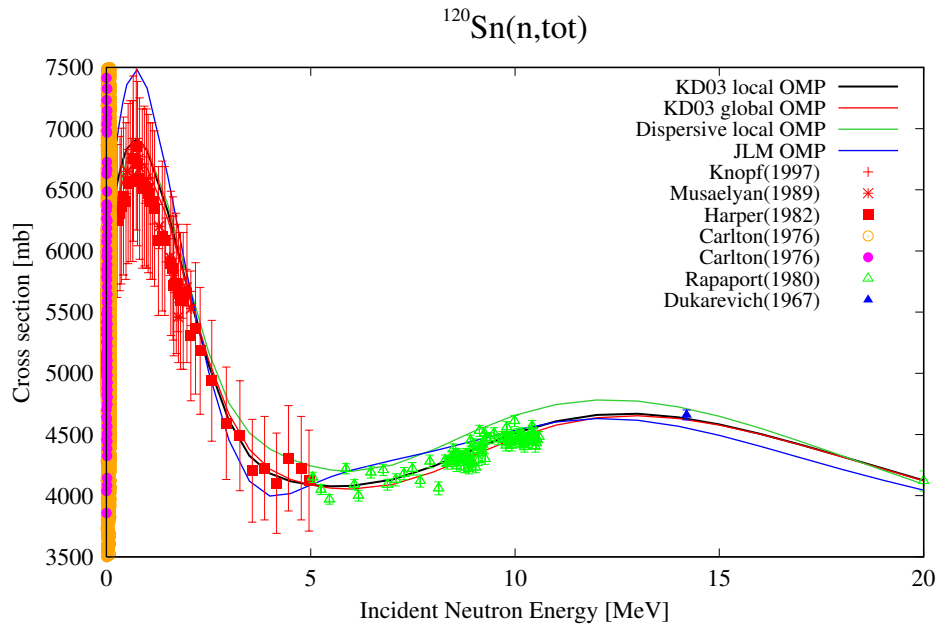


Figure 10.1: Total cross section for neutrons incident on  $^{120}\text{Sn}$  for different optical model potentials.

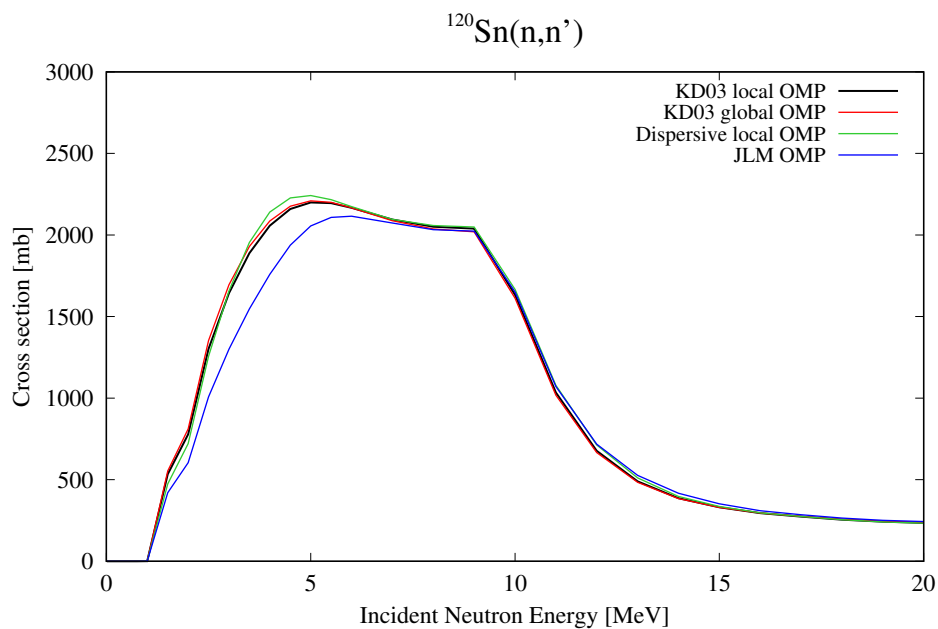


Figure 10.2: Total inelastic cross section for neutrons incident on  $^{120}\text{Sn}$  for different optical model potentials.

Note that several of these formulae are somewhat more general than necessary, since the nucleon potentials mostly have geometry parameters, and also potential depths such as  $V_{SO}$ , which are equal for neutrons and protons ( $a_D$  is an exception). The general formulae above have been implemented to also account for other potentials, if necessary.

We are well aware of the fact that others have constructed specific potentials for deuterons that may outperform the Watanabe-type potential described here. Therefore, we have also added the deuteron potentials of Daehnick et al. [62], Bojowald et al. [63], Han et al. [64], and Haixia An et al. [65] as options.

### 10.3.2 Tritons

For tritons, the real central potential depth at incident energy  $E$  is

$$V_V^{triton}(E) = 2V_V^{neutron}(E/3) + V_V^{proton}(E/3), \quad (10.34)$$

and similarly for  $W_V$  and  $W_D$ . For the spin-orbit potential depth we have

$$V_{SO}^{triton}(E) = (V_{SO}^{neutron}(E) + V_{SO}^{proton}(E))/6, \quad (10.35)$$

and similarly for  $W_{SO}$ . For the radius and diffuseness parameter of the real central potential we have

$$\begin{aligned} r_V^{triton} &= (2r_V^{neutron} + r_V^{proton})/3, \\ a_V^{triton} &= (2a_V^{neutron} + a_V^{proton})/3, \end{aligned} \quad (10.36)$$

and similarly for the geometry parameters of the other potentials.

### 10.3.3 Helium-3

For Helium-3, the real central potential depth at incident energy  $E$  is

$$V_V^{Helium-3}(E) = V_V^{neutron}(E/3) + 2V_V^{proton}(E/3), \quad (10.37)$$

and similarly for  $W_V$  and  $W_D$ . For the spin-orbit potential depth we have

$$V_{SO}^{Helium-3}(E) = (V_{SO}^{neutron}(E) + V_{SO}^{proton}(E))/6, \quad (10.38)$$

and similarly for  $W_{SO}$ . For the radius and diffuseness parameter of the real central potential we have

$$\begin{aligned} r_V^{Helium-3} &= (r_V^{neutron} + 2r_V^{proton})/3, \\ a_V^{Helium-3} &= (a_V^{neutron} + 2a_V^{proton})/3, \end{aligned} \quad (10.39)$$

and similarly for the geometry parameters of the other potentials.

### 10.3.4 Alpha particles

For alpha's, the real central potential depth at incident energy  $E$  is

$$V_V^{alphas}(E) = 2V_V^{neutron}(E/4) + 2V_V^{proton}(E/4), \quad (10.40)$$

and similarly for  $W_V$  and  $W_D$ . For the spin-orbit potential depth we have

$$V_{SO}^{alphas}(E) = W_{SO}^{alphas}(E) = 0. \quad (10.41)$$



For the radius and diffuseness parameter of the real central potential we have

$$\begin{aligned} r_V^{alphas} &= (r_V^{neutron} + r_V^{proton})/2, \\ a_V^{alphas} &= (a_V^{neutron} + a_V^{proton})/2, \end{aligned} \quad (10.42)$$

and similarly for the geometry parameters of the other potentials.

As extra options the alpha potentials by McFadden and Satchler[66] and Demetriou et al [67] are available.

All optical model parameters for complex particles can be altered via adjustable parameters through the **v1adjust** (p. 450) etc. keywords, with which the standard values can be multiplied. Also local energy-dependent adjustment of the geometry is possible as a last resort to fit data.

**Options 10.3.1** All phenomenological optical model parameters mentioned in this Chapter can be adjusted. The keywords **rvadjust** (p. 448), **avadjust** (p. 449), **v1adjust** (p. 450), **v2adjust** (p. 451), **v3adjust** (p. 452), **v4adjust** (p. 453), **rwadjust** (p. 454), **awadjust** (p. 455), **w1adjust** (p. 456), **w2adjust** (p. 457), **w3adjust** (p. 458), **w4adjust** (p. 459), **rvdadjust** (p. 460), **avdadjust** (p. 461), **rwdadjust** (p. 462), **awdadjust** (p. 463), **d1adjust** (p. 464), **d2adjust** (p. 465), **d3adjust** (p. 466), **rvsoadjust** (p. 471), **vso1adjust** (p. 467), **vso2adjust** (p. 468), **rwsoadjust** (p. 473), **awsoadjust** (p. 474), **wso1adjust** (p. 469), **wso2adjust** (p. 470), **rcadjust** (p. 475), are available to adjust the standard values that are built-in or given in the input. Dispersive OMP's can be invoked with **dispersion** (p. 476), one can switch between local and global OMP's with **localomp** (p. 477), and distinguish between incident and outgoing channel with **incadjust** (p. 494). As for most parameters, even energy-dependent adjustment of the geometry is possible as a last resort to fit data. Extensions to 1 GeV can be steered with **Vinfadjust** (p. 493) and **Ejoin** (p. 492). One may also provide a particular table of parameters for the KD03 potential by means of the keywords for local parameter files **optmod** (p. 496), **optmodfileN** (p. 497) and **optmodfileP** (p. 498). A full tabulated OMP file can be provided with **ompenergyfile** (p. 322).

**Implementation 10.3.1** The optical model parameters are stored in the *talys/structure/optical/* directory. All the parameters are based on one and the same functional form, see Section 10. There are two subdirectories: *neutron/* and *proton/*. For each isotope, optical potential parameters can be given. Per isotope, we have stored  $Z$ ,  $A$ , number of different optical potentials (2), character to determine coupled-channels potential, nuclear symbol with the format (3i4,3x,a1,58x,i4,a2). On the next line we read the OMP index (1: non-dispersive, 2: dispersive). Fermi energy and reduced Coulomb radius with the format (i4,f7.2,f8.3). On the next 3 lines we read the optical model parameters as defined in Section 10 with the following format

```
(2f8.3,f6.1,f10.4,f9.6,f6.1,f7.1) rv,av,v1,v2,v3,w1,w2
(2f8.3,f6.1,f10.4,f7.2) rvd,avd,d1,d2,d3
(2f8.3,f6.1,f10.4,f7.2,f6.1) rvso,avso,vso1,vso2,wso1,wso2
```

For neutrons, a dispersive potential may also be available. (These potentials have not (yet) been unpublished). In that case, another block of data is given per isotope, but now with an OMP index equal to 2, to denote the parameters for a dispersive potential. As an example, here are the parameters for the Fe-isotopes from *neutron/n-Fe.omp*.

```

1 -11.34  0.000
1.186  0.663  58.2  0.0071 0.000019  13.2  78.0
1.278  0.536  15.4  0.0223  10.90
1.000  0.580  6.1  0.0040  -3.1  160.0
2 -11.34  0.000
1.215  0.670  54.2  0.0077 0.000022  9.5  88.0
1.278  0.536  15.4  0.0223  10.90
1.000  0.580  6.1  0.0040  -3.1  160.0
26 56  2
1 -9.42  0.000
1.186  0.663  56.8  0.0071 0.000019  13.0  80.0
1.282  0.532  15.3  0.0211  10.90
1.000  0.580  6.1  0.0040  -3.1  160.0
2 -9.42  0.000
1.212  0.670  53.2  0.0079 0.000023  10.0  88.0
1.282  0.532  15.3  0.0211  10.90
1.000  0.580  6.1  0.0040  -3.1  160.0

```

56Fe

### Sample case 10.2 Different alpha-particle optical model potentials: alpha + $^{165}\text{Ho}$

To demonstrate the variety of (spherical) optical model potentials for alpha-particles available in TALYS, we include a sample case in which 4 OMPs for alpha-particles on  $^{165}\text{Ho}$  are compared. The results are given in Fig. 10.3 for the ( $\alpha$ ,n) reaction cross sections within the incident-energy range of most recent measured data [68] below the Coulomb barrier.

#### Case a: Watanabe folding approach with Koning-Delaroche nucleon potentials

The input file is

```

#
# a-Ho165-omp1
#
# General
#
projectile a
element Ho
mass 165
energy 12. 40. 0.5
#
# Model
#
alphaomp 1
#
# Output
#
outomp y

```

where the file energies consists of energies between 7 and 15.5 MeV with 0.5 MeV energy steps, corresponding to the energy range of the measured data. This is the default calculation. Fig. 10.3 displays the resulting ( $\alpha$ ,n) reaction cross sections for the target nucleus  $^{165}\text{Ho}$ , as

obtained in the file rp069168.tot.

**Case b: McFadden-Satchler (21) potential**

The input file, using the alphaomp keyword value of 2, is

```
#
# a-Ho165-omp2
#
# General
#
projectile a
element Ho
mass 165
energy 12. 40. 0.5
#
# Model
#
alphaomp 2
#
# Output
#
outomp y
```

**Case c: Demetriou, Grama and Goriely (67) double folding dispersive potential**

The input file, using the alphaomp keyword value of 5, is

```
#
# a-Ho165-omp5
#
# General
#
projectile a
element Ho
mass 165
energy 12. 40. 0.5
#
# Model
#
alphaomp 5
#
# Output
#
outomp y
```

**Case d: Avrigeanu et al. (69) potential**

The input file, using the alphaomp keyword additional value of 6, is

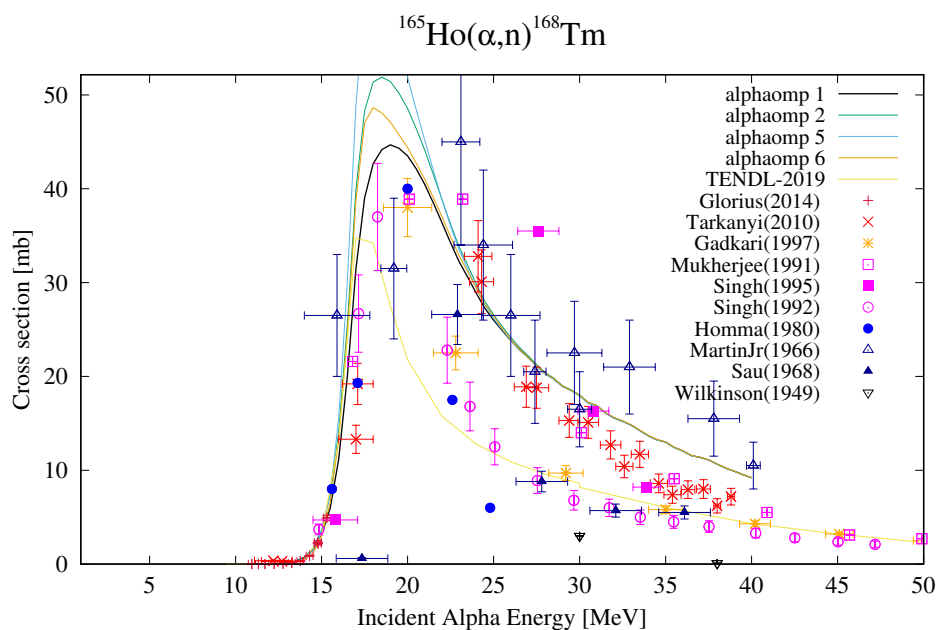


Figure 10.3:  $^{165}\text{Ho}(\alpha,n)^{168}\text{Tm}$  reaction cross sections for different OMP's.

```
#
# a-Ho165-omp6
#
# General
#
projectile a
element Ho
mass 165
energy 12. 40. 0.5
#
# Model
#
alphaomp 6
#
# Output
#
outomp y
```

where the file energies consists in addition of energies up to 33 MeV corresponding to all energy ranges which are considered in Table II of Ref. [69].

#### 10.4 Semi-microscopic optical model (JLM)

Besides the phenomenological OMP, it is also possible to perform TALYS calculations with the semi-microscopic nucleon-nucleus spherical optical model potential as described by Bauge et al. [70]. We have implemented Eric Bauge's MOM code [6] as a subroutine to perform so called

Jeukenne-Lejeune-Mahaux (JLM) OMP calculations. The optical model potential of Bauge et al. [70] and its extension to deformed and unstable nuclei has been widely tested [70, 71, 72, 73, 74] and produces predictions from  $A=30$  to 240 and for energies ranging from 10 keV up to 200 MeV. MOM stands for “Modèle Optique Microscopique” in French, or “Microscopic Optical Model” in English. In this version of TALYS, only spherical JLM OMP’s are included.

The MOM module reads the radial matter densities from the nuclear structure database and performs the folding of the Nuclear Matter (NM) optical model potential described by Bauge et al. [70] with the densities to obtain a local OMP. This is then put in the ECIS-06 routine to compute observables by solving the Schrödinger equation for the interaction of the projectile with the aforementioned OMP. The OMP’s are calculated by folding the target radial matter density with an OMP in nuclear matter based on the Brückner-Hartree-Fock work of Jeukenne, Lejeune and Mahaux [75, 76, 77, 78]. This NM OMP was then phenomenologically altered by Bauge et al. [70, 79] in order to improve the agreement of predicted finite nuclei observables with a large set of experimental data. These improvements consist in unifying the low and high energy parameterizations of the NM interaction in [79], and in studying the energy variations of the potential depth normalization factors [70]. These factors now include non-negligible normalizations of isovector components [70] that are needed in order to account simultaneously for (p,p) and (n,n) elastic scattering as well as (p,n)<sub>IAS</sub> quasi-elastic scattering. The final NM potential for a given nuclear matter density  $\rho = \rho_n - \rho_p$  and asymmetry  $\alpha = (\rho_n + \rho_p)/\rho$  reads

$$U_{NM}(E)_{\rho,\alpha} = \lambda_V(E) [V_0(\tilde{E}) \pm \lambda_{V1}(E)\alpha V_1(\tilde{E})] + i\lambda_W(E) [W_0(\tilde{E}) \pm \lambda_{W1}(E)\alpha W_1(\tilde{E})], \quad (10.43)$$

with  $E$  the incident nucleon energy,  $\tilde{E} = E - V_c$  (where  $V_c$  is the Coulomb field),  $V_0$ ,  $V_1$ ,  $W_0$ , and  $W_1$  the real isoscalar, real isovector, imaginary isoscalar, and imaginary isovector NM OMP components respectively, and  $\lambda_V$ ,  $\lambda_{V1}$ ,  $\lambda_W$ , and  $\lambda_{W1}$  the real (isoscalar+isovector), real isovector, imaginary, and imaginary isovector normalization factors respectively. The values of  $\lambda_V$ ,  $\lambda_{V1}$ ,  $\lambda_W$ , and  $\lambda_{W1}$  given in [70] read

$$\lambda_V(E) = 0.951 + 0.0008 \ln(1000E) + 0.00018 [\ln(1000E)]^2 \quad (10.44)$$

$$\lambda_W(E) = \left[ 1.24 - \left[ 1 + e^{\left(\frac{E-4.5}{2.9}\right)} \right]^{-1} \right] \left[ 1 + 0.06 e^{-\left(\frac{E-14}{3.7}\right)^2} \right] \times \left[ 1 - 0.09 e^{-\left(\frac{E-80}{78}\right)^2} \right] \left[ 1 + \left(\frac{E-80}{400}\right) \Theta(E-80) \right] \quad (10.45)$$

$$\lambda_{V1}(E) = 1.5 - 0.65 \left[ 1 + e^{\frac{E-1.3}{3}} \right]^{-1} \quad (10.46)$$

$$\lambda_{W1}(E) = \left[ 1.1 + 0.44 \left[ 1 + \left( e^{\frac{E-40}{50.9}} \right)^4 \right]^{-1} \right] \times \left[ 1 - 0.065 e^{-\left(\frac{E-40}{13}\right)^2} \right] \left[ 1 - 0.083 e^{-\left(\frac{E-200}{80}\right)^2} \right]. \quad (10.47)$$

with the energy  $E$  expressed in MeV. As stated in [70], in the 20 to 50 MeV range, the uncertainties related to  $\lambda_V$ ,  $\lambda_{V1}$ ,  $\lambda_W$ , and  $\lambda_{W1}$  are estimated to be 1.5%, 10%, 10%, and 10%, respectively. Outside this energy range, uncertainties are estimated to be 1.5 times larger.

In order to apply the NM OMP to finite nuclei an approximation had to be performed. This is the Local Density Approximation (LDA) where the local value of the finite nucleus OMP is taken to be the NM OMP value for the local finite nucleus density:  $U_{FN}(r) = U_{NM}(\rho(r))$ . Since this LDA produces potentials with too small rms radii, the improved LDA, which broadens the OMP with a Gaussian form factor (10.48), is introduced. In [79, 70] different prescriptions for the improved Local Density approximation are compared and LDA range parameters are optimized.

$$U_{FN}(r, E) = (t\sqrt{\pi})^{-3} \int \frac{U_{NM}(\rho(r'), E)}{\rho(r')} \exp(-|\vec{r} - \vec{r}'|^2/t_r^2) \rho(r') d\vec{r}', \quad (10.48)$$

with  $t$  the range of the Gaussian form factor. The  $t_r = 1.25$  fm and  $t_i = 1.35$  fm values were found [70] to be global optimal values for the real and imaginary ranges, respectively.

Finally, since no spin-orbit (SO) potential exists between a nucleon and NM, the Scheerbaum [80] prescription was selected in [79], coupled with the phenomenological complex potential depths  $\lambda_{v_{so}}$ , and  $\lambda_{w_{so}}$ . The SO potential reads

$$U_{n(p)}^{so}(r) = (\lambda_{v_{so}}(E) + i\lambda_{w_{so}}(E)) \frac{1}{r} \frac{d}{dr} \left( \frac{2}{3} \rho_{p(n)} + \frac{1}{3} \rho_{n(p)} \right), \quad (10.49)$$

with

$$\lambda_{v_{so}} = 130 \exp(-0.013 E) + 40 \quad (10.50)$$

and

$$\lambda_{w_{so}} = -0.2(E - 20). \quad (10.51)$$

All JLM OMP parameters can be altered via adjustable parameters through the **lvadjust**, **lwadjust** etc. keywords, with which the standard values can be multiplied.

**Options 10.4.1** Choices for JLM-based OMP's are available with **jlmomp** (p. 482) and **jlm-mode** (p. 483), **radialfile** (p. 491), **radialmodel** (p. 490), and the associated parameters **lvadjust** (p. 484), **lwadjust** (p. 485), **lv1adjust** (p. 486), **lw1adjust** (p. 487), **lvsoadjust** (p. 488), **lwsoadjust** (p. 489). with which the standard values can be multiplied. For alpha particle OMP's, the **alphaomp** (p. 479), **adepthcor** (p. 481) and **aradialcor** (p. 480) keywords are available, and for deuteron OMP's **deuteronomp** (p. 478).

**Implementation 10.4.1** For the calculation of the JLM OMP, both Stephane Goriely and Stephane Hilaire have provided radial matter densities from dripline to dripline. They are stored in *talys/structure/optical/jlm/*. As an example, below are the data of some of the Fe-isotopes as given in file *talys/structure/optical/jlm/goriely/Fe.rad*. First we give  $Z$ ,  $A$ , number of radii (lines), incremental step between radii, all in fm, with the format (2i4,i5,f7.3). Next we give the radius, and the radial densities for different deformations, first for protons and then for neutrons, with the format (f8.3,10(e12.5)),

```
26 56 200 0.100
0.100 8.04162E-02 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 9.06018E-02
0.200 8.02269E-02 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 9.05443E-02
```

```

0.300 7.98895E-02 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 9.04407E-02
0.400 7.94588E-02 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 9.03063E-02
0.500 7.89124E-02 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 9.01315E-02

```

Note that only the spherical components in this database are non-zero. In the files of Hilaire, see *talys/structure/optical/jlm/hilaire/*, there are also deformed components, but they will only become relevant when deformed JLM calculations are included in TALYS.

## 10.5 Systematics for non-elastic cross sections

Since the reaction cross sections for complex particles as predicted by the OMP have not been tested and rely on a relatively simple folding model, we added a possibility to estimate the non-elastic cross sections from empirical expressions. The adopted *tripathi.f* subroutine that provides this does not coincide with the published expression as given by Tripathi et al. [81], but checking our results with the published figures of Ref. [81] made us decide to adopt this empirical model as an option. We sometimes use it for deuterons up to alpha-particles. A high-quality OMP for complex particles would make this option redundant.

### Sample case 10.3 Local parameter adjustment for $n + {}^{93}\text{Nb}$

```

#
# n-Nb093-adjust
#
# General
#
projectile n
element nb
mass 93
energy 0.2 5. 0.2
outomp y
#
# Local parameter adjustment
#
rvadjust n 1.02 1. 3. 2. 1.10
avadjust n 1.02 1. 3. 2. 0.95

##### OPTICAL MODEL PARAMETERS #####

          neutron on 93Nb

Energy   V    rv    av    W    rw    aw    Vd    rvd    avd    Wd    rwd
0.001   51.02 1.239 0.676 0.14 1.239 0.676 0.00 1.274 0.534 3.32 1.274
0.002   51.02 1.239 0.676 0.14 1.239 0.676 0.00 1.274 0.534 3.32 1.274
.....
0.910   50.69 1.239 0.676 0.17 1.239 0.676 0.00 1.274 0.534 3.80 1.274
1.011   50.66 1.239 0.676 0.18 1.239 0.676 0.00 1.274 0.534 3.85 1.274

```

1.112	50.62	1.240	0.676	0.18	1.239	0.676	0.00	1.274	0.534	3.90	1.274
1.213	50.58	1.244	0.675	0.18	1.239	0.676	0.00	1.274	0.534	3.95	1.274
1.314	50.55	1.257	0.671	0.19	1.239	0.676	0.00	1.274	0.534	4.00	1.274
1.415	50.51	1.282	0.665	0.19	1.239	0.676	0.00	1.274	0.534	4.05	1.274
1.516	50.47	1.309	0.657	0.20	1.239	0.676	0.00	1.274	0.534	4.10	1.274
1.617	50.44	1.331	0.651	0.20	1.239	0.676	0.00	1.274	0.534	4.15	1.274
1.718	50.40	1.346	0.647	0.20	1.239	0.676	0.00	1.274	0.534	4.20	1.274
1.820	50.37	1.355	0.645	0.21	1.239	0.676	0.00	1.274	0.534	4.24	1.274
1.921	50.33	1.360	0.643	0.21	1.239	0.676	0.00	1.274	0.534	4.29	1.274
2.022	50.29	1.363	0.643	0.22	1.239	0.676	0.00	1.274	0.534	4.34	1.274
2.224	50.22	1.352	0.646	0.22	1.239	0.676	0.00	1.274	0.534	4.43	1.274
2.426	50.15	1.323	0.653	0.23	1.239	0.676	0.00	1.274	0.534	4.52	1.274
2.628	50.08	1.270	0.668	0.24	1.239	0.676	0.00	1.274	0.534	4.61	1.274
2.830	50.00	1.241	0.676	0.25	1.239	0.676	0.00	1.274	0.534	4.70	1.274
3.033	49.93	1.239	0.676	0.26	1.239	0.676	0.00	1.274	0.534	4.78	1.274
3.235	49.86	1.239	0.676	0.27	1.239	0.676	0.00	1.274	0.534	4.86	1.274
3.437	49.79	1.239	0.676	0.28	1.239	0.676	0.00	1.274	0.534	4.95	1.274

.....

Note that outside 1 and 3 MeV both  $r_v$  and  $a_v$  have their constant values while in between these energies the parameters vary according to the prescription given in Section 3.2.1. ■



# 11. Direct reactions

Various models for direct reactions are included in the program: DWBA for (near-)spherical nuclides, coupled-channels for deformed nuclides, the weak-coupling model for odd nuclei, and also a giant resonance contribution in the continuum. In all cases, TALYS drives the ECIS-06 code to perform the calculations. The results are presented as discrete state cross sections and angular distributions, or as contributions to the continuum.

## 11.1 Deformed nuclei: Coupled-channels

The formalism outlined in the OMP Chapter works, theoretically, for nuclides which are spherical and, in practice, for nuclides which are not too strongly deformed. In general, however, the more general coupled-channels method should be invoked to describe simultaneously the elastic scattering channel and the low-lying states which are, due to their collective nature, strongly excited by inelastic scattering. These collective excitations can be described as the result of static or dynamic deformations, which cause the homogeneous neutron-proton fluid to rotate or vibrate. The associated deformation parameters can be predicted from a (semi-)microscopic model or can be derived from an analysis of the experimental angular distributions.

The coupled-channels formalism for scattering and reaction studies is well known and will not be described in this manual. For a detailed presentation, we refer to Tamura [82]. We will only state the main aspects here to put the formalism into practice. Delaroche et al. [83] and Olsson et al. [84] have been used as guidance, as these papers explain clearly how to translate collective effects of various nuclides into ECIS calculations. In general various different channels, usually the ground state and several inelastic states, are included in a coupling scheme while the associated coupled equations are solved. In ECIS-06, this is done in a so called sequential iterative approach by Raynal [5]. Besides Ref. [5], Carlsson's lecture [85] is also recommended for more insight in the use of the ECIS code.

Various collective models for deformed nuclei exist. Note that the spherical optical model of Eq. (10.2) is described in terms of the nuclear radius  $R_i = r_i A^{1/3}$ . For deformed nuclei, this expression is generalized to include collective motions. Various models have been implemented in ECIS-06, which enables us to cover many nuclides of interest. We will describe the ones that can be invoked by TALYS. The collective models are automatically applied upon reading the deformation parameter database, see Section 7.2.

### 11.1.1 Symmetric rotational model

In the symmetric rotational model, the radii of the different terms of the OMP are expressed as

$$R_i = r_i A^{1/3} \left[ 1 + \sum_{\lambda=2,4,\dots} \beta_\lambda Y_\lambda^0(\Omega) \right], \quad (11.1)$$

where the  $\beta_\lambda$ 's are permanent, static deformation parameters, and the  $Y$  functions are spherical harmonics. The quadrupole deformation  $\beta_2$  plays a leading role in the interaction process. Higher order deformations  $\beta_\lambda$  (with  $\lambda = 4, 6, \dots$ ) are systematically smaller in magnitude than  $\beta_2$ . The inclusion of  $\beta_4$  and  $\beta_6$  deformations in coupled-channels calculations produces changes in the predicted observables, but in general, only  $\beta_2$  and  $\beta_4$  are important in describing inelastic scattering to the first few levels in a rotational band. For even-even nuclides like  $^{184}\text{W}$  and  $^{232}\text{Th}$ , the symmetric rotational model provides a good description of the lowest  $0^+, 2^+, 4^+$  (and often  $6^+, 8^+$ , etc.) rotational band. The nuclear model and parameter database of TALYS specifies whether a rotational model can be used for a particular nucleus, together with the included levels and deformation parameters. Either a deformation parameter  $\beta_\lambda$  or a deformation length  $\delta_\lambda = \beta_\lambda r_i A^{1/3}$  may be given. The latter one is generally recommended since it has more physical meaning than  $\beta_\lambda$  and should not depend on incident energy (while  $r_i$  may, in some optical models, depend on energy). We take  $\delta_\lambda$  equal for the three OMP components  $V_V, W_V$  and  $W_D$  and take the spin-orbit potential undeformed. The same holds for the vibrational and other collective models.

By default, TALYS uses the global optical model by Soukhovitskii et al. [86] for actinides. For rotational non-fissile nuclides, if no specific potential is specified through one of the various input methods, we take our local or global spherical potential and subtract 15% from the imaginary surface potential parameter  $d_1$ , if rotational or vibrational levels are included in the coupling scheme.

### 11.1.2 Harmonic vibrational model

A vibrational nucleus possesses a spherically symmetric ground state. Its excited states undergo shape oscillations about the spherical equilibrium model. In the harmonic vibrational model, the radii of the different terms of the OMP are expressed as

$$R_i = r_i A^{1/3} \left[ 1 + \sum_{\lambda\mu} \alpha_{\lambda\mu} Y_\lambda^\mu(\Omega) \right], \quad (11.2)$$

where the  $\alpha_{\lambda\mu}$  operators can be related to the coupling strengths  $\beta_\lambda$ , describing the vibration amplitude with multipolarity  $\lambda$ . Expanding the OMP to first or second order with this radius gives the OMP expressions for the excitation of one-phonon (first order vibrational model) and two-phonon (second order vibrational model) states [5]. For vibrational nuclei, the minimum number of states to couple is two. For even-even nuclei, we generally use the  $(0^+, 2^+)$  coupling, where the  $2^+$  level is a one-quadrupole phonon excitation. The level scheme of a vibrational nucleus (e.g.  $^{110}\text{Pd}$ )

often consists of a one-phonon state ( $2^+$ ) followed by a ( $0^+, 2^+, 4^+$ ) triplet of two-phonon states. When this occurs, all levels are included in the coupling scheme with the associated deformation length  $\delta_2$  (or deformation parameter  $\beta_2$ ). If the  $3^-$  and  $5^-$  states are strongly collective excitations, that is when  $\beta_3$  and  $\beta_5$  are larger than 0.1, these levels may also be included in the coupling scheme. An example is  $^{120}\text{Sn}$  [87], where the low lying ( $0^+, 2^+, 3^-, 4^+, 5^-$ ) states can all be included as one-phonon states in a single coupling scheme.

Again, if no specific potential is specified through one of the various input methods, we take our local or global spherical potential and subtract 15% from the imaginary surface potential parameter  $d_1$ .

### 11.1.3 Vibration-rotational model

For certain nuclides, the level scheme consists not only of one or more rotational bands, but also of one or more vibrational bands that can be included in the coupling scheme. An example is  $^{238}\text{U}$ , where many vibrational bands can be coupled. In Chapter 5 on discrete level information, it is explained how such calculations are automatically performed by TALYS. Depending on the number of levels included, the calculations can be time-consuming.

### 11.1.4 Asymmetric rotational model

In the asymmetric rotational model, in addition to the spheroidal equilibrium deformation, the nucleus can oscillate such that ellipsoidal shapes are produced. In this model the nucleus has rotational bands built on the statically deformed ground state and on the  $\gamma$ -vibrational state. The radius is now angular dependent,

$$R_i(\Theta) = r_i A^{1/3} \left[ 1 + \beta_2 \cos \gamma Y_2^0(\Omega) + \sqrt{\frac{1}{2}} \beta_2 \sin \gamma (Y_2^2(\Omega) + Y_2^{-2}(\Omega)) + \beta_4 Y_4^0(\Omega) \right], \quad (11.3)$$

where we restrict ourselves to a few terms. The deformation parameters  $\beta_2$ ,  $\beta_4$  and  $\gamma$  need to be specified.  $^{24}\text{Mg}$  is an example of a nucleus that can be analyzed with the asymmetric rotational model. Mixing between bands is not yet automated as an option in TALYS.

**Options 11.1.1** With **rotational** (p. 519) it is possible to disable a coupled-channel calculation for certain particles. With **autorot** (p. 527, TALYS looks for a coupled levels scheme in its database and automatically starts a coupled channels calculation. Coupled-channels calculations can be disabled with **spherical** (p. 518) Since coupled-channels calculations can be very time consuming, the keywords **ecissave** (p. 522), **inccalc** (p. 524), **eciscalc** (p. 523), are available to re-use ECIS information from the previous run. The complexity of the coupled-levels scheme can be steered with **maxrot** (p. 525) and **maxband** (p. 526).

#### Sample case 11.1 Coupled-channels rotational model: $n + ^{28}\text{Si}$

In this sample case, we consider spherical OMP and rotational coupled-channels calculations for the deformed nucleus  $^{28}\text{Si}$ .

##### Case a: Spherical optical model

In the first case, we treat  $^{28}\text{Si}$  as a spherical nucleus and include the first ( $2^+$ ), second ( $4^+$ ) and sixth ( $3^-$ ) level as weakly coupled levels, i.e. the cross sections are calculated with DWBA. The input file is

```

#
# n-Si028-cc
#
# General
#
projectile n
element si
mass 28
energy energies
#
# Parameters
#
spherical y
#
# Output
#
channels y
filechannels y

```

For the default calculation, TALYS will look in the *deformation/exp* database to see whether a coupling scheme is given. Since this is the case for  $^{28}\text{Si}$ , we have to put **spherical y** to enforce a spherical calculation.

#### Case b: Symmetric rotational model

In the second case, we include the first and second level of the ground state rotational band and the  $3^-$  state in the coupling scheme. This is accomplished with the input file

```

#
# n-Si028-cc-sym
#
# General
#
projectile n
element si
mass 28
energy energies
#
# Output
#
channels y
filechannels y

```

In Fig. 11.1, the calculated total inelastic scattering for cases a and b are plotted. ■

#### Sample case 11.2 Coupled-channels vibrational model: $n + ^{74}\text{Ge}$

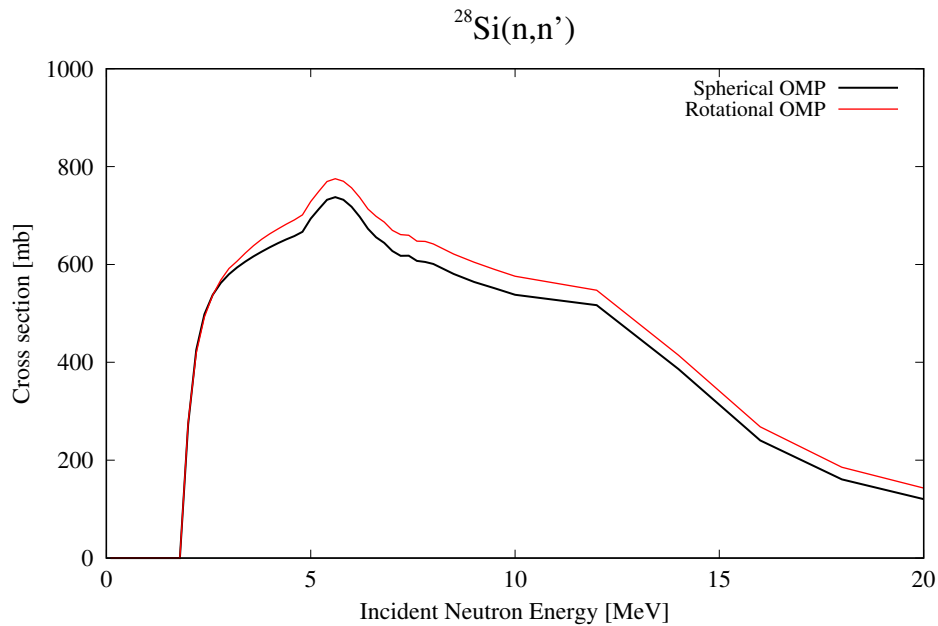


Figure 11.1: Total inelastic neutron scattering off  $^{28}\text{Si}$  for a spherical and a deformed OMP.

In this sample case we consider a neutron-induced reaction on the vibrational nucleus  $^{74}\text{Ge}$  which consists of a one-phonon state ( $2^+$ ) followed by a ( $0^+, 2^+, 4^+$ ) triplet of two-phonon states, and a  $3^-$  phonon state. The coupling scheme as stored in *structure/deformation/exp/z032* is automatically adopted. The following input file is used:

```
#
# n-Ge074-vib
#
# General
#
projectile n
element ge
mass 74
energy energies
#
# Output
#
outexcitation n
outdiscrete y
filediscrete 1
```

In Fig. 11.2, the calculated inelastic scattering to the first discrete state is plotted. ■

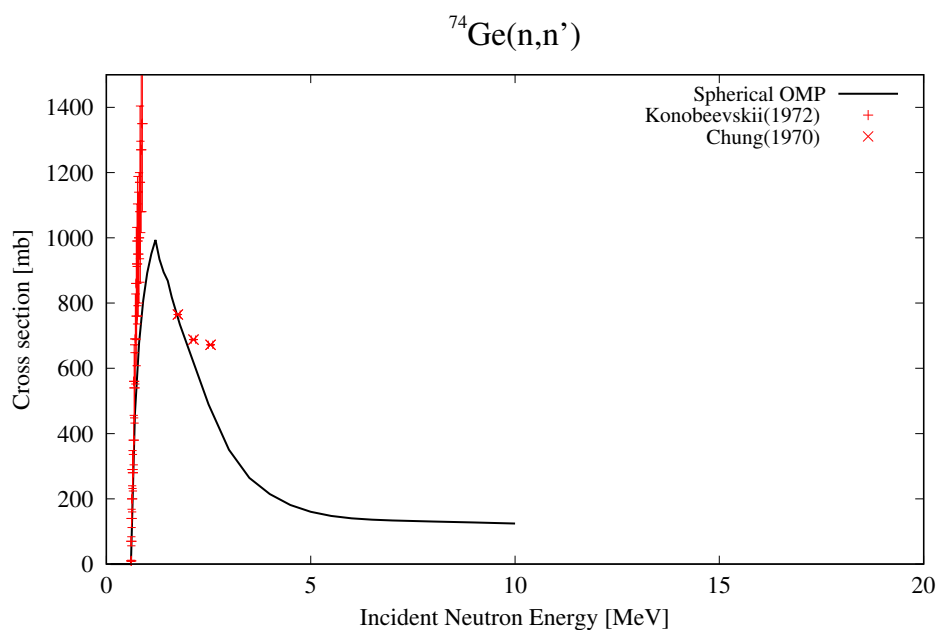


Figure 11.2: Inelastic scattering to the first discrete state of  $^{74}\text{Ge}$ .

## 11.2 Distorted Wave Born Approximation

The Distorted Wave Born Approximation (DWBA) is only valid for small deformations. Until the advent of the more general coupled-channels formalism, it was the commonly used method to describe inelastic scattering, for both weakly and strongly coupled levels. Nowadays, we see DWBA as a first order vibrational model for small deformation, with only a single iteration to be performed for the coupled-channels solution. (See, however Satchler [88] for the exact difference between this so called distorted wave method and DWBA). The interaction between the projectile and the target nucleus is modeled by the derivative of the OMP for elastic scattering times a strength parameter. The latter, the deformation parameter  $\beta_\lambda$ , is then often used to vary the overall magnitude of the cross section (which is proportional to  $\beta_\lambda^2$ ).

In TALYS, we use DWBA

- if a deformed OMP is not available. This applies for the spherical OMPs mentioned in the previous Section, which are all based on elastic scattering observables only. Hence, if we have not constructed a coupled-channels potential, TALYS will automatically use (tabulated or systematical) deformation parameters for DWBA calculations.
- if a deformed OMP is used for the first excited states only. For the levels that do not belong to that basic coupling scheme, e.g. for the many states at somewhat higher excitation energy, we use DWBA with (very) small deformation parameters.

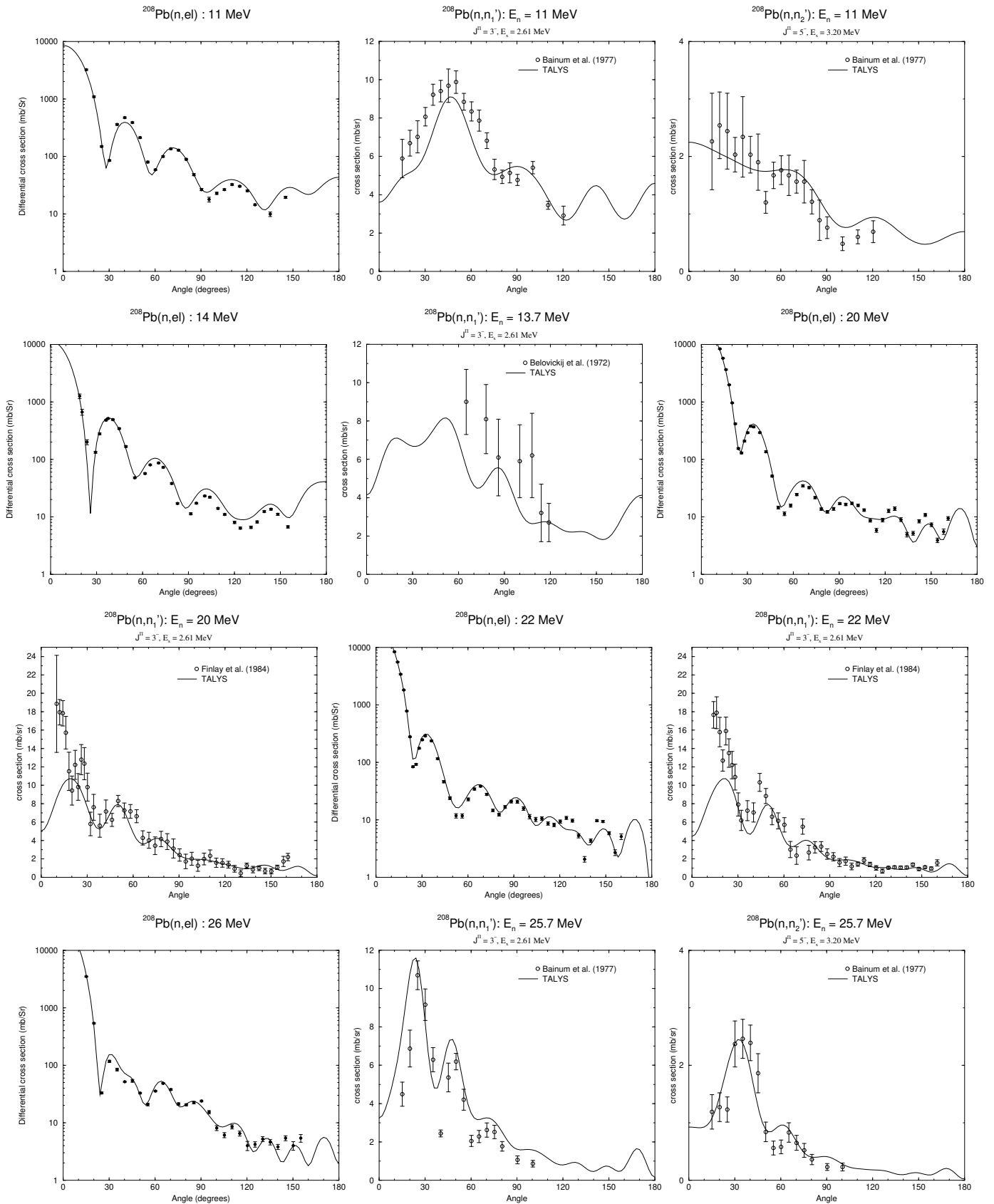
**Options 11.2.1** By default, OMP calculations will take deformation parameters and coupled level schemes into account, but with **spherical** (p. 518) this can be disabled and turned into a spherical OMP calculation.

**Sample case 11.3** Spherical optical model and DWBA:  $n + ^{208}\text{Pb}$

Three types of optical model calculations are included in the set of sample cases. In this first one, we treat  $^{208}\text{Pb}$  as a spherical nucleus and request calculations for the elastic angular distributions and inelastic angular distributions at a few incident energies. This is accomplished with the input file

```
#
# n-Pb208-omp
#
# General
#
projectile n
element pb
mass 208
energy energies
#
# Avoid unnecessary calculations and output
#
ejectiles n
preequilibrium n
compound n
maxZ 0
maxN 0
bins 5
fileresidual n
filetotal n
#
# Output
#
outangle y
fileelastic y
fileangle 1
fileangle 2
```

where the file *energies* consists of the energies 11., 13.7, 20., 22., and 25.7 MeV (for which experimental data exists). The keyword *fileelastic y* has created the files *nn011.000ang.L00*, etc. which contain the elastic scattering angular distribution and are compared with experimental data in Fig. 11.3. With **fileangle 1** and **fileangle 2** we have created the files *nn010.000ang.L01*, etc. with the inelastic scattering angular distribution to the first and second discrete state. These are also plotted in Figs. 11.3. Note that the keywords in the middle block (**ejectiles n** up to **filetotal n**) have been added to avoid a full calculation of all the cross sections. For the present sample case we assume that only elastic scattering and DWBA angular distributions are of interest, so we economize on output options, number of bins, ejectiles and nuclides that can be reached. Obviously, for reliable results for all observables this middle block would have to be deleted. See also sample case (1f) for obtaining more specific information from the output. ■

Figure 11.3: Elastic and inelastic scattering angular distributions between 11 and 26 MeV for  $^{208}\text{Pb}$ .



### 11.3 Odd nuclei: Weak coupling

Direct inelastic scattering off odd- $A$  nuclei can be described by the weak-coupling model [89], which assumes that a valence particle or hole interacts only weakly with a collective core excitation. Hence the model implies that the nucleon inelastic scattering by the odd- $A$  nucleus is very similar to that by the even core alone, i.e. the angular distributions have a similar shape. Let  $L$  be the spin of the even core state, and  $J_0$  and  $J$  the spin of the ground and excited state, respectively, of the odd- $A$  nucleus, resulting from the angular momentum coupling. Then, the spins  $J$  of the multiplet states in the odd- $A$  nucleus range from  $|L - J_0|$  to  $(L + J_0)$ . If the strength of the inelastic scattering is characterized by the square of the deformation parameters  $\beta_{L,J}^2$ , then the sum of all  $\beta_{L,J}^2$  or  $\sigma(E)$  for the transitions in the odd- $A$  nucleus should be equal to the value  $\beta_L^2$  or  $\sigma(E)$  for the single transition in the even core nucleus:

$$\sum_J \beta_{L,J}^2 = \beta_L^2, \quad \sum_J \sigma_{J_0 \rightarrow J} = \sigma_{0 \rightarrow L}, \quad (11.4)$$

where the symbol  $0 \rightarrow L$  indicates a transition between the ground state to the excited state with spin  $L$  in the even core nucleus. The deformation parameters  $\beta_{L,J}^2$  are now given by

$$\beta_{L,J}^2 = \frac{2J+1}{(2J_0+1)(2L+1)} \beta_L^2. \quad (11.5)$$

In practice, the DWBA cross sections are calculated for the real mass of the target nucleus and at the exact excitation energies of the odd- $A$  states, but for the even-core spin  $L$  and with deformation parameters  $\beta_{L,J}$ .

We stress that our weak-coupling model is not full-proof. First of all, there are always two choices for the even-even core. The default used in TALYS (by means of the keyword **core -1**) is to use the even-even core obtained by subtracting a nucleon, but the other choice (**core 1**), to obtain the even-even core by adding a nucleon, may sometimes be more appropriate. The next uncertainty is the choice of levels in the odd- $A$  core. We select the levels that are closest to the excitation energy of the even-spin state of the even-even core. Again, this may not always be the most appropriate choice. A future option is to designate these levels manually.

**Options 11.3.1** With **core** (p. 545), one can choose between the  $A - 1$  or  $A + 1$  nucleus for the even-even core.

**Implementation 11.3.1** In *weakcoupling.f90* the strength is distributed over the odd-spin levels.

### 11.4 Giant resonances

The high-energy part of the continuum spectra are generally described by pre-equilibrium models. These models are essentially of a single-particle nature. Upon inspection of continuum spectra, some structure in the high-energy tail is observed that can not be accounted for by the smooth background of the single-particle pre-equilibrium model. For example, many 14 MeV inelastic neutron spectra show a little hump at excitation energies around 6-10 MeV. This structure is due to collective excitations of the nucleus that are known as giant resonances [90, 91]. We use a macroscopic, phenomenological model to describe giant resonances in the inelastic channel. For each multipolarity, an energy weighted sum rule (EWSR)  $S_\ell$  applies,

$$S_\ell = \sum_i E_{\ell,i} \beta_{\ell,i}^2 = 57.5 A^{-5/3} l(2l+1) \text{ MeV}, \quad (11.6)$$

where  $E_{\ell,i}$  is the excitation energy of the  $i$ -th state with multipolarity  $\ell$ . The summation includes all the low-lying collective states, for each  $\ell$ , that have already been included in the coupled-channels or DWBA formalism. The EWSR thus determines the remaining collective strength that is spread over the continuum. Our treatment is phenomenological in the sense that we perform a DWBA calculation with ECIS-06 for each giant resonance state and spread the cross section over the continuum with a Gaussian distribution. The central excitation energy for these states and the spreading width is different for each multipolarity and has been empirically determined. For the giant monopole resonance (GMR) EWSR we have

$$S_0 = 23A^{-5/3} \text{ MeV}, \quad (11.7)$$

with excitation energy and width

$$E_{0,GMR} = 18.7 - 0.025A \text{ MeV}, \quad \Gamma_{GMR} = 3 \text{ MeV}. \quad (11.8)$$

The EWSR for the giant quadrupole resonance (GQR) is

$$S_2 = 575A^{-5/3} \text{ MeV}, \quad (11.9)$$

with

$$E_{0,GQR} = 65A^{-1/3} \text{ MeV}, \quad \Gamma_{GQR} = 85A^{-2/3} \text{ MeV}. \quad (11.10)$$

The EWSR for the giant octupole resonance is

$$S_3 = 1208A^{-5/3} \text{ MeV}, \quad (11.11)$$

which has a low-energy (LEOR) and a high-energy (HEOR) component. Following Kalbach [91], we assume

$$S_{3,LEOR} = 0.3S_3, \quad S_{3,HEOR} = 0.7S_3, \quad (11.12)$$

with excitation energy and width

$$E_{0,LEOR} = 31A^{-1/3} \text{ MeV}, \quad \Gamma_{LEOR} = 5 \text{ MeV}, \quad (11.13)$$

and

$$E_{0,HEOR} = 115A^{-1/3} \text{ MeV}, \quad \Gamma_{HEOR} = 9.3 - A/48 \text{ MeV}, \quad (11.14)$$

respectively. We also take as width for the actual Gaussian distribution  $\Gamma_{Gauss} = 0.42\Gamma_\ell$ .

The contribution from giant resonances is automatically included in the total inelastic cross section. The effect is most noticeable in the single- and double-differential energy spectra.

**Options 11.4.1** With **giantresonance** (p. 532) we can disable or enable the giant resonance contribution.

**Implementation 11.4.1** In *giant.f90* the Giant Resonance contribution is implemented.

## 12. Pre-equilibrium reactions

It is now well-known that the separation of nuclear reaction mechanisms into direct and compound is too simplistic. As Fig. 4.2 shows, the cross section as predicted by the pure compound process is too small with respect to measured continuum spectra, and the direct processes described in the previous section only excite the discrete levels at the highest outgoing energies. Furthermore, the measured angular distributions in the region between direct and compound are anisotropic, indicating the existence of a memory-preserving, direct-like reaction process. Apparently, as an intermediate between the two extremes, there exists a reaction type that embodies both direct- and compound-like features. These reactions are referred to as *pre-equilibrium*, *precompound* or, when discussed in a quantum-mechanical context, *multi-step processes*. Pre-equilibrium emission takes place after the first stage of the reaction but long before statistical equilibrium of the compound nucleus is attained. It is imagined that the incident particle step-by-step creates more complex states in the compound system and gradually loses its memory of the initial energy and direction. Pre-equilibrium processes cover a sizable part of the reaction cross section for incident energies between 10 and (at least) 200 MeV. Pre-equilibrium reactions have been modeled both classically and quantum-mechanically and both are included in TALYS.

### 12.1 Exciton model

In the exciton model (see Refs. [92, 93, 94] for extensive reviews), the nuclear state is characterized at any moment during the reaction by the total energy  $E^{tot}$  and the total number of particles above and holes below the Fermi surface. Particles ( $p$ ) and holes ( $h$ ) are indiscriminately referred to as excitons. Furthermore, it is assumed that all possible ways of sharing the excitation energy between different particle-hole configurations with the same exciton number  $n = p + h$  have equal a-priori probability. To keep track of the evolution of the scattering process, one merely traces the temporal development of the exciton number, which changes in time as a result of intranuclear

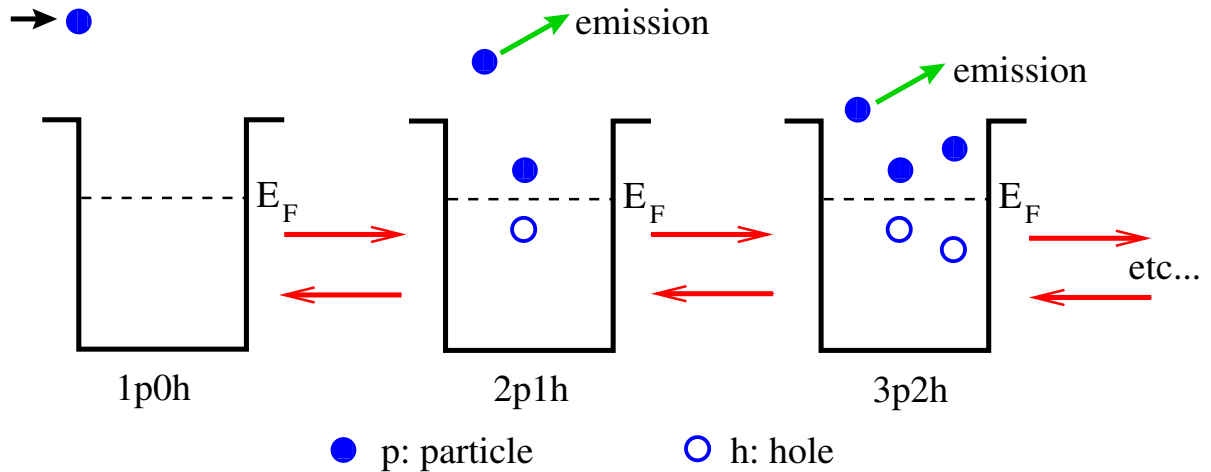


Figure 12.1: Reaction flow in exciton model

two-body collisions. The basic starting point of the exciton model is a time-dependent master equation, which describes the probability of transitions to more and less complex particle-hole states as well as transitions to the continuum (emission). Upon integration over time, the energy-averaged emission spectrum is obtained. These assumptions make the exciton model amenable for practical calculations. The price to be paid, however, is the introduction of a free parameter, namely the average matrix element of the residual two-body interaction, occurring in the transition rates between two exciton states. When this matrix element is properly parameterized, a very powerful model is obtained.

Qualitatively, the equilibration process of the excited nucleus is imagined to proceed as follows, see Fig. 12.1. After entering the target nucleus, the incident particle collides with one of the nucleons of the Fermi sea, with depth  $E_F$ . The formed state with  $n = 3$  (2p1h), in the case of a nucleon-induced reaction, is the first that is subject to particle emission, confirming the picture of the exciton model as a compound-like model rather than a direct-like model. Subsequent interactions result in changes in the number of excitons, characterized by  $\Delta n = +2$  (a new particle-hole pair) or  $\Delta n = -2$  (annihilation of a particle-hole pair) or  $\Delta n = 0$  (creation of a different configuration with the same exciton number). In the first stage of the process, corresponding to low exciton numbers, the  $\Delta n = +2$  transitions are predominant. Apart from transitions to more complex or less complex exciton states, at any stage there is a non-zero probability that a particle is emitted. Should this happen at an early stage, it is intuitively clear that the emitted particle retains some “memory” of the incident energy and direction: the hypothesis of a fully equilibrated compound nucleus is not valid. This phase is called the pre-equilibrium phase, and it is responsible for the experimentally observed high-energy tails and forward-peaked angular distributions. If emission does not occur at an early stage, the system eventually reaches a (quasi-) equilibrium. The equilibrium situation, corresponding to high exciton numbers, is established after a large number of interactions, i.e. after a long lapse of time, and the system has “forgotten” about the initial state. Accordingly, this stage may be called the compound or evaporation stage. Hence, in principle the exciton model enables to compute the emission cross sections in a unified way, without introducing adjustments between equilibrium and pre-equilibrium contributions. However, in practical cases it turns out that it is simpler and even more accurate to distinguish between a pre-equilibrium and an equilibrium phase and to perform the latter with the usual Hauser-Feshbach formalism. This is the approach followed in TALYS.

Two versions of the exciton model are implemented in TALYS: The default is the two-component model in which the neutron or proton types of particles and holes are followed throughout the reaction. We describe this model first, and then discuss the simpler, and more generally known, one-component model which is also implemented as an option. The following Section contains basically the most important equations of the recent exciton model study of [92].

### 12.1.1 Two-component exciton model

In the following reaction equations, we use a notation in which  $p_\pi$  ( $p_\nu$ ) is the proton (neutron) particle number and  $h_\pi$  ( $h_\nu$ ) the proton (neutron) hole number. From this, we define the proton exciton number  $n_\pi = p_\pi + h_\pi$  and the neutron exciton number  $n_\nu = p_\nu + h_\nu$ . From this, we can construct the charge-independent particle number  $p = p_\pi + p_\nu$ , the hole number  $h = h_\pi + h_\nu$  and the exciton number  $n = n_\pi + n_\nu$ .

The temporal development of the system can be described by a master equation, describing the gain and loss terms for a particular class of exciton states, see [92]. Integrating the master equation over time up to the equilibration time  $\tau_{eq}$  yields the mean lifetime of the exciton state  $\tau$  that can be used to calculate the differential cross section [95]. The primary pre-equilibrium differential cross section for the emission of a particle  $k$  with emission energy  $E_k$  can then be expressed in terms of  $\tau$ , the composite-nucleus formation cross section  $\sigma^{CF}$ , and an emission rate  $W_k$ ,

$$\begin{aligned} \frac{d\sigma_k^{PE}}{dE_k} &= \sigma^{CF} \sum_{p_\pi=p_\pi^0}^{p_\pi^{\max}} \sum_{p_\nu=p_\nu^0}^{p_\nu^{\max}} W_k(p_\pi, h_\pi, p_\nu, h_\nu, E_k) \tau(p_\pi, h_\pi, p_\nu, h_\nu) \\ &\times P(p_\pi, h_\pi, p_\nu, h_\nu), \end{aligned} \quad (12.1)$$

where the factor  $P$  represents the part of the pre-equilibrium population that has survived emission from the previous states and now passes through the  $(p_\pi, h_\pi, p_\nu, h_\nu)$  configurations, averaged over time. Expressions for all quantities appearing in this expression will be detailed in the rest of this Section. The initial proton and neutron particle numbers are  $p_\pi^0 = Z_p$ , and  $p_\nu^0 = N_p$ , respectively with  $Z_p$  ( $N_p$ ) the proton (neutron) number of the projectile. For any exciton state in the reaction process,  $h_\pi = p_\pi - p_\pi^0$  and  $h_\nu = p_\nu - p_\nu^0$ , so that for primary pre-equilibrium emission the initial hole numbers are  $h_\pi^0 = h_\nu^0 = 0$ . For e.g. a neutron-induced reaction, the initial exciton number is given by  $n^0 = n_\nu^0 = 1$  ( $0p_\pi 0h_\pi 1p_\nu 0h_\nu$ ), but only pre-equilibrium gamma emission can occur from this state (nucleon emission from this state is essentially elastic scattering and this is already covered by the optical model). Particle emission only occurs from  $n = 3$  ( $2p1h$ ) and higher exciton states. We use a hardwired value of  $p_\pi^{\max} = p_\nu^{\max} = 6$  as the upper limit of the summation, see [92]. We use the never-come-back approximation, i.e. throughout the cascade one neglects the interactions that decrease the exciton number, although the adopted solution of Eq. (12.1) does include transitions that convert a proton particle-hole pair into a neutron pair and vice versa. The maximum values  $p_\pi^{\max}$  and  $p_\nu^{\max}$  thus entail an automatic separation of the pre-equilibrium population and the compound nucleus population. The latter is then handled by the more adequate Hauser-Feshbach mechanism. We now discuss the various ingredients of Eq. (12.1).

#### A. Reaction cross sections

The basic feeding term for pre-equilibrium emission is the composite formation cross section  $\sigma^{CF}$ , which is given by

$$\sigma^{CF} = \sigma_{\text{reac}} - \sigma_{\text{direct}}, \quad (12.2)$$

where the reaction cross section  $\sigma_{\text{reac}}$  is directly obtained from the optical model and  $\sigma_{\text{direct}}$  is the sum of the cross sections for direct reactions to discrete states  $\sigma^{\text{disc.direct}}$  as defined in Eq. (4.18), and for giant resonances, see Section 11.4.

### B. Emission rates and particle-hole state densities

The emission rate  $W_k$  has been derived by Cline and Blann [96] from the principle of microreversibility, and can easily be generalized to a two-component version as shown by Dobes and Betak [97]. The emission rate for an ejectile  $k$  with relative mass  $\mu_k$  and spin  $s_k$  is

$$W_k(p_\pi, h_\pi, p_\nu, h_\nu, E_k) = \frac{2s_k + 1}{\pi^2 \hbar^3} \mu_k E_k \sigma_{k,\text{inv}}(E_k) \times \frac{\omega(p_\pi - Z_k, h_\pi, p_\nu - N_k, h_\nu, E^{\text{tot}} - E_k)}{\omega(p_\pi, h_\pi, p_\nu, h_\nu, E^{\text{tot}})}, \quad (12.3)$$

where  $\sigma_{k,\text{inv}}(E_k)$  is the inverse reaction cross section, again calculated with the optical model,  $Z_k$  ( $N_k$ ) is the charge (neutron) number of the ejectile and  $E^{\text{tot}}$  is the total energy of the composite system.

For the particle-hole state density  $\omega(p_\pi, h_\pi, p_\nu, h_\nu, E_x)$  we use the expression of Běťák and Dobeš [97, 98]. Their formula is based on the assumption of equidistant level spacing and is corrected for the effect of the Pauli exclusion principle and for the finite depth of the potential well. The two-component particle-hole state density is

$$\omega(p_\pi, h_\pi, p_\nu, h_\nu, E_x) = \frac{g_\pi^{n_\pi} g_\nu^{n_\nu}}{p_\pi! h_\pi! p_\nu! h_\nu! (n-1)!} (U - A(p_\pi, h_\pi, p_\nu, h_\nu))^{n-1} \times f(p, h, U, V), \quad (12.4)$$

where  $g_\pi$  and  $g_\nu$  are the single-particle state densities,  $A$  the Pauli correction,  $f$  the finite well function, and  $U = E_x - P_{p,h}$  with  $P_{p,h}$  Fu's pairing correction [99],

$$P_{p,h} = \Delta - \Delta \left[ 0.996 - 1.76 \left( \frac{n}{n_{\text{crit}}} \right)^{1.6} / \left( \frac{E_x}{\Delta} \right)^{0.68} \right]^2 \\ \text{if } E_x/\Delta \geq 0.716 + 2.44 \left( \frac{n}{n_{\text{crit}}} \right)^{2.17}, \\ = \Delta \text{ otherwise}, \quad (12.5)$$

with

$$n_{\text{crit}} = 2gT_{\text{crit}} \ln 2, \quad (12.6)$$

where  $T_{\text{crit}} = 2\sqrt{\Delta/\frac{1}{4}g}/3.5$  and  $g = g_\pi + g_\nu$ . The pairing energy  $\Delta$  for total level densities is given by

$$\Delta = \chi \frac{12}{\sqrt{A}}, \quad (12.7)$$

where here  $A$  is the mass number and  $\chi = 0, 1$  or  $2$  for odd-odd, odd or even-even nuclei. The Pauli correction term is given by

$$A(p_\pi, h_\pi, p_\nu, h_\nu) = \frac{[\max(p_\pi, h_\pi)]^2}{g_\pi} + \frac{[\max(p_\nu, h_\nu)]^2}{g_\nu} - \frac{p_\pi^2 + h_\pi^2 + p_\pi + h_\pi}{4g_\pi} - \frac{p_\nu^2 + h_\nu^2 + p_\nu + h_\nu}{4g_\nu}. \quad (12.8)$$

For the single-particle state densities we take

$$g_\pi = Z/15, \quad g_\nu = N/15, \quad (12.9)$$

which is, through the relationship  $g = a\pi^2/6$ , in line with the values for our total level density parameter  $a$ , see Eq. (8.13), and also provides a globally better description of spectra than the generally adopted  $g = A/13$ .

The finite well function  $f(p, h, E_x, V)$  accounts for the fact that a hole cannot have an energy below that of the bottom of the potential well depth  $V$ . It is given by

$$f(p, h, E_x, V) = 1 + \sum_{i=1}^h (-1)^i \binom{h}{i} \left[ \frac{E_x - iV}{E_x} \right]^{n-1} \Theta(E_x - iV), \quad (12.10)$$

where  $\Theta$  is the unit step function. Note that  $f$  is different from 1 only for excitation energies greater than  $V$ . In the original version of Běťák and Dobeš,  $V$  is given by the depth  $E_f$  of the Fermi well. This was generalized by Kalbach [100, 91] to obtain an effective method to include surface effects in the first stage of the interaction, leading to a harder pre-equilibrium spectrum. For the first stage the maximum depth of the hole should be significantly reduced, since in the surface region the potential is shallower than in the interior. This automatically leaves more energy to be adopted by the excited particle, yielding more emission at the highest outgoing energies. We use the following functional form for  $V$  in terms of the projectile energy  $E_p$  and the mass  $A$ ,

$$\begin{aligned} V &= 22 + 16 \frac{E_p^4}{E_p^4 + (450/A^{1/3})^4} \text{ MeV for } h = 1 \text{ and incident protons,} \\ V &= 12 + 26 \frac{E_p^4}{E_p^4 + (245/A^{1/3})^4} \text{ MeV for } h = 1 \text{ and incident neutrons,} \\ V &= E_f = 38 \text{ MeV for } h > 1, \end{aligned} \quad (12.11)$$

see Koning and Duijvestijn [92] for a further justification of this parameterisation.

### C. Lifetimes

The lifetime  $\tau$  of exciton state  $(p_\pi, h_\pi, p_\nu, h_\nu)$  in Eq. (12.1) is defined as the inverse sum of the total emission rate and the various internal transition rates,

$$\begin{aligned} \tau(p_\pi, h_\pi, p_\nu, h_\nu) &= [\lambda_\pi^+(p_\pi, h_\pi, p_\nu, h_\nu) + \lambda_\nu^+(p_\pi, h_\pi, p_\nu, h_\nu) \\ &+ \lambda_{\pi\nu}^0(p_\pi, h_\pi, p_\nu, h_\nu) + \lambda_{\nu\pi}^0(p_\pi, h_\pi, p_\nu, h_\nu) + W(p_\pi, h_\pi, p_\nu, h_\nu)]^{-1}, \end{aligned} \quad (12.12)$$

where  $\lambda_\pi^+$  ( $\lambda_\nu^+$ ) is the internal transition rate for proton (neutron) particle-hole pair creation,  $\lambda_{\pi\nu}^0$  ( $\lambda_{\nu\pi}^0$ ) is the rate for the conversion of a proton (neutron) particle-hole pair into a neutron (proton) particle-hole pair, and  $\lambda_\pi^-$  ( $\lambda_\nu^-$ ) is the rate for particle-hole annihilation. These transition rates will be discussed in Sec. 12.1.1. The total emission rate  $W$  is the integral of Eq. (12.3) over all outgoing energies, summed over all outgoing particles,

$$W(p_\pi, h_\pi, p_\nu, h_\nu) = \sum_{k=\gamma, n, p, d, t, h, \alpha} \int dE_k W_k(p_\pi, h_\pi, p_\nu, h_\nu, E_k). \quad (12.13)$$

The final ingredient of the exciton model equation, Eq. (12.1), is the part of the pre-equilibrium population  $P$  that has survived emission from all previous steps and has arrived at the exciton state  $(p_\pi, h_\pi, p_\nu, h_\nu)$ . The expression for  $P$  is somewhat more complicated than that of the depletion factor that appears in the one-component exciton model [94]. For two components, contributions

from both particle creation *and* charge exchange reactions need to be taken into account, whereas transitions that do not change the exciton number cancel out in the one-component model.

For the  $(p_\pi, h_\pi, p_\nu, h_\nu)$  state,  $P$  is given by a recursive relation:

$$\begin{aligned}
P(p_\pi, h_\pi, p_\nu, h_\nu) &= P(p_\pi - 1, h_\pi - 1, p_\nu, h_\nu) \Gamma_\pi^+(p_\pi - 1, h_\pi - 1, p_\nu, h_\nu) & (A) \\
&+ P(p_\pi, h_\pi, p_\nu - 1, h_\nu - 1) \Gamma_\nu^+(p_\pi, h_\pi, p_\nu - 1, h_\nu - 1) & (B) \\
&+ [P(p_\pi - 2, h_\pi - 2, p_\nu + 1, h_\nu + 1) \Gamma_\pi^+(p_\pi - 2, h_\pi - 2, p_\nu + 1, h_\nu + 1) \\
&+ P(p_\pi - 1, h_\pi - 1, p_\nu, h_\nu) \Gamma_\nu^+(p_\pi - 1, h_\pi - 1, p_\nu, h_\nu)] \\
&\times \Gamma_{\nu\pi}^0(p_\pi - 1, h_\pi - 1, p_\nu + 1, h_\nu + 1) & (C+D) \\
&+ [P(p_\pi, h_\pi, p_\nu - 1, h_\nu - 1) \Gamma_\pi^+(p_\pi, h_\pi, p_\nu - 1, h_\nu - 1) \\
&+ P(p_\pi + 1, h_\pi + 1, p_\nu - 2, h_\nu - 2) \Gamma_\nu^+(p_\pi + 1, h_\pi + 1, p_\nu - 2, h_\nu - 2)] \\
&\times \Gamma_{\pi\nu}^0(p_\pi + 1, h_\pi + 1, p_\nu - 1, h_\nu - 1) & (E+F).
\end{aligned} \tag{12.14}$$

This relation contains 6 distinct feeding terms: (A) creation of a proton particle-hole pair from the  $(p_\pi - 1, h_\pi - 1, p_\nu, h_\nu)$  state, (C) creation of a proton particle-hole pair from the  $(p_\pi - 2, h_\pi - 2, p_\nu + 1, h_\nu + 1)$  state followed by the conversion of a neutron particle-hole pair into a proton particle-hole pair, and (D) creation of a neutron particle-hole pair from the  $(p_\pi - 1, h_\pi - 1, p_\nu, h_\nu)$  state followed by its conversion into a proton particle-hole pair. The three remaining terms (B), (E), and (F) are obtained by changing protons into neutrons and vice versa. The probabilities of creating new proton or neutron particle-hole pairs and for converting a proton (neutron) pair into a neutron (proton) pair are calculated as follows:

$$\begin{aligned}
\Gamma_\pi^+(p_\pi, h_\pi, p_\nu, h_\nu) &= \lambda_\pi^+(p_\pi, h_\pi, p_\nu, h_\nu) \tau(p_\pi, h_\pi, p_\nu, h_\nu), \\
\Gamma_\nu^+(p_\pi, h_\pi, p_\nu, h_\nu) &= \lambda_\nu^+(p_\pi, h_\pi, p_\nu, h_\nu) \tau(p_\pi, h_\pi, p_\nu, h_\nu), \\
\Gamma_\pi^{\prime+}(p_\pi, h_\pi, p_\nu, h_\nu) &= \lambda_\pi^+(p_\pi, h_\pi, p_\nu, h_\nu) \tau'(p_\pi, h_\pi, p_\nu, h_\nu), \\
\Gamma_\nu^{\prime+}(p_\pi, h_\pi, p_\nu, h_\nu) &= \lambda_\nu^+(p_\pi, h_\pi, p_\nu, h_\nu) \tau'(p_\pi, h_\pi, p_\nu, h_\nu), \\
\Gamma_{\pi\nu}^0(p_\pi, h_\pi, p_\nu, h_\nu) &= \lambda_{\pi\nu}^0(p_\pi, h_\pi, p_\nu, h_\nu) \tau(p_\pi, h_\pi, p_\nu, h_\nu), \\
\Gamma_{\nu\pi}^0(p_\pi, h_\pi, p_\nu, h_\nu) &= \lambda_{\nu\pi}^0(p_\pi, h_\pi, p_\nu, h_\nu) \tau(p_\pi, h_\pi, p_\nu, h_\nu), \\
\tau'(p_\pi, h_\pi, p_\nu, h_\nu) &= [\lambda_\pi^+(p_\pi, h_\pi, p_\nu, h_\nu) + \lambda_\nu^+(p_\pi, h_\pi, p_\nu, h_\nu) \\
&+ W(p_\pi, h_\pi, p_\nu, h_\nu)]^{-1}.
\end{aligned} \tag{12.15}$$

The use of  $\tau'$  in the probability  $\Gamma_\pi^{\prime+}$  ( $\Gamma_\nu^{\prime+}$ ), to create a new proton (neutron) particle-hole pair preceding an exchange interaction, originates from the approximation that only one exchange interaction is allowed in each pair-creation step. The appropriate lifetime in this case consists merely of pair creation and emission rates, see Kalbach [101].

The initial condition for the recursive equations is

$$P(p_\pi^0, h_\pi^0, p_\nu^0, h_\nu^0) = 1, \tag{12.16}$$

after which  $P$  can be solved for any configuration.

To calculate the pre-equilibrium spectrum, the only quantities left to determine are the internal transition rates  $\lambda_\pi^+$ ,  $\lambda_\nu^+$ ,  $\lambda_{\pi\nu}^0$  and  $\lambda_{\nu\pi}^0$ .



#### D. Internal transition rates

The transition rate  $\lambda_{\pi}^{+}$  for the creation of a proton particle-hole pair is given by four terms, accounting for  $p_{\pi}$ ,  $h_{\pi}$ ,  $p_{\nu}$  and  $h_{\nu}$  scattering that leads to a new  $(p_{\pi}, h_{\pi})$  pair,

$$\begin{aligned} \lambda_{\pi}^{+}(p_{\pi}, h_{\pi}, p_{\nu}, h_{\nu}) &= \frac{1}{\omega(p_{\pi}, h_{\pi}, p_{\nu}, h_{\nu}, E^{\text{tot}})} \\ & \left[ \int_{L_1^{p_{\pi}}}^{L_2^{p_{\pi}}} \lambda_{\pi\pi}^{1p}(u) \omega(p_{\pi} - 1, h_{\pi}, p_{\nu}, h_{\nu}, E^{\text{tot}} - u) \omega(1, 0, 0, 0, u) du \right. \\ & + \int_{L_1^{h_{\pi}}}^{L_2^{h_{\pi}}} \lambda_{\pi\pi}^{1h}(u) \omega(p_{\pi}, h_{\pi} - 1, p_{\nu}, h_{\nu}, E^{\text{tot}} - u) \omega(0, 1, 0, 0, u) du \\ & + \int_{L_1^{p_{\nu}}}^{L_2^{p_{\nu}}} \lambda_{\nu\pi}^{1p}(u) \omega(p_{\pi}, h_{\pi}, p_{\nu} - 1, h_{\nu}, E^{\text{tot}} - u) \omega(0, 0, 1, 0, u) du \\ & \left. + \int_{L_1^{h_{\nu}}}^{L_2^{h_{\nu}}} \lambda_{\nu\pi}^{1h}(u) \omega(p_{\pi}, h_{\pi}, p_{\nu}, h_{\nu} - 1, E^{\text{tot}} - u) \omega(0, 0, 0, 1, u) du \right], \end{aligned} \quad (12.17)$$

where the first and third term represent particle scattering and the second and fourth term hole scattering. The integration limits correct for the Pauli exclusion principle,

$$\begin{aligned} L_1^{p_{\pi}} &= A(p_{\pi} + 1, h_{\pi} + 1, p_{\nu}, h_{\nu}) - A(p_{\pi} - 1, h_{\pi}, p_{\nu}, h_{\nu}), \\ L_2^{p_{\pi}} &= E^{\text{tot}} - A(p_{\pi} - 1, h_{\pi}, p_{\nu}, h_{\nu}), \\ L_1^{h_{\pi}} &= A(p_{\pi} + 1, h_{\pi} + 1, p_{\nu}, h_{\nu}) - A(p_{\pi}, h_{\pi} - 1, p_{\nu}, h_{\nu}), \\ L_2^{h_{\pi}} &= E^{\text{tot}} - A(p_{\pi}, h_{\pi} - 1, p_{\nu}, h_{\nu}), \\ L_1^{p_{\nu}} &= A(p_{\pi}, h_{\pi}, p_{\nu} + 1, h_{\nu} + 1) - A(p_{\pi}, h_{\pi}, p_{\nu} - 1, h_{\nu}), \\ L_2^{p_{\nu}} &= E^{\text{tot}} - A(p_{\pi}, h_{\pi}, p_{\nu} - 1, h_{\nu}), \\ L_1^{h_{\nu}} &= A(p_{\pi} + 1, h_{\pi} + 1, p_{\nu}, h_{\nu}) - A(p_{\pi}, h_{\pi}, p_{\nu}, h_{\nu} - 1), \\ L_2^{h_{\nu}} &= E^{\text{tot}} - A(p_{\pi}, h_{\pi}, p_{\nu}, h_{\nu} - 1), \end{aligned} \quad (12.18)$$

which demands that (a) the minimal energy available to the scattering particle or hole creating a new particle-hole pair equals the Pauli energy of the final state minus the Pauli energy of the inactive particles and holes not involved in the scattering process, and (b) the maximal energy available equals the total excitation energy minus the latter Pauli energy.

The term  $\lambda_{\pi\pi}^{1p}(u)$  is the collision probability per unit time for a proton-proton interaction leading to an additional proton particle-hole pair. In general, the corresponding term for a hole is obtained by relating it to the particle collision probability through the accessible state density of the interacting particles and holes,

$$\lambda_{\pi\pi}^{1h}(u) = \lambda_{\pi\pi}^{1p}(u) \frac{\omega(1, 2, 0, 0, u)}{\omega(2, 1, 0, 0, u)}. \quad (12.19)$$

Similarly,  $\lambda_{\nu\pi}^{1p}(u)$  is the collision probability per unit time for a neutron-proton interaction leading to an additional proton particle-hole pair, and for the corresponding hole term we have

$$\lambda_{\nu\pi}^{1h}(u) = \lambda_{\nu\pi}^{1p}(u) \frac{\omega(1, 1, 0, 1, u)}{\omega(1, 1, 1, 0, u)}. \quad (12.20)$$

The transition rate for conversion of a proton particle-hole pair into a neutron pair is

$$\lambda_{\pi\nu}^0(p_{\pi}, h_{\pi}, p_{\nu}, h_{\nu}) = \frac{1}{\omega(p_{\pi}, h_{\pi}, p_{\nu}, h_{\nu}, E^{\text{tot}})} \int_{L_1^{p_{\pi}}}^{L_2^{p_{\pi}}} \lambda_{\pi\nu}^{1p1h}(u) \omega(p_{\pi} - 1, h_{\pi} - 1, p_{\nu}, h_{\nu}, E^{\text{tot}} - u) \omega(1, 1, 0, 0, u) du, \quad (12.21)$$

with integration limits

$$\begin{aligned} L_1^{p\pi} &= A(p_\pi, h_\pi, p_\nu, h_\nu) - A(p_\pi - 1, h_\pi - 1, p_\nu, h_\nu) \\ L_2^{p\pi} &= E^{\text{tot}} - A(p_\pi - 1, h_\pi - 1, p_\nu, h_\nu). \end{aligned} \quad (12.22)$$

The term  $\lambda_{\pi\nu}^{1p1h}$  is the associated collision probability. Interchanging  $\pi$  and  $\nu$  in Eqs. (12.17-12.22) gives the expressions for  $\lambda_\nu^+$ ,  $\lambda_{\nu\nu}^{1h}$ ,  $\lambda_{\pi\nu}^{1h}$  and  $\lambda_{\nu\pi}^0$ .

We distinguish between two options for the collision probabilities:

#### D1. Effective squared matrix element

Expressing the transition rate in terms of an effective squared matrix element has been used in many exciton model analyses. Also in TALYS, it is one of the options for our calculations and comparisons with data. The collision probabilities of Eqs. (12.17) and (12.21) are determined with the aid of Fermi's golden rule of time-dependent perturbation theory, which for a two-component model gives

$$\begin{aligned} \lambda_{\pi\pi}^{1p}(u) &= \frac{2\pi}{\hbar} M_{\pi\pi}^2 \omega(2, 1, 0, 0, u), \\ \lambda_{\pi\pi}^{1h}(u) &= \frac{2\pi}{\hbar} M_{\pi\pi}^2 \omega(1, 2, 0, 0, u), \\ \lambda_{\nu\pi}^{1p}(u) &= \frac{2\pi}{\hbar} M_{\nu\pi}^2 \omega(1, 1, 1, 0, u), \\ \lambda_{\nu\pi}^{1h}(u) &= \frac{2\pi}{\hbar} M_{\nu\pi}^2 \omega(1, 1, 0, 1, u), \\ \lambda_{\pi\nu}^{1p1h}(u) &= \frac{2\pi}{\hbar} M_{\pi\nu}^2 \omega(0, 0, 1, 1, u), \end{aligned} \quad (12.23)$$

where the relations (12.19)-(12.20) have been applied. Interchanging  $\pi$  and  $\nu$  gives the expressions for  $\lambda_{\nu\nu}^{1p}$ ,  $\lambda_{\nu\nu}^{1h}$ ,  $\lambda_{\pi\nu}^{1p}$ ,  $\lambda_{\pi\nu}^{1h}$ , and  $\lambda_{\nu\pi}^{1p1h}$ . Here, the  $M_{\pi\pi}^2$ , etc. are average squared matrix elements of the residual interaction, which are assumed to depend on the total energy  $E^{\text{tot}}$  of the whole composite nucleus only. Such a matrix element thus represents a truly *effective* residual interaction, whereby all individual residual interactions taking place inside the nucleus can be cast into an average form for the squared matrix element to which one assigns a global  $E^{\text{tot}}$ -dependence a posteriori.

The average residual interaction inside the nucleus is not necessarily the same for like and unlike nucleons. The two-component matrix elements are given, in terms of an average  $M^2$ , by

$$\begin{aligned} M_{\pi\pi}^2 &= R_{\pi\pi} M^2, \\ M_{\nu\nu}^2 &= R_{\nu\nu} M^2, \\ M_{\pi\nu}^2 &= R_{\pi\nu} M^2, \\ M_{\nu\pi}^2 &= R_{\nu\pi} M^2. \end{aligned} \quad (12.24)$$

In TALYS, we take

$$R_{\nu\nu} = 1.5, R_{\nu\pi} = R_{\pi\pi} = R_{\pi\nu} = 1., \quad (12.25)$$

which is in line with the, more parameter-free, optical model based exciton model that we describe later. Note that this deviates somewhat from the parameters adopted by Koning and Duijvestijn [92]. The current parameterization gives slightly better performance for cross section excitation functions.

In TALYS, the above parameters are adjustable with Eq. (12.25) as default. The following semi-empirical expression for the squared matrix element has been shown to work for incident energies between 7 and 200 MeV [92]:

$$M^2 = \frac{C_1 A_p}{A^3} \left[ 7.48 C_2 + \frac{4.62 \times 10^5}{\left(\frac{E^{tot}}{n A_p} + 10.7 C_3\right)^3} \right]. \quad (12.26)$$

Eq. (12.26) is a generalization of older parameterisations such as given in Refs. [101, 91], which apply in smaller (lower) energy ranges. Here  $C_1, C_2$  and  $C_3$  are adjustable constants that are all equal to 1 by default, and  $A_p$  is the mass number of the projectile, which allows generalization for complex-particle reactions. Again, Eq. (12.26) is slightly different (10%) from the expression given in Ref. [92] to allow for better fits of excitation functions.

Finally, for matrix element based transition rates and equidistant particle-hole level densities, the integrals in the transition rates can be approximated analytically [101], giving

$$\begin{aligned} \lambda_{\pi+}(p_\pi, h_\pi, p_\nu, h_\nu) &= \frac{2\pi}{\hbar} \frac{g_\pi^2}{2n(n+1)} \frac{[E^{tot} - A(p_\pi + 1, h_\pi + 1, p_\nu, h_\nu)]^{n+1}}{[E^{tot} - A(p_\pi, h_\pi, p_\nu, h_\nu)]^{n-1}} \\ &\times (n_\pi g_\pi M_{\pi\pi}^2 + 2n_\nu g_\nu M_{\pi\nu}^2) f(p+1, h+1, E^{tot}, V) \\ \lambda_{\nu+}(p_\pi, h_\pi, p_\nu, h_\nu) &= \frac{2\pi}{\hbar} \frac{g_\nu^2}{2n(n+1)} \frac{[E^{tot} - A(p_\pi, h_\pi, p_\nu + 1, h_\nu + 1)]^{n+1}}{[E^{tot} - A(p_\pi, h_\pi, p_\nu, h_\nu)]^{n-1}} \\ &\times (n_\nu g_\nu M_{\nu\nu}^2 + 2n_\pi g_\pi M_{\nu\pi}^2) f(p+1, h+1, E^{tot}, V) \\ \lambda_{\pi\nu}(p_\pi, h_\pi, p_\nu, h_\nu) &= \frac{2\pi}{\hbar} M_{\pi\nu}^2 \frac{p_\pi h_\pi}{n} g_\nu^2 f(p, h, E^{tot}, V) \left[ \frac{E^{tot} - B_{\pi\nu}(p_\pi, h_\pi, p_\nu, h_\nu)}{E^{tot} - A(p_\pi, h_\pi, p_\nu, h_\nu)} \right]^{n-1} \\ &\times (2[E^{tot} - B_{\pi\nu}(p_\pi, h_\pi, p_\nu, h_\nu)] \\ &+ n|A(p_\pi, h_\pi, p_\nu, h_\nu) - A(p_\pi - 1, h_\pi - 1, p_\nu + 1, h_\nu + 1)|) \\ \lambda_{\nu\pi}(p_\pi, h_\pi, p_\nu, h_\nu) &= \frac{2\pi}{\hbar} M_{\nu\pi}^2 \frac{p_\nu h_\nu}{n} g_\pi^2 f(p, h, E^{tot}, V) \left[ \frac{E^{tot} - B_{\nu\pi}(p_\pi, h_\pi, p_\nu, h_\nu)}{E^{tot} - A(p_\pi, h_\pi, p_\nu, h_\nu)} \right]^{n-1} \\ &\times (2[E^{tot} - B_{\nu\pi}(p_\pi, h_\pi, p_\nu, h_\nu)] \\ &+ n|A(p_\pi, h_\pi, p_\nu, h_\nu) - A(p_\pi + 1, h_\pi + 1, p_\nu - 1, h_\nu - 1)|), \quad (12.27) \end{aligned}$$

with

$$\begin{aligned} B_{\pi\nu}(p_\pi, h_\pi, p_\nu, h_\nu) &= \max[A(p_\pi, h_\pi, p_\nu, h_\nu), A(p_\pi - 1, h_\pi - 1, p_\nu + 1, h_\nu + 1)] \\ B_{\nu\pi}(p_\pi, h_\pi, p_\nu, h_\nu) &= \max[A(p_\pi, h_\pi, p_\nu, h_\nu), A(p_\pi + 1, h_\pi + 1, p_\nu - 1, h_\nu - 1)]. \quad (12.28) \end{aligned}$$

which is also included as an option. The default is however to use the numerical solutions for the internal transition rates. This analytical solution requires a value for  $M^2$  that is 20% larger than that of Eq. (12.26), which apparently is the energy-averaged effect of introducing such approximations.

## D2. Collision rates based on the optical model

Instead of modeling the intranuclear transition rate by an average squared matrix element, one may also relate the transition rate to the average imaginary optical model potential depth [92]. The collision probabilities, when properly averaged over all particle-hole configurations as in Eq. (12.17), in principle would yield a parameter free expression for the transition rate.

The average well depth  $W_i$  can be obtained by averaging the total imaginary part of the potential  $\mathcal{W}$  over the whole volume of the nucleus

$$W_i(E) = \frac{\int \mathcal{W}_i(r, E) \rho(r) dr}{\int \rho(r) dr}, \quad (12.29)$$

where  $\rho$  represents the density of nuclear matter for which we take the form factor  $f(r, R, a)$  of the volume part of the optical model potential, given by the usual Woods-Saxon shape of Eq. (10.3). The total imaginary potential is given by

$$\mathcal{W}_i(r, E) = \mathcal{W}_{V,i}(E) f(r, R_{V,i}, a_{V,i}) - 4a_{D,i} \mathcal{W}_{D,i}(E) \frac{d}{dr} f(r, R_{D,i}, a_{D,i}). \quad (12.30)$$

Next, we define an *effective* imaginary optical potential [92] related to nucleon-nucleon collisions in nuclear matter:

$$W_i^{\text{eff}}(E) = C^{\text{omp}} W_i(E). \quad (12.31)$$

We use as best overall parameter

$$C^{\text{omp}} = 0.55. \quad (12.32)$$

This parameter can be adjusted with the **M2constant** keyword, which serves as a multiplier for the value given in Eq. (12.32). The collision probabilities are now related as follows to the *effective* imaginary optical potential:

$$\begin{aligned} \lambda_{\pi\pi}^{1p}(u) &= \frac{1}{4} \frac{2W_p^{\text{eff}}(u - S(p))}{\hbar} \\ \lambda_{\pi\pi}^{1h}(u) &= \frac{1}{4} \frac{2W_p^{\text{eff}}(u - S(p))}{\hbar} \frac{\omega(1, 2, 0, 0, u)}{\omega(2, 1, 0, 0, u)} \\ \lambda_{\nu\pi}^{1p}(u) &= \frac{3}{4} \frac{2W_n^{\text{eff}}(u - S(n))}{\hbar} \\ \lambda_{\nu\pi}^{1h}(u) &= \frac{3}{4} \frac{2W_n^{\text{eff}}(u - S(n))}{\hbar} \frac{\omega(1, 1, 0, 1, u)}{\omega(1, 1, 1, 0, u)} \\ \lambda_{\nu\pi}^{1p1h}(u) &= \frac{1}{2} \frac{2W_n^{\text{eff}}(u - S(n))}{\hbar}, \end{aligned} \quad (12.33)$$

and similarly for the components of  $\lambda_{\nu}^+$  and  $\lambda_{\nu\pi}^0$ . The transition rates for the exciton model are then obtained by inserting these terms in Eqs. (12.17) and (12.21). Apart from Eq. (12.32), a parameter-free model is obtained.

### 12.1.2 One-component exciton model

The one-component exciton model has been made redundant by the more flexible and physically more justified two-component model. Nevertheless, it is included as an option since it connects to many older pre-equilibrium studies and thus may be helpful as comparison. In the one-component exciton model, the pre-equilibrium spectrum for the emission of a particle  $k$  at an energy  $E_k$  is given by

$$\frac{d\sigma_k^{\text{PE}}}{dE_k} = \sigma^{\text{CF}} \sum_{p=p_0}^{p_{\text{max}}} W_k(p, h, E_k) \tau(p, h), \quad (12.34)$$

where  $p_{\max} = 6$  and  $\sigma^{\text{CF}}$  are defined as below Eq. (12.1). For the initial particle number  $p_0$  we have  $p_0 = A_p$  with  $A_p$  the mass number of the projectile. In general, the hole number  $h = p - p_0$  in Eq. (12.34), so that the initial hole number is always zero, i.e.  $h_0 = 0$  for primary pre-equilibrium emission.

The emission rate  $W_k$  is

$$W_k(p, h, E_k) = \frac{2s_k + 1}{\pi^2 \hbar^3} \mu_k E_k \sigma_{k, \text{inv}}(E_k) \frac{\omega(p - A_k, h, E_{\text{tot}} - E_k)}{\omega(p, h, E_{\text{tot}})} Q_k(p), \quad (12.35)$$

with all quantities explained below Eq. (12.3), and  $A_k$  is the mass number of the ejectile and  $Q_k(p)$  is a factor accounting for the distinguishability of neutrons and protons [102]

$$Q_k(p) = \frac{(p - A_k)!}{p!} \left[ \sum_{p_\pi = Z_k}^{p - N_k} \left(\frac{Z}{A}\right)^{n_\pi - Z_k} \left(\frac{A}{N}\right)^{n_\nu - N_k} \frac{1}{h_\pi! h_\nu!} \frac{1}{(p_\pi - Z_k)! (p_\nu - N_k)!} \right] / \left[ \sum_{p_\pi = Z_k}^{p - N_k} \left(\frac{Z}{A}\right)^{n_\pi} \left(\frac{N}{A}\right)^{n_\nu} \frac{1}{p_\pi! p_\nu! h_\pi! h_\nu!} \right]. \quad (12.36)$$

For gamma's we set  $Q_\gamma(p) = 1$ .

Finally,  $\omega(p, h, E_x)$  is the particle-hole state density for which we use the one-component expression by Běťák and Dobeš [98], again corrected for the effect of the Pauli exclusion principle and for the finite depth of the potential well. The one-component particle-hole state density has a simpler form than that of Eq. (12.4),

$$\omega(p, h, E_x) = \frac{g^n}{p! h! (n-1)!} [E_x - A(p, h)]^{n-1} f(p, h, E_x, V), \quad (12.37)$$

where  $g = A/15$  is the single-particle state density and

$$A(p, h) = \frac{[\max(p, h)]^2}{g} - \frac{p^2 + h^2 + p + h}{4g}, \quad (12.38)$$

is the Pauli correction factor. The finite well function  $f$  is given by Eq. (12.10).

To obtain the lifetimes  $\tau(p, h)$  that appear in Eq. (12.34), we first define the total emission rate  $W(p, h)$  as the integral of Eq. (12.35) over all outgoing energies, summed over all outgoing particles:

$$W(p, h) = \sum_{k=\gamma, n, p, d, t, h, \alpha} \int dE_k W_k(p, h, E_k). \quad (12.39)$$

As mentioned already, we have implemented the never-come-back solution of the master equation. This is based on the assumption that at the beginning of the cascade one neglects the interactions that decrease the exciton number. Then, for the one-component model the expression for the lifetime is (see e.g. Ref. [94])

$$\tau(p, h) = \frac{1}{\lambda^+(p, h) + W(p, h)} D_{p, h}, \quad (12.40)$$

where  $D_{p, h}$  is a depletion factor that accounts for the removal of reaction flux, through emission, by the previous stages

$$D_{p, h} = \prod_{p'=p_0}^{p-1} \frac{\lambda^+(p', h')}{\lambda^+(p', h') + W(p', h')}, \quad (12.41)$$

with again  $h' = p' - p_0$ . The initial case of Eq. (12.40) is

$$\tau(p_0, h_0) = \frac{1}{\lambda^+(p_0, h_0) + W(p_0, h_0)}. \quad (12.42)$$

To calculate the pre-equilibrium spectrum, the only quantity left to determine is the internal transition rate  $\lambda^+(p, h)$  from state  $(p, h)$  to state  $(p+1, h+1)$ . The general definition of  $\lambda^+(p, h)$  is

$$\begin{aligned} \lambda^+(p, h) = & \frac{1}{\omega(p, h, E^{\text{tot}})} \left[ \int_{L_1^p}^{L_2^p} du \lambda^{1p}(u) \omega(p-1, h, E^{\text{tot}} - u) \omega(1, 0, u) \right. \\ & \left. + \int_{L_1^h}^{L_2^h} du \lambda^{1h}(u) \omega(p, h-1, E^{\text{tot}} - u) \omega(0, 1, u) \right]. \end{aligned} \quad (12.43)$$

where the two terms account for particle and hole scattering, respectively, and the integration limits

$$\begin{aligned} L_1^p &= A(p+1, h+1) - A(p-1, h) \\ L_2^p &= E^{\text{tot}} - A(p-1, h) \\ L_1^h &= A(p+1, h+1) - A(p, h-1) \\ L_2^h &= E^{\text{tot}} - A(p, h-1), \end{aligned} \quad (12.44)$$

correct for the Pauli exclusion principle.

We again distinguish between two options:

### 1. Effective squared matrix element

The collision probabilities are determined with the aid of Fermi's golden rule of time-dependent perturbation theory, which for the one-component model are

$$\begin{aligned} \lambda^{1p}(u) &= \frac{2\pi}{\hbar} M^2 \omega(2, 1, u) \\ \lambda^{1h}(u) &= \frac{2\pi}{\hbar} M^2 \omega(1, 2, u), \end{aligned} \quad (12.45)$$

with  $M^2$  the average squared matrix element of the residual interaction. In the one-component model, "forbidden" transitions are taken into account, so that a squared matrix element smaller than that of Eq. (12.26) of the two-component model is needed to compensate for these transitions. We find that for the one-component model we need to multiply  $M^2$  of Eq. (12.26) by 0.5 to obtain a global comparison with data that is closest to our two-component result, i.e.

$$M^2 = \frac{0.5C_1A_p}{A^3} \left[ 7.48C_2 + \frac{4.62 \times 10^5}{\left(\frac{E^{\text{tot}}}{n.A_p} + 10.7C_3\right)^3} \right]. \quad (12.46)$$

For completeness, we note that the transition rate can be well approximated by an analytical form as discussed in Refs. [103, 98, 100]. The result is

$$\lambda^+(p, h) = \frac{2\pi}{\hbar} M^2 \frac{g^3}{2(n+1)} \frac{[E^{\text{tot}} - A(p+1, h+1)]^{n+1}}{[E^{\text{tot}} - A(p, h)]^{n-1}} f(p+1, h+1, E^{\text{tot}}, V). \quad (12.47)$$

However, the overall description of experimental data obtained with the one-component model is however worse than that of the two-component model, so we rarely use it.

## 2. Collision rates based on the optical model

Also in the one-component model the transition rates can be related to the effective nucleon-nucleon interaction  $\bar{\sigma}$  and thereby to the imaginary optical potential,

$$\begin{aligned}\lambda^{1p}(u) &= \frac{1}{2} \frac{2W^{\text{eff}}(u-S)}{\hbar} \\ \lambda^{1h}(u) &= \frac{1}{2} \frac{2W^{\text{eff}}(u-S)}{\hbar} \frac{\omega(1,2,u)}{\omega(2,1,u)},\end{aligned}\quad (12.48)$$

with  $S$  the separation energy of the particle. Since the one-component model makes no distinction between neutron and proton particle-hole pairs,  $W_V^{\text{eff}}$  is evaluated as follows,

$$W_i^{\text{eff}}(E) = 0.5C^{\text{omp}} W_i(E), \quad (12.49)$$

analogous to the multiplication with a factor 0.5 for  $M^2$ .

### Energy width representation

The formalism given above, i.e. Eqs. (12.34), (12.35) and (12.40), forms a representation in which the time appears, i.e. the dimensions of  $W(p, h)$  and  $\tau(p, h)$  are  $[s]^{-1}$  and  $[s]$  respectively. An alternative expression for the exciton model that is often used is in terms of energy widths. Since this may be more recognisable to some users we also give it here. The partial escape width  $\Gamma_k^\uparrow(p, h, E_k)$  is related to the emission rate by

$$\Gamma_k^\uparrow(p, h, E_k) = \hbar W_k(p, h, E_k). \quad (12.50)$$

Integrated over energy we have

$$\Gamma_k^\uparrow(p, h) = \hbar W_k(p, h), \quad (12.51)$$

and the total escape width is

$$\Gamma^\uparrow(p, h) = \sum_{k=\gamma, n, p, d, t, h, \alpha} \Gamma_k^\uparrow(p, h) = \hbar W(p, h). \quad (12.52)$$

The damping width  $\Gamma^\downarrow$  is related to the internal transition rate by

$$\Gamma^\downarrow(p, h) = \hbar \lambda^+(p, h). \quad (12.53)$$

Defining the total width by

$$\Gamma^{\text{tot}}(p, h) = \Gamma^\downarrow(p, h) + \Gamma^\uparrow(p, h), \quad (12.54)$$

we can rewrite the exciton model cross section (12.34) as

$$\frac{d\sigma_k^{\text{PE}}}{dE_k} = \sigma_{\text{CF}} \sum_{p=p_0}^{p_{\text{max}}} \frac{\Gamma_k^\uparrow(p, h, E_k)}{\Gamma^{\text{tot}}(p, h)} \left( \prod_{p'=p_0}^{p-1} \frac{\Gamma^\downarrow(p', h')}{\Gamma^{\text{tot}}(p', h')} \right). \quad (12.55)$$

In the output file of TALYS, the results for the various quantities in both the time and the energy width representation are given.

In sum, the default model used by TALYS is the two-component exciton model with collision probabilities based on the effective squared matrix element of Eq. (12.26).

**Options 12.1.1** Pre-equilibrium calculations can be enabled or disabled with the **preequilibrium** keyword (p. 566).

## 12.2 Photon exciton model

For pre-equilibrium photon emission, we have implemented the model of Akkermans and Grupelaar [104]. This model gives a simple but powerful simulation of the direct-semidirect capture process within the framework of the exciton model. Analogous to the particle emission rates, the continuum  $\gamma$ -ray emission rates may be derived from the principle of detailed balance or microscopic reversibility, assuming that only  $E1$ -transitions contribute. This yields

$$W_\gamma(p, h, E_\gamma) = \frac{E_\gamma^2}{\pi^2 \hbar^3 c^2} \frac{\sigma_{\gamma,abs}(E_\gamma)}{\omega(p, h, E^{tot})} \left( \frac{g^2 E_\gamma \omega(p-1, h-1, E_x - E_\gamma)}{g(n-2) + g^2 E_\gamma} + \frac{gn \omega(p, h, E_x - E_\gamma)}{gn + g^2 E_\gamma} \right) \quad (12.56)$$

where  $\sigma_{\gamma,abs}(E_\gamma)$  is the photon absorption cross section (Eq. (9.18)). The initial particle-hole configuration in Eq. (12.34) is  $n_0 = 1$  ( $1p0h$ ) for photon emission. For “direct”  $\gamma$ -ray emission in nucleon-induced reactions only the second term between brackets ( $n = 1$ ) contributes. The “semi-direct”  $\gamma$ -ray emission ( $n = 3$ ) consists of both terms.

The emission rate (12.56) is included in Eqs. (12.34) and (12.39) so that the pre-equilibrium photon cross section automatically emerges.

For the two-component model, we use

$$\begin{aligned} W_\gamma(p_\pi, h_\pi, p_\nu, h_\nu, E_\gamma) &= \frac{E_\gamma^2}{\pi^2 \hbar^3 c^2} \frac{\sigma_{\gamma,abs}(E_\gamma)}{\omega(p_\pi, h_\pi, p_\nu, h_\nu, E^{tot})} \\ &\times \left( \frac{g^2 E_\gamma^{\frac{1}{2}} [\omega(p_\pi - 1, h_\pi - 1, p_\nu, h_\nu, E_x - E_\gamma) + \omega(p_\pi, h_\pi, p_\nu - 1, h_\nu - 1, E_x - E_\gamma)]}{g(n-2) + g^2 E_\gamma} \right. \\ &\left. + \frac{gn \omega(p_\pi, h_\pi, p_\nu, h_\nu, E_x - E_\gamma)}{gn + g^2 E_\gamma} \right). \end{aligned} \quad (12.57)$$

**Options 12.2.1** The strength of the photonuclear contribution can be steered with **Rgamma** (p. 593).

## 12.3 Pre-equilibrium spin distribution

Since the exciton model described above does not provide a spin distribution for the residual states after pre-equilibrium emission, a model needs to be adopted that provides the spin population in the continuum in binary reactions. TALYS provides two options for this. The default is to adopt the compound nucleus spin distribution (described in Chapter 13) also for the excited states resulting from pre-equilibrium emission. Another option that has been quite often used in the past is to assign a spin distribution to the particle-hole state density. For that, we adopt the usual decomposition of the state density into a  $J$ -dependent part and an energy-dependent part,

$$\rho(p, h, J, E_x) = (2J + 1) R_n(J) \omega(p, h, E_x). \quad (12.58)$$

The function  $R_n(J)$  represents the spin distribution of the states in the continuum. It is given by

$$R_n(J) = \frac{2J + 1}{\pi^{1/2} n^{3/2} \sigma^3} \exp \left[ -\frac{(J + \frac{1}{2})^2}{n \sigma^2} \right], \quad (12.59)$$



and satisfies, for any exciton number  $n$ ,

$$\sum_J (2J+1)R_n(J) = 1. \quad (12.60)$$

The used expression for the spin cut-off parameter  $\sigma$  is that by Gruppelaar et al. [105],

$$\sigma^2 = 0.24nA^{\frac{2}{3}}, \quad (12.61)$$

where  $A$  is the mass number of the nucleus. Similarly, for the two-component particle-hole level density we have

$$\rho(p_\pi, h_\pi, p_\nu, h_\nu, J, E_x) = (2J+1)R_n(J)\omega(p_\pi, h_\pi, p_\nu, h_\nu, E_x). \quad (12.62)$$

In practice, with this option the residual states formed by pre-equilibrium reactions would be multiplied by  $R_n$  a posteriori. There are various arguments to prefer the compound nucleus spin distribution, so we use that default.

■ **Options 12.3.1** With `preeqspin y` (p. 570) Eq. (12.61) is invoked.

## 12.4 Continuum stripping, pick-up, break-up and knock-out reactions

For pre-equilibrium reactions involving deuterons, tritons, Helium-3 and alpha particles, a contribution from the exciton model is automatically calculated with the formalism of the previous subsections. It is however well-known that for nuclear reactions involving projectiles and ejectiles with different particle numbers, mechanisms like stripping, pick-up, break-up and knock-out play an important role and these direct-like reactions are not covered by the exciton model. Therefore, Kalbach [106] developed a phenomenological contribution for these mechanisms, which we have included in TALYS. In total, the pre-equilibrium cross section for these reactions is given by the sum of an exciton model (EM), nucleon transfer (NT), and knock-out (KO) contribution:

$$\frac{d\sigma_k^{\text{PE}}}{dE_k} = \frac{d\sigma_k^{\text{EM}}}{dE_k} + \frac{d\sigma_k^{\text{NT}}}{dE_k} + \frac{d\sigma_k^{\text{KO}}}{dE_k} \quad (12.63)$$

where the contribution from the exciton model was outlined in the previous subsection.

### 12.4.1 Transfer reactions

The general differential cross section formula for a nucleon transfer reaction of the type  $A(a,b)B$  is

$$\begin{aligned} \frac{d\sigma_{a,b}^{\text{NT}}}{dE_b} &= \frac{2s_b + 1}{2s_a + 1} \frac{A_b E_b \sigma_{b,\text{inv}}(E_b)}{A_a} K \left( \frac{A_a}{E_a + V_a} \right)^{2n} \left( \frac{C}{A_B} \right)^n \\ &\times N_a \left( \frac{2Z_A}{A_A} \right)^{2(Z_a+2)h_\pi+2p_\nu} \omega_{\text{NT}}(p_\pi, h_\pi, p_\nu, h_\nu, U) \end{aligned} \quad (12.64)$$

where

$$\begin{aligned} C_a &= 5500 \text{ for incident neutrons,} \\ &= 3800 \text{ for incident charged particles,} \end{aligned} \quad (12.65)$$

$$\begin{aligned} N_a &= \frac{1}{80E_a} \text{ for pickup,} \\ &= \frac{1}{580\sqrt{E_a}} \text{ for stripping,} \\ &= \frac{1}{1160\sqrt{E_a}} \text{ for exchange.} \end{aligned} \quad (12.66)$$

$K$  is an enhancement factor taking into account the fact that  $d$ ,  $t$  and  $3\text{-He}$  are loosely bound:

$$\begin{aligned} K &= 12 \text{ for } (N, \alpha), \\ &= 12 - 11 \frac{E_a - 20}{E_a} \text{ for } (\alpha, N) \text{ and } E_a > 20, \\ &= 1 \text{ otherwise,} \end{aligned} \quad (12.67)$$

where  $N$  stands for either neutron or proton. The well depth  $V_a$  is set at

$$V_a = 12.5A_a \text{ MeV}, \quad (12.68)$$

and represents the average potential drop seen by the projectile between infinity and the Fermi level. The possible degrees of freedom for the reaction are all included in the residual state density  $\omega_{NT}(p_\pi, h_\pi, p_\nu, h_\nu, U)$ . Since we do not use this model to describe exchange reactions in inelastic scattering, there is no need to sum the various terms of Eq. (12.64) over  $p_\pi$ , as in Ref. [106]. The exciton numbers are automatically determined by the transfer reaction, i.e.  $n = |A_a - A_b|$ ,  $n_\pi = h_\pi = |Z_a - Z_b|$ ,  $n_\nu = h_\nu = |N_a - N_b|$ ,  $p_\pi = p_\nu = 0$ . The accessible state density that is directly determined by the reaction is  $\omega(p_\pi, h_\pi, p_\nu, h_\nu, U)$ , given by Eq. (12.4). The total residual state density however also takes into account more complex configurations that can be excited by the transfer reaction. It is given by

$$\begin{aligned} \omega_{NT}(p_\pi, h_\pi, p_\nu, h_\nu, U) &= \sum_{i=0}^3 \sum_{j=0}^{3-i} (X_{NT})^{i+j} \omega(p_\pi + i, h_\pi + i, p_\nu + j, h_\nu + j, U) \\ &+ \sum_{i=0}^{p_\pi} \sum_{j=0}^{h_\pi} \sum_{k=0}^{p_\nu} \sum_{l=0}^{h_\nu} \omega(p_\pi - i, h_\pi - j, p_\nu - k, h_\nu - l, U) \Theta(i + j + k + l - \frac{1}{2}) \end{aligned} \quad (12.69)$$

The first term allows that up to three particle-hole pairs can be excited in a transfer reaction. The factor  $X_{NT}$  represents the probability for exciting such a pair and is given by

$$X_{NT} = \frac{7\sqrt{E_a/A_a}}{V_1 A_A^2} (p_\nu^2 + p_\pi^2 + h_\nu^2 + 1.5h_\pi^2) \quad (12.70)$$

For neutrons and protons we adopt for  $V_1$  the value given by Eq.(12.11), for deuterons and tritons we take  $V_1=17$  MeV, and for Helium-3 and alpha particles we take  $V_1=25$  MeV. The finite well depth correction for Eq. (12.69) are made using a well depth of

$$\begin{aligned} V &= V_1 \left( \frac{2Z}{A} \right) \text{ if } n_\pi = 0 \\ &= V_1 \text{ otherwise.} \end{aligned} \quad (12.71)$$

The second term of Eq. (12.69) allows for transfer of nucleons at the Fermi level. Here, the Heaviside function is merely used to avoid double counting of  $\omega(p_\pi, h_\pi, p_\nu, h_\nu, U)$ .

### 12.4.2 Knockout reactions

For (*nucleon*,  $\alpha$ ) reactions a knockout contribution is added. The general differential cross section formula for a knockout reaction of the type  $A(a, b)B$  is

$$\begin{aligned} \frac{d\sigma_{a,b}^{KO}}{dE_b} &= \frac{\sigma_{a,inv}(E_a)}{14} (2s_b + 1) A_b E_b \sigma_{b,inv}(E_b) \\ &\times \frac{P_b g_a g_b [U - A_{KO}(p_a, h_b)]}{\sum_{c=a,b} (2s_c + 1) A_c \langle \sigma_c \rangle (E_{max} + 2B_{coul,c}) (E_{max} - B_{coul,c})^2 g_a g_b^2 / 6g_c} \end{aligned} \quad (12.72)$$

where  $P_b$  is the probability of exciting a  $b$ -type particle-hole pair,  $E_{max}$  is the maximum emission energy, and  $B_{coul,c}$  is the Coulomb barrier for a particle  $c$ . The average inverse cross section  $\langle\sigma_c\rangle$  is given by

$$\langle\sigma_c\rangle = \int_{B_{coul,c}}^{E_{max}} dE \sigma_c(E) \quad (12.73)$$

For the knockout model, the single-particle state density parameters for the cluster degrees of freedom  $g$  represent the number of cluster states per unit energy. The relevant values are given by

$$g_n = N/13, \quad g_p = Z/13, \quad g_\alpha = A/208 \text{ MeV}. \quad (12.74)$$

The Pauli correction factor  $A_{KO}$  is given by

$$A_{KO}(p_a, h_b) = \frac{1}{2g_a^2} - \frac{1}{2g_b^2} \quad (12.75)$$

The probabilities for exciting the various particle-hole pairs are

$$\begin{aligned} P_n &= \frac{N_A - \phi Z_A}{A_A - 2\phi Z_A + \phi Z_A/2} \\ P_p &= \frac{Z_A - \phi Z_A}{A_A - 2\phi Z_A + \phi Z_A/2} \\ P_\alpha &= \frac{\phi Z_A/2}{A_A - 2\phi Z_A + \phi Z_A/2} \end{aligned} \quad (12.76)$$

The factors  $\phi$  are a kind of pre-formation parameters [106]. The following values are adopted

$$\begin{aligned} N_A \leq 116 &: \phi = 0.08 \\ 116 \leq N_A < 126 &: \phi = 0.02 + 0.06(126 - N_A)/10 \\ 126 \leq N_A < 129 &: \phi = 0.02 + 0.06(N_A - 126)/3 \\ 129 \leq N_A &: \phi = 0.08 \end{aligned} \quad (12.77)$$

### 12.4.3 Break-up reactions

For reactions induced by complex particles, break-up may play an important role. This holds especially for weakly bound projectiles like deuterons. Break-up is here defined as having a projectile fragment emerge from the reaction in a relatively narrow peak centered close to the beam velocity and strongly directed toward forward angles. For deuterons only, a simple model by Kalbach has been included [107]. This leads to an extra contribution in the (d,n) and (d,p) channels.

The centroid energy of the breakup peak, in MeV, is given by

$$\varepsilon_0 = \frac{A_b}{A_a} \left( \varepsilon_a - B_{a,b} - \frac{Z_a Z_A}{9.5} \right) + \frac{Z_b Z_B}{9.5}, \quad (12.78)$$

where  $\varepsilon_a$  represents the channel energy (the energy of both the emitted particle and the recoiling nucleus in the center of mass), and  $B_{a,b}$  is the binding energy in the projectile for the indicated breakup channel (2.224 MeV for deuterons). The peak is assumed to be described by a Gaussian line shape with a width parameter of

$$\Gamma = 1.15 + 0.12E_a - \frac{A_A}{140}, \quad (12.79)$$

where  $E_a$  is the laboratory energy of the incident deuteron, and the width parameter is given in MeV. The break-up cross section is assumed to be

$$\sigma_{BU} = K_{d,b} \frac{(A_A^{1/3} + 0.8)^2}{1 + \exp(\frac{13-E_a}{6})}, \quad (12.80)$$

where the normalization factors are

$$\begin{aligned} K_{d,n} &= 18, \\ K_{d,p} &= 21. \end{aligned} \quad (12.81)$$

Finally, the differential break-up cross section is given by

$$\frac{d\sigma_{a,b}^{BU}}{dE_b} = \sigma_{BU} \frac{1}{\Gamma\sqrt{2\pi}} \exp\left(-\frac{(\epsilon_0 - E_b)^2}{\Gamma^2}\right). \quad (12.82)$$

In the output, we have stored the break-up contribution in the column “knockout” (which is normally only used for nucleon-induced reactions with alpha particles as ejectiles).

**Options 12.4.1** With **preeqcomplex** (p. 574) the various complex particle pre-equilibrium contributions can be enabled. The stripping, pick-up, break-up and knock-out contributions can be adjusted with the **Cstrip** (p. 594), **Cbreak** (p. 596) and **Cknock** (p. 595) keywords.

## 12.5 Angular distribution systematics

A sound pre-equilibrium theory should, besides the angle-integrated spectra, also describe the smooth forward peaked angular distributions in the continuum. A physics method to do so has been included in TALYS (multi-step direct reactions, but only as a global collective-like method). Semi-classical models, such as the exciton model, have always had some problems to describe angular distributions (essentially because it is based on a compound-like concept instead of a direct one [108]). A powerful phenomenological method is given by Kalbach [109]. It is based on experimental information only and the insight that in general, a pre-equilibrium process consists of a forward peaked part (multi-step direct) and an isotropic part (multi-step compound), and that the angular distributions are fairly structureless and all look alike. The Kalbach formula for the double-differential cross section for a projectile  $a$  and an ejectile  $b$  is

$$\frac{d^2\sigma_{a,b}}{dE_b d\Omega} = \frac{1}{4\pi} \left[ \frac{d\sigma^{PE}}{dE_b} + \frac{d\sigma^{comp}}{dE_b} \right] \frac{a}{\sinh(a)} [\cosh(a \cos \Theta) + f_{MSD}(E_b) \sinh(a \cos \Theta)] \quad (12.83)$$

where  $\frac{d\sigma^{PE}}{dE_b}$  and  $\frac{d\sigma^{comp}}{dE_b}$  are the angle-integrated pre-equilibrium and compound spectra, respectively, and  $f_{MSD}$  is the so-called multi-step direct or pre-equilibrium ratio:

$$f_{MSD}(E_b) = \frac{d\sigma^{PE}}{dE_b} / \left[ \frac{d\sigma^{PE}}{dE_b} + \frac{d\sigma^{comp}}{dE_b} \right] \quad (12.84)$$

which thus increases from practically 0 at very low emission energy to 1 at the highest emission energies. Hence, once the angle-integrated spectra are known, the parameter  $a$  determines the

angular distribution. Kalbach parameterized it as

$$\begin{aligned}
a(e'_a, e'_b) &= 0.04 \frac{E_1 e'_b}{e'_a} + 1.8 \times 10^{-6} \left( \frac{E_1 e'_b}{e'_a} \right)^3 + 6.7 \times 10^{-7} M_a m_b \left( \frac{E_3 e'_b}{e'_a} \right)^4, \\
E_1 &= \min(e'_a, 130 \text{ MeV}) \\
E_3 &= \min(e'_a, 41 \text{ MeV}) \\
e'_b &= E_b + S_b \\
e'_a &= E_a + S_a. \\
M_a &= 1 \text{ for neutrons, protons, deuterons, tritons and Helium} - 3 \\
&= 0 \text{ for alpha's} \\
m_b &= 1 \text{ for protons, deuterons, tritons and Helium} - 3 \\
&= \frac{1}{2} \text{ for neutrons} \\
&= 2 \text{ for alpha's} \\
S_b &= 15.68(A_C - A_B) - 28.07 \left[ \frac{(N_C - Z_C)^2}{A_C} - \frac{(N_B - Z_B)^2}{A_B} \right] \\
&\quad - 18.56(A_C^{2/3} - A_B^{2/3}) + 33.22 \left[ \frac{(N_C - Z_C)^2}{A_C^{4/3}} - \frac{(N_B - Z_B)^2}{A_B^{4/3}} \right] \\
&\quad - 0.717 \left[ \frac{Z_C^2}{A_C^{1/3}} - \frac{Z_B^2}{A_B^{1/3}} \right] + 1.211 \left[ \frac{Z_C^2}{A_C} - \frac{Z_B^2}{A_B} \right] - I_b \\
I_d &= 2.225 \\
I_t &= 8.482 \\
I_h &= 7.718 \\
I_\alpha &= 28.296,
\end{aligned} \tag{12.85}$$

Here,  $E_a$  and  $E_b$  are the incident and the outgoing energy, respectively. The number  $M_a$  represents the incident particle, while  $m_b$  represents the outgoing particle, C is a label for the compound nucleus, B for the final nucleus and the Myers and Swiatecki mass formula [27] for spherical nuclides should be used here to determine the separation energy  $S$ . Finally  $I_b$  is the energy required to break the emitted particle up into its constituents.

Since we calculate the pre-equilibrium and compound cross sections explicitly (and actually only use  $f_{MSD}$  for ENDF-6 data libraries), Eq. (12.83) can be reduced to a formula for the double-differential pre-equilibrium cross section

$$\frac{d^2 \sigma_{a,xb}^{\text{PE}}}{dE_b d\Omega} = \frac{1}{4\pi} \frac{d\sigma^{\text{PE}}}{dE_b} \frac{a}{\sinh(a)} \exp(a \cos \Theta), \tag{12.86}$$

to which the isotropic compound angular distribution can be added. In sum, given the angle-integrated spectrum  $\frac{d\sigma^{\text{PE}}}{dE_b}$  by some physics model, the double-differential cross section is returned quite simply and reasonably accurate by Eq. (12.86).

### Sample case 12.1 Continuum spectra at 63 MeV for Bi(n,xp)...Bi(n,x $\alpha$ )

In this sample case, we calculate angle-integrated and double-differential particle spectra for 63 MeV neutrons on  $^{209}\text{Bi}$ , see Ref. [110].

**Case a: Default calculation**

The following input file is used

```
#
# n-Bi209-preeq
#
# General
#
projectile n
element bi
mass 209
energy 63.
#
# Output
#
ddxmode 2
filespectrum n p d t h a
fileddxa p 20.
fileddxa p 70.
fileddxa p 110.
fileddxa d 20.
fileddxa d 70.
fileddxa d 110.
fileddxa t 20.
fileddxa t 70.
fileddxa t 110.
fileddxa h 20.
fileddxa h 70.
fileddxa h 110.
fileddxa a 20.
fileddxa a 70.
fileddxa a 110.
```

Note that we request that angle-integrated spectra for all particles are written on separate files through *filespectrum n p d t h a*. At 20, 70 and 110 degrees, we also ask for the double-differential spectrum for protons up to alpha-particles. The resulting files *pspec063.000.tot*, *pddx020.0.deg*, etc. are presented, together with experimental data, in Figs. 12.2 and 12.3.

The default results of case (a) for the proton spectra are a bit high. Therefore, as a second version of this sample case, we adjust a pre-equilibrium parameter and add the following to the input above:

```
#
# Parameters
#
M2constant 0.80
```

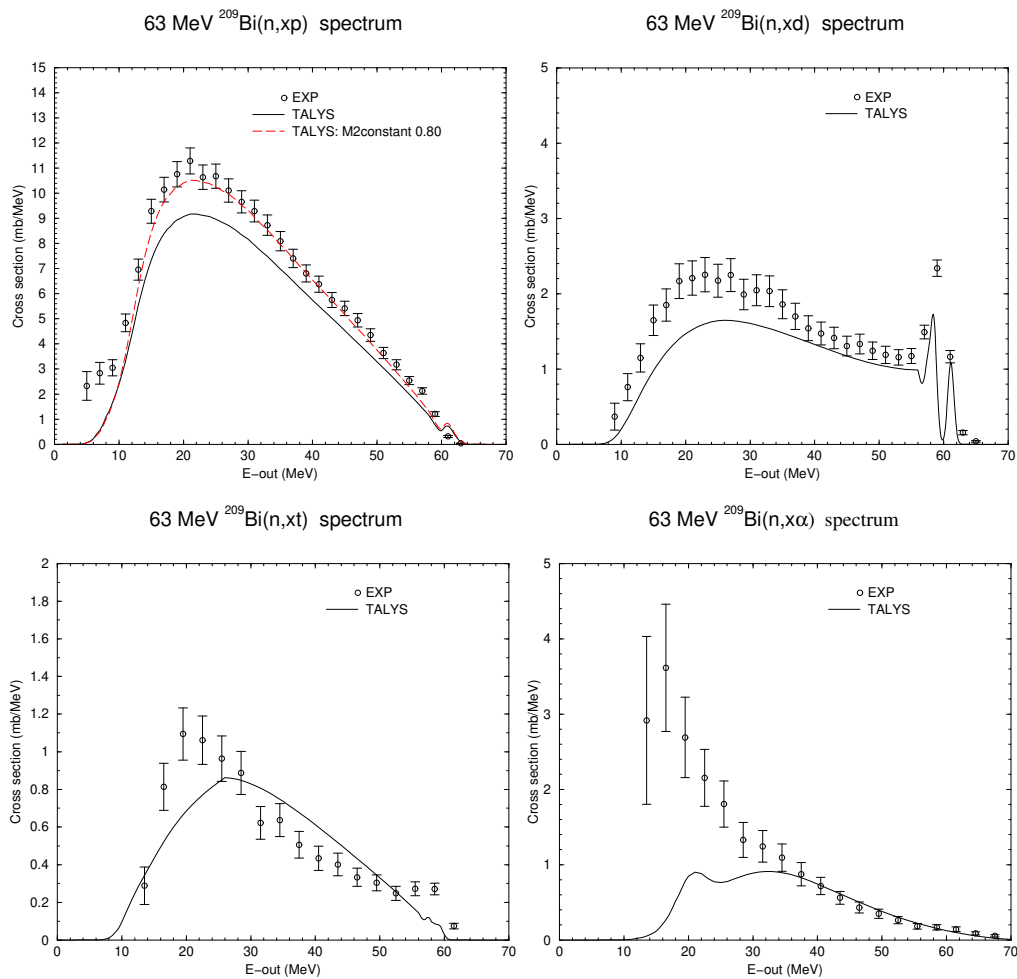


Figure 12.2: Angle-integrated proton, deuteron, triton and alpha emission spectra for 63 MeV neutrons on  $^{209}\text{Bi}$ . The experimental data are from [110].

By decreasing **M2constant** by 20% (which is a common and acceptable deviation from the average), we favor the pre-equilibrium emission rate over the rate rate, leading to a harder spectrum. The result is shown in Fig. 12.2 for the proton spectrum. ■

### Sample case 12.2 Pre-equilibrium angular dist. and multiple pre-equilibrium emission

At high incident energies, multiple pre-equilibrium reactions play a significant role. In this sample case, we show the results for this mechanism in the output file. Also, as an alternative to the previous sample case, we present another way of producing double-differential cross sections, namely as a function of angle at a fixed outgoing energy. With the following input file, the reaction of 120 MeV protons incident on  $^{90}\text{Zr}$  is simulated:

```
#
# p-Zr090-preeq
#
```

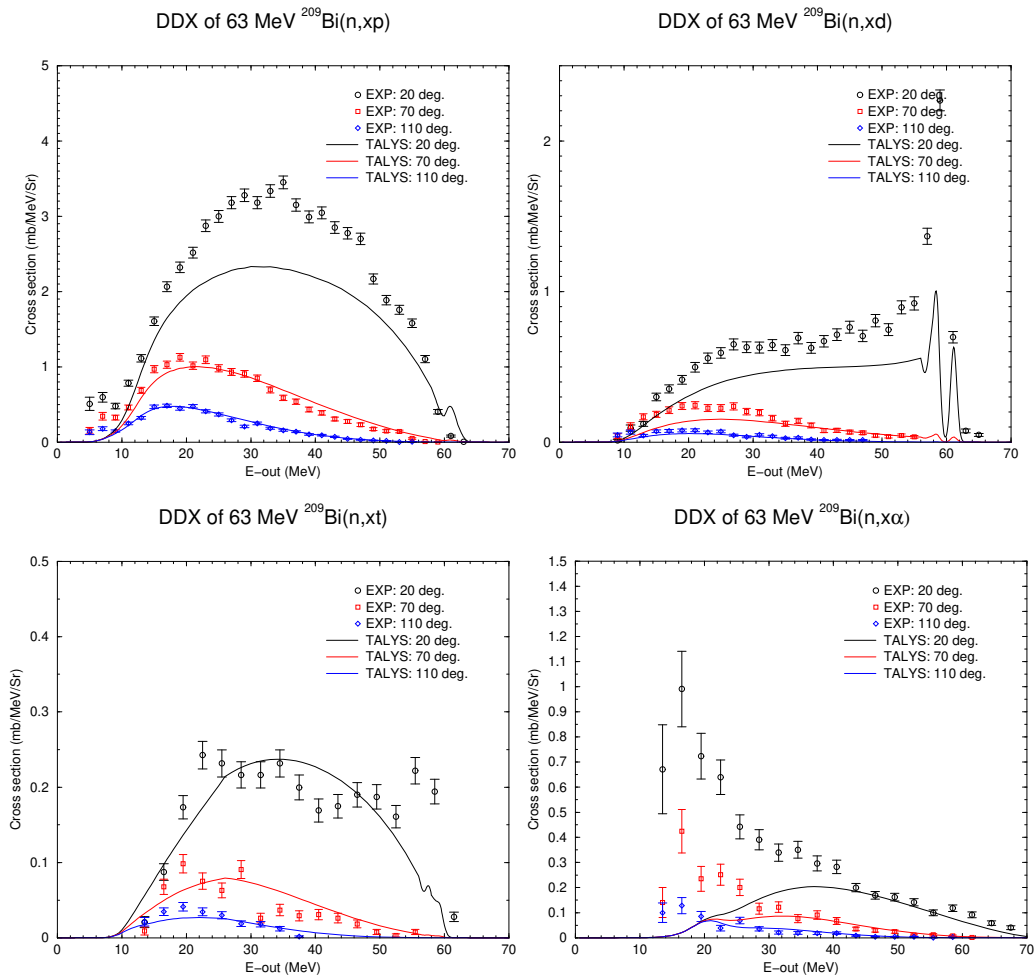


Figure 12.3: Double-differential proton, deuteron, triton and alpha emission spectra for 63 MeV neutrons on  $^{209}\text{Bi}$ . The experimental data are from [110].



```
# General
#
projectile p
element zr
mass 90
energy 120.
#
# Output
#
outpopulation y
ddxmode 1
filespectrum n p
fileddxe p 20.
fileddxe p 40.
fileddxe p 60.
fileddxe p 80.
fileddxe p 100.
```

The results are presented in Figs. 12.4 and 12.5. For this sample case, since **outpopulation y**, after each print of the population for each residual nucleus (as already described in the first sample case), a block with multiple pre-equilibrium decay information is printed. This output block begins with

Multiple preequilibrium emission from Z= 41 N= 49 ( 90Nb):

bin	Ex	Mpe	ratio	neutron		proton		Feeding terms from					
				emission	emission	emission	emission	1 1 0 0	0 0 1 1	1 0 0 1	0 1 1 0		
11	2.208	0.00000	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	5.287E-01	0.000E+00				
12	4.980	0.00000	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	1.249E+00	0.000E+00				
13	7.752	0.00000	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	1.961E+00	0.000E+00				
14	10.524	0.00001	0.000E+00	4.343E-05	1.999E-02	0.000E+00	0.000E+00	2.645E+00	0.000E+00				
15	13.296	0.00555	3.214E-04	1.999E-02	0.000E+00	0.000E+00	0.000E+00	3.300E+00	0.000E+00				
16	16.067	0.04999	5.901E-03	2.049E-01	0.000E+00	0.000E+00	0.000E+00	3.928E+00	0.000E+00				
17	18.839	0.10870	2.305E-02	5.024E-01	0.000E+00	0.000E+00	0.000E+00	4.527E+00	0.000E+00				
18	21.611	0.17335	5.014E-02	8.952E-01	0.000E+00	0.000E+00	0.000E+00	5.098E+00	0.000E+00				

.....

For each continuum bin, with excitation energy  $E_x$ , we print the fraction of the population that is emitted as multiple pre-equilibrium. Also the total neutron and proton emission per residual nucleus is printed, as well as the feeding terms from previous particle-hole configurations. With this input file, files *nspec120.000.tot* and *pspec120.000.tot* are created through the *filespectrum n p* keyword. The results are displayed in Fig. 12.4. Also, the combination of **ddxmode 1** and the various **fileddxe** keywords generate the *pddx100.0.mev*, etc. files that are compared with experimental data in Fig. 12.5.

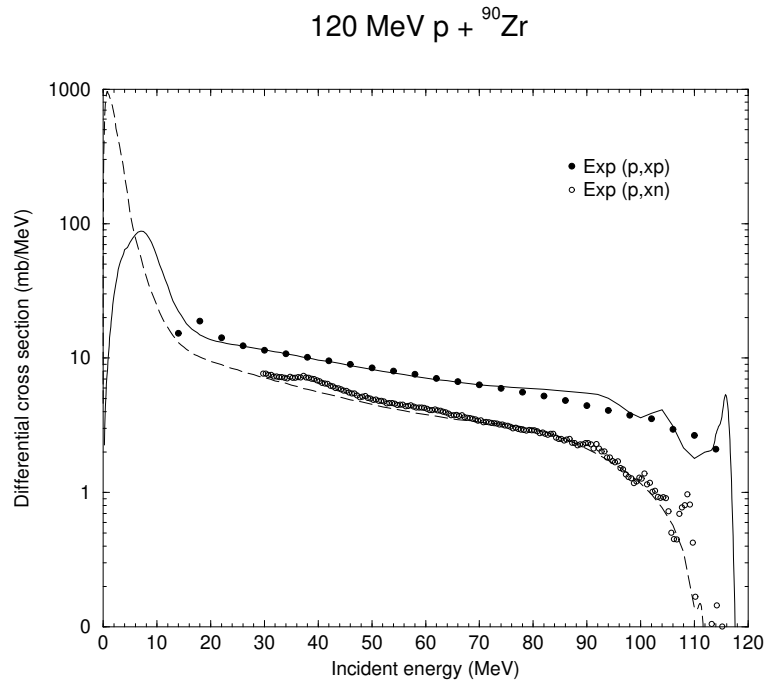


Figure 12.4: Angle-integrated (p,xn) and (p,xp) spectra for 120 MeV protons on  $^{90}\text{Zr}$ . Experimental data are taken from [111, 112]

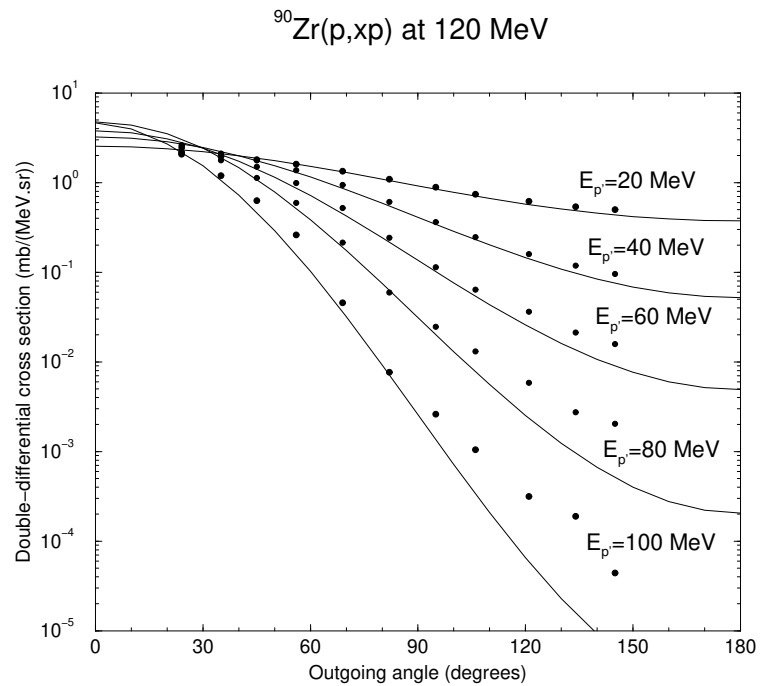


Figure 12.5: Double-differential (p,xp) spectra for 120 MeV protons on  $^{90}\text{Zr}$ . Experimental data are taken from [111]

**Sample case 12.3 Calculations up to 500 MeV for p + <sup>209</sup>Bi**

In TALYS, the maximum allowed incident energy has formally been extended to 1 GeV. The term 'formally' means that the code seems to produce reasonable results, without crashing, while we are well aware that the physics beyond about 200 MeV is different from that at lower energies. The question is at which energy the "low-energy" models will start to fail. To investigate this, the extension to 1 GeV has been made. In this sample case we perform calculations for p + <sup>209</sup>Bi at 5 different energies,

```
#
# p-Bi209-500MeV
#
# General
#
projectile p
element bi
mass 209
energy 100. 500. 100.
bins 60
```

Here we use the option to give equidistant incident energies from 100 to 500 MeV with steps of 100 MeV. All results can be found in the usual files, e.g. the residual production cross sections of <sup>204</sup>Pb are in *rp082204.tot*, which look as follows

```
# p + 209Bi: Production of 204Pb - Total
# Q-value    =-4.42737E+00 mass= 203.973043
# E-threshold= 4.44871E+00
# # energies = 5
#   E          xs
1.000E+02 2.81964E+01
2.000E+02 3.70364E+01
3.000E+02 4.09552E+01
4.000E+02 3.84491E+01
5.000E+02 3.62129E+01
```

Obviously, the user may extend this case up to 1 GeV and with a finer incident energy grid. ■



## 13. Compound reactions

The term compound nucleus reaction is commonly used for two different mechanisms: (i) the process of the capture of the projectile in the target nucleus to form a compound nucleus, which subsequently emits a particle or gamma, (ii) the multiple emission process of highly excited residual nuclei formed after the binary reaction. The latter, which is known as multiple compound emission, will be explained in Section 14. We first treat the binary compound nucleus reaction that plays a role at low incident energy. It differs from the multiple compound emission at two important points: (a) the presence of width fluctuation corrections and (b) non-isotropic, though still symmetric, angular distributions.

### 13.1 Binary compound cross section and angular distribution

In the compound nucleus picture, the projectile and the target nucleus form a compound nucleus with a total energy  $E^{tot}$  and a range of values for the total spin  $J$  and parity  $\Pi$ . The following energy, angular momentum and parity conservation laws need to be obeyed,

$$\begin{aligned}
 E_a + S_a &= E_{a'} + E_x + S_{a'} = E^{tot} \\
 s + I + l &= s' + I' + l' = J \\
 \pi_0 \Pi_0 (-1)^l &= \pi_f \Pi_f (-1)^{l'} = \Pi.
 \end{aligned} \tag{13.1}$$

The compound nucleus formula for the binary cross section is given by

$$\begin{aligned}
 \sigma_{\alpha\alpha'}^{comp} &= D^{comp} \frac{\pi}{k^2} \sum_{J=\text{mod}(I+s,1)}^{l_{max}+I+s} \sum_{\Pi=-1}^1 \frac{2J+1}{(2I+1)(2s+1)} \sum_{j=|J-I|}^{J+I} \sum_{l=|j-s|}^{j+s} \sum_{j'=|J-I|}^{J+I'} \sum_{l'=|j'-s'|}^{j'+s'} \\
 &\times \delta_{\pi}(\alpha) \delta_{\pi}(\alpha') \frac{T_{\alpha l j}^J(E_a) \langle T_{\alpha' l' j'}^J(E_{a'}) \rangle}{\sum_{\alpha'', l'', j''} \delta_{\pi}(\alpha'') \langle T_{\alpha'' l'' j''}^J(E_{a''}) \rangle} W_{\alpha l j \alpha' l' j'}^J,
 \end{aligned} \tag{13.2}$$

In the above equations, the symbols have the following meaning:

$E_a$  = projectile energy

$s$  = spin of the projectile

$\pi_0$  = parity of the projectile

$l$  = orbital angular momentum of the projectile

$j$  = total angular momentum of the projectile

$\delta_\pi(\alpha) = 1$ , if  $(-1)^l \pi_0 \Pi_0 = \Pi$  and 0 otherwise

$\alpha$  = channel designation of the initial system of projectile and target nucleus:

$\alpha = \{a, s, E_a, E_x^0, I, \Pi_0\}$ , where  $a$  is the projectile type and  $E_x^0$  the excitation energy of the target nucleus (usually zero)

$l_{max}$  = maximum  $l$ -value for projectile

$S_a$  = separation energy

$E_{a'}$  = ejectile energy

$s'$  = spin of the ejectile

$\pi_f$  = parity of the ejectile

$l'$  = orbital angular momentum of the ejectile

$j'$  = total angular momentum of the ejectile

$\delta_\pi(\alpha') = 1$ , if  $(-1)^{l'} \pi_f \Pi_f = \Pi$  and 0 otherwise

$\alpha'$  = channel designation of the final system of ejectile and residual nucleus:

$\alpha' = \{a', s', E_{a'}, E_x, I', \Pi_f\}$ , where  $a'$  is the ejectile type,  $E_x$  the excitation energy of the residual nucleus

$I$  = spin of the target nucleus

$\Pi_0$  = parity of the target

$I'$  = spin of the residual nucleus

$\Pi_f$  = parity of the residual nucleus

$\Pi$  = parity of the compound system

$J$  = total angular momentum of the compound system

$D^{comp}$  = depletion factor to account for direct and pre-equilibrium effects

$k$  = wave number of relative motion

$T$  = transmission coefficient

$W$  = width fluctuation correction (WFC) factor, see the next Section.

In order to let Eq. (13.2) represent the general case, we have denoted the outgoing transmission coefficient by  $\langle T_{\alpha'l'j'}^J \rangle$ . For this, two cases can be distinguished. If the excitation energy  $E_x$ , that is implicit in the definition of channel  $\alpha'$ , corresponds to a discrete state of the final nucleus, then we simply have

$$\langle T_{\alpha'l'j'}^J(E_{a'}) \rangle = T_{\alpha'l'j'}^J(E_{a'}) \quad (13.3)$$

and  $E_{a'}$  is exactly determined by Eq. (13.1). For  $\alpha'$  channels in which  $E_x$  is in the continuum, we have an effective transmission coefficient for an excitation energy bin with width  $\Delta E_x$ ,

$$\langle T_{\alpha'l'j'}^J(E_{a'}) \rangle = \int_{E_x - \frac{1}{2}\Delta E_x}^{E_x + \frac{1}{2}\Delta E_x} dE_x \rho(E_x, J, \Pi) T_{\alpha'l'j'}^J(E_{a'}) \quad (13.4)$$

where  $\rho$  is the level density, see Chapter 8, and  $T$  is evaluated at an emission energy  $E_{a'}$  that corresponds to the middle of the excitation energy bin, i.e.  $E_{a'} = E^{tot} - E_x - S_{a'}$ . Hence, both transitions to discrete states and transitions to the whole accessible continuum are covered by the

sum over  $\alpha'$  in Eq. (13.2). The normalization factor  $D^{comp}$  is

$$D^{comp} = [\sigma_{reac} - \sigma^{disc,direct} - \sigma^{PE}] / \sigma_{reac} \quad (13.5)$$

This indicates that in TALYS we assume that direct and compound contributions can be added incoherently. This formula for  $D^{comp}$  is only applied for weakly coupled channels that deplete the flux, such as contributions from DWBA or pre-equilibrium. In the case of coupled-channels calculations for the discrete collective states, the transmission coefficients of Eq. (13.2) are automatically reduced by ECIS-06 to account for direct effects and TALYS only subtracts the direct cross section for the weakly coupled levels (DWBA), i.e. if

$$\sigma^{disc,direct} = \sigma^{disc,cc} + \sigma^{disc,DWBA} \quad (13.6)$$

then

$$D^{comp} = [\sigma_{reac} - \sigma^{disc,DWBA} - \sigma^{PE}] / \sigma_{reac} \quad (13.7)$$

TALYS also computes the compound nucleus formula for the angular distribution. It is given by

$$\frac{d\sigma_{\alpha\alpha'}^{comp}(\theta)}{d\Omega} = \sum_L C_L^{comp} P_L(\cos\Theta), \quad (13.8)$$

where  $P_L$  are Legendre polynomials. The Legendre coefficients  $C_L^{comp}$  are given by

$$C_L^{comp} = D^{comp} \frac{\pi}{k^2} \sum_{J,\Pi} \frac{2J+1}{(2I+1)(2s+1)} \sum_{j=|J-I|}^{J+I} \sum_{l=|j-s|}^{j+s} \sum_{j'=|J-I'}^{J+I'} \sum_{l'=|j'-s'}^{j'+s'} \\ \times \delta_\pi(\alpha) \delta_\pi(\alpha') \frac{T_{\alpha l j}^J(E_a) \langle T_{\alpha' l' j'}^J(E_{a'}) \rangle}{\sum_{\alpha'', l'', j''} \delta_\pi(\alpha'') \langle T_{\alpha'' l'' j''}^J(E_{a''}) \rangle} W_{\alpha l j \alpha' l' j'}^J A_{ll' j l' j'; L}^J, \quad (13.9)$$

where the Blatt-Biedenharn factor  $A$  is given by

$$A_{ll' j l' j'; L}^J = \frac{(-1)^{l'-s'-I+s}}{4\pi} (2J+1)(2j+1)(2l+1)(2j'+1)(2l'+1) \\ (ll00|L0) \mathscr{W}(JjJj; lL) \mathscr{W}(jjll; Ls) (l'l'00|L0) \mathscr{W}(Jj'Jj'; l'L) \mathscr{W}(j'j'l'l'; Ls'), \quad (13.10)$$

where  $( \quad | \quad )$  are Clebsch-Gordan coefficients and  $\mathscr{W}$  are Racah coefficients.

Formulae (13.2) and (13.8-13.10) show that the width fluctuation correction factors and the angular distribution factors depend on all the angular momentum quantum numbers involved, and thus have to be re-evaluated each time inside all the summations. We generally need these formulae for relatively low incident energy, where the WFC has a significant impact and where the compound nucleus cross section to each individual discrete state is large enough to make its angular distribution of interest. For projectile energies above several MeV (we generally take the neutron separation energy for safety), the width fluctuations have disappeared, meaning that  $W_{\alpha l j \alpha' l' j'}^J = 1$  for all channels. Then for the angle-integrated compound cross section, instead of performing the full calculation, Eq. (13.2) can be decoupled into two parts that represent the incoming and outgoing reaction flux, respectively. It simplifies to

$$\sigma_{\alpha\alpha'}^{comp} = \sum_{J=\text{mod}(I+s,1)}^{l_{max}+I+s} \sum_{\Pi=-1}^1 \sigma_{J\Pi}^{CF}(E^{tot}) \frac{\Gamma_{\alpha'}(E^{tot}, J, \Pi \longrightarrow E_x, I', \Pi_f)}{\Gamma^{tot}(E^{tot}, J, \Pi)} \quad (13.11)$$

where  $\sigma_{J\Pi}^{CF}$  is the compound formation cross section per spin and parity:

$$\sigma_{J\Pi}^{CF}(E^{tot}) = D^{comp} \frac{\pi}{k^2} \frac{2J+1}{(2I+1)(2s+1)} \sum_{j=|J-I|}^{J+I} \sum_{l=|j-s|}^{j+s} T_{\alpha l j}^J(E_a) \delta_{\pi}(\alpha) \quad (13.12)$$

which itself obeys

$$\sum_{J=\text{mod}(I+s,1)}^{l_{\max}+I+s} \sum_{\Pi=-1}^1 \sigma_{J\Pi}^{CF}(E^{tot}) = D^{comp} \sigma_{\text{reac}} \quad (13.13)$$

The partial decay widths are

$$\Gamma_{\alpha'}(E^{tot}, J, \Pi \longrightarrow E_x, I', \Pi_f) = \frac{1}{2\pi\rho(E^{tot}, J, \Pi)} \sum_{j'=|J-I'|}^{J+I'} \sum_{l'=|j'-s'}^{j'+s'} \delta_{\pi}(\alpha') \langle T_{\alpha' l' j'}^J(E_{\alpha'}^l) \rangle \quad (13.14)$$

and the total decay width is

$$\Gamma^{tot}(E^{tot}, J, \Pi) = \sum_{\alpha''} \Gamma_{\alpha''}(E^{tot}, J, \Pi \longrightarrow E_x, I'', \Pi_f) \quad (13.15)$$

where we sum over all possible states in the residual nuclides through the sum over  $\alpha''$ . Note that the term with the compound nucleus level density,  $2\pi\rho$ , is present in both Eq. (13.14) and Eq. (13.15) and therefore does not need to be calculated in practice for Eq. (13.11). A formula similar to Eq. (13.11) is used for multiple emission, see Section 14.

In sum, we use Eqs. (13.2) and (13.9) if either width fluctuations (**widthfluc y**, (p. 549) or compound angular distributions (**outangle y**, (p. 534) are to be calculated and Eq. (13.11) if they are both not of interest.

A final note to make here is that the formulae of this whole Section can also be applied for excited (isomeric) target states.

## 13.2 Width fluctuation correction factor

The WFC factor  $W$  accounts for the correlations that exist between the incident and outgoing waves. From a qualitative point of view, these correlations enhance the elastic channel and accordingly decrease the other open channels. Above a few MeV of projectile energy, when many competing channels are open, the WFC factor can be neglected and the simple Hauser-Feshbach model is adequate to describe the compound nucleus decay. To explain the WFC factors, we now switch to a more compact notation in which we leave out  $J$  and define  $a = \{\alpha, l, j\}$  and  $b = \{\alpha', l', j'\}$ . With such a notation the compound nucleus cross section can be written in the compact form

$$\sigma_{ab} = \frac{\pi}{k_a^2} \frac{T_a T_b}{\sum_c T_c} W_{ab} \quad (13.16)$$

for each combination of  $a$  and  $b$ . In general, the WFC factor may be calculated using three different expressions, which have all been implemented in TALYS: The Hofmann-Richert-Tepel-Weidenmüller (HRTW) model [113, 114, 115], the Moldauer model [116, 117], and the model using the Gaussian Orthogonal Ensemble (GOE) of Hamiltonian matrices [118]. A comparison between the three models is given by Hilaire et al. [119].

For each expression, flux conservation implies that

$$T_a = \sum_b \frac{T_a T_b}{\sum_c T_c} W_{ab} \quad (13.17)$$

This equation can be used to check the numerical accuracy of the WFC calculation (see the **flagcheck** keyword in Part III).



### 13.2.1 The HRTW method

The simplest approach is the HRTW method. It is based on the assumption that the main effect of the correlation between incident and outgoing waves is in the elastic channel. In that case, it is convenient to express the compound nucleus cross section (13.16) as

$$\sigma_{ab} = \frac{\pi}{k^2} \frac{V_a V_b}{\sum_c V_c} [1 + \delta_{ab}(W_a - 1)], \quad (13.18)$$

where the  $V_i$ 's are effective transmission coefficients that take into account the correlations.

This expression means that only the elastic channel enhancement is described since for  $a = b$ , Eq. (13.18) becomes

$$\sigma_{aa} = \frac{\pi}{k_a^2} \frac{V_a^2}{\sum_c V_c} W_a \quad (13.19)$$

while for  $a \neq b$ ,

$$\sigma_{ab} = \frac{\pi}{k_a^2} \frac{V_a V_b}{\sum_c V_c} \quad (13.20)$$

An expression for the  $V_i$  values can be determined from the flux conservation condition

$$\sum_b \sigma_{ab} = \frac{\pi}{k_a^2} T_a, \quad (13.21)$$

which yields using Eq. (13.18)

$$T_a = V_a + (W_a - 1) \frac{V_a^2}{\sum_c V_c}, \quad (13.22)$$

or

$$V_a = \frac{T_a}{1 + \frac{(W_a - 1)V_a}{\sum_c V_c}}. \quad (13.23)$$

The only required information is thus the expression for  $W_a$ , which can be derived from an analysis using random matrix calculations. In TALYS, the expression of Ref. [115] is used. It reads

$$W_a = 1 + \frac{2}{1 + T_a^F} + 87 \left( \frac{T_a - \bar{T}}{\sum_c T_c} \right)^2 \left( \frac{T_a}{\sum_c T_c} \right)^5, \quad (13.24)$$

with  $\bar{T} = \frac{\sum_c T_c^2}{\sum_c T_c}$  and the exponent  $F = \frac{4 \frac{\bar{T}}{\sum_c T_c} \left( 1 + \frac{T_a}{\sum_c T_c} \right)}{1 + \frac{3\bar{T}}{\sum_c T_c}}$ .

The result for  $V_a$  is obtained after iterating Eq. (13.23) several times, starting from the initial value

$$V_a(i=0) = \frac{T_a}{1 + (W_a - 1) \frac{T_a}{\sum_c T_c}} \quad (13.25)$$

and calculating  $V_a(i+1)$  using

$$V_a(i+1) = \frac{T_a}{1 + (W_a - 1) \frac{V_a(i)}{\sum_c V_c(i)}} \quad (13.26)$$

until  $V_a(i+1) \approx V_a(i)$ . In a calculation, a few tens of iterations are generally required to reach a stable result.

For each  $J$  and  $\Pi$ , expressions (13.23)-(13.26) only need to be evaluated once. This is done in *hrtwprepare.f*, before all the loops over  $l, j, l'$  and  $j'$ , etc. quantum numbers are performed. For the calculation of  $W_{\alpha l j \alpha' l' j'}$  in Eq. (13.2), which takes place inside all loops, the correct  $V_a$  and  $V_b$  are then addressed. The WFC factor can then be derived from Eqs. (13.16) and (13.18),

$$W_{ab} = \frac{V_a V_b}{\sum_c V_c} [1 + \delta_{ab}(W_a - 1)] \frac{\sum_c T_c}{T_a T_b} \quad (13.27)$$

which is calculated in *hrtw.f*.

### 13.2.2 Moldauer expression

This is the default option for the WFC in TALYS. Moldauer's expression for  $W_{ab}$  is based on the assumption that a  $\chi^2$  law with  $\nu$  degrees of freedom applies for the partial widths  $\Gamma$ , which can be calculated from a Porter-Thomas distribution. These are associated with transmission coefficients as

$$T = \frac{2\pi \langle \Gamma \rangle}{D} \quad (13.28)$$

provided  $\langle \Gamma \rangle \ll D$ , where  $D$  is the mean level spacing. The WFC factor  $W_{ab}$  reads

$$W_{ab} = \left(1 + \frac{2\delta_{ab}}{\nu_a}\right) \int_0^{+\infty} \prod_c \left(1 + \frac{2T_c x}{\nu_c \sum_i T_i}\right)^{-(\delta_{ac} + \delta_{bc} + \nu_c/2)} dx \quad (13.29)$$

Moldauer has parameterised  $\nu$  using Monte Carlo calculations, giving

$$\nu_a = 1.78 + (T_a^{1.212} - 0.78) \exp\left(-0.228 \sum_c T_c\right) \quad (13.30)$$

In TALYS, the integral in Eq. (13.29) is evaluated numerically. For this, the Gauss-Laguerre method has been chosen and we find that 40 integration points are enough to reach convergence, the criterion being the flux conservation of Eq. (13.17). As for the HRTW model, the calculation can be split into parts dependent and independent of the channel quantum numbers. First, in *molprepare.f*, for each  $J$  and  $\Pi$ , we calculate Eq. (13.30) for all channels and the product  $\prod_c \left(1 + \frac{2T_c x}{\nu_c \sum_i T_i}\right)^{-\nu_c/2}$  that appears in Eq. (13.29). Inside all the loops, we single out the correct  $a$  and  $b$  channel and calculate Eq. (13.29) in *moldauer.f*.

Eq. (13.29) involves a product over all possible open channels. When the number of channels is large, the product calculation drastically increases the time of computation, forcing us to consider another method. Many open channels are considered for capture reactions and reactions to the continuum.

### A. Capture reactions

If the projectile is captured by the target nucleus, the compound nucleus is formed with an excitation energy at least equal to the projectile separation energy in the compound system. Since the  $\gamma$  transmission coefficient calculation involves all the possible states to which a photon can be emitted from the initial compound nucleus state, the number of radiative open channels is almost infinite, but each has a very small transmission coefficient. Following Ref. [120], the product over the radiative channels in Eq. (13.29) can be transformed as

$$\prod_{c \in \gamma} \left( 1 + \frac{2T_c x}{v_c \sum_i T_i} \right)^{-v_c/2} \approx \lim_{v_\gamma \rightarrow +\infty} \left( 1 + \frac{2T_\gamma x}{v_\gamma \sum_i T_i} \right)^{-v_\gamma/2} = \exp \left( -\frac{T_\gamma^{eff} x}{\sum_i T_i} \right) \quad (13.31)$$

where  $T_\gamma^{eff}$  is given by the procedure sketched in Section 9. The derivation is based on the hypothesis that all the individual  $T_\gamma$  are almost identical to 0. Therefore, to calculate  $W_{ab}$  when  $b$  denotes the gamma channel, we set  $T_b = 0$  in Eqs. (13.29) and use Eqs. (13.31) to calculate the product for  $\gamma$  channels.

### B. Continuum reactions

For high excitation energies, it is impossible to describe all the open channels individually. It is then necessary to introduce energy bins to discretize the continuum of levels and define continuum (or effective) transmission coefficients as

$$T_{eff}(U) = \int_{E_{min}}^{E_{max}} \rho(\varepsilon) T(\varepsilon) d\varepsilon, \quad (13.32)$$

where  $U$  is generally taken as the middle of the interval  $[E_{min}, E_{max}]$  and  $\rho$  is the density of levels under consideration. This effective transmission coefficient corresponds to an effective number of channels  $N_{eff}(U)$ , given by

$$N_{eff}(U) = \int_{E_{min}}^{E_{max}} \rho(\varepsilon) d\varepsilon. \quad (13.33)$$

Calculating the product term in Eq. (13.29) is tedious, unless one assumes that the energy variation of  $T(\varepsilon)$  is smooth enough to warrant that each of the  $N_{eff}(U)$  channels has the same average transmission coefficient

$$T_{mean}(U) = \frac{T_{eff}(U)}{N_{eff}(U)}. \quad (13.34)$$

Then, the product over the channels  $c$  belonging to such a continuum bin in the Moldauer integral Eq. (13.29) can be replaced by a single term, i.e.

$$\prod_c \left( 1 + \frac{2T_c}{v_c \sum_i T_i} x \right)^{-v_c/2} \approx \left( 1 + \frac{2T_{mean}(U)}{v_{mean} \sum_i T_i} x \right)^{-N_{eff}(U)v_{mean}/2}, \quad (13.35)$$

where

$$v_{mean} = 1.78 + (T_{mean}^{1.212} - 0.78) \exp \left( -0.228 \sum_c T_c \right) \quad (13.36)$$

### C. Fission reactions

The fission reaction is treated as one global channel, regardless of the nature of the fission fragments that result from fission. In Chapter 15, it is explained how the global fission transmission coefficient is calculated. It is however important to state here that the fission transmission coefficient is generally greater than 1 since it results from a summation over several fission paths and can therefore be defined as

$$T_{fis}(U) = \int_{E_{min}}^{E_{max}} \rho_{fis}(\epsilon) T_f(\epsilon) d\epsilon. \quad (13.37)$$

Of course, one has  $0 \leq T_f(\epsilon) \leq 1$ , but one can not assume that  $T_f$  is constant over the whole integration energy range as in the case of continuum reactions. To bypass this problem, instead of using a global fission transmission coefficient, we have grouped the various components of Eq. (13.37) according to their values. Instead of dealing with a global fission transmission coefficient, we use  $N$  different global transmission coefficients (where  $N$  is an adjustable parameter) such that

$$T_{fis}(U) = \sum_{i=0}^N T_{fis}(i, U) \quad (13.38)$$

where

$$T_{fis}(i, U) = \int_{E_{min}}^{E_{max}} \rho_{fis}(\epsilon) T_f(\epsilon) \delta_{i,N} d\epsilon \quad (13.39)$$

and  $\delta_{i,N} = 1$  is  $i/N \leq T_f(\epsilon) \leq (i+1)/N$  and 0 otherwise.

In this case one can define, as for continuum reactions, an effective number of channels  $N_{fis}(i, U)$ , and use  $N$  average fission transmission coefficients defined by

$$T_{fismean}(i) = \frac{T_{fis}(i, U)}{N_{fis}(i, U)}. \quad (13.40)$$

If  $N$  is large enough, these  $N$  average coefficients can be used for the width fluctuation calculation without making a too crude approximation.

### 13.2.3 The GOE triple integral

The two previously described methods to obtain  $W_{ab}$  are readily obtained since both are relatively simple to implement. However, in each case, a semi-empirical parameterisation is used. The GOE formulation avoids such a parameterisation, in which sense it is the more general expression. In the GOE approach,  $W_{ab}$  reads

$$W_{ab} = \frac{\sum_c T_c}{8} \int_0^{+\infty} d\lambda_1 \int_0^{+\infty} d\lambda_2 \int_0^1 d\lambda f(\lambda_1, \lambda_2, \lambda) \prod_c (\lambda_1, \lambda_2, \lambda) g_{ab}(\lambda_1, \lambda_2, \lambda) \quad (13.41)$$

with

$$f(\lambda_1, \lambda_2, \lambda) = \frac{\lambda(1-\lambda)|\lambda_1 - \lambda_2|}{\sqrt{\lambda_1 \lambda_2 (1+\lambda_1)(1+\lambda_2)} (\lambda + \lambda_1)^2 (\lambda + \lambda_2)^2}, \quad (13.42)$$

$$\prod_c (\lambda_1, \lambda_2, \lambda) = \prod_c \frac{1 - \lambda T_c}{\sqrt{(1 + \lambda_1 T_c)(1 + \lambda_2 T_c)}}, \quad (13.43)$$

and

$$g_{ab}(\lambda_1, \lambda_2, \lambda) = \delta_{ab}(1 - T_a) \left( \frac{\lambda_1}{1 + \lambda_1 T_a} + \frac{\lambda_2}{1 + \lambda_2 T_a} + \frac{2\lambda}{1 - \lambda T_a} \right)^2 + (1 + \delta_{ab}) \\ \times \left[ \frac{\lambda_1(1 + \lambda_1)}{(1 + \lambda_1 T_a)(1 + \lambda_1 T_b)} + \frac{\lambda_2(1 + \lambda_2)}{(1 + \lambda_2 T_a)(1 + \lambda_2 T_b)} + \frac{2\lambda(1 - \lambda)}{(1 - \lambda T_a)(1 - \lambda T_b)} \right] \quad (13.44)$$

The numerical method employed to compute this complicated triple integral is explained by Hilaire et al. [119].

Also here, a particular situation exists for the capture channel, where we set

$$\prod_{c \in \gamma} \frac{1 - \lambda T_c}{\sqrt{(1 + \lambda_1 T_c)(1 + \lambda_2 T_c)}} \approx \exp \left[ - (2\lambda + \lambda_1 + \lambda_2) T_\gamma^{eff} / 2 \right] \quad (13.45)$$

and for the continuum, for which we set

$$\prod_c (\lambda_1, \lambda_2, \lambda) = \prod_{c \in \text{continuum}} \frac{(1 - \lambda \bar{T}_c)^{N_c}}{\sqrt{(1 + \lambda_1 \bar{T}_c)^{N_c} (1 + \lambda_2 \bar{T}_c)^{N_c}}}. \quad (13.46)$$

Again, for each  $J$  and  $\Pi$ , the multiplications that do not depend on  $a$  or  $b$  are first prepared in *goeprepare.f*, while the actual WFC calculation takes place in *goef*.

**Options 13.2.1** With **widthfluc** (p. 549) the WFC calculation can be enabled and disabled. The various WFC models can be set with **widthmode** (550).

### Sample case 13.1 Comparison of compound nucleus WFC models: 10 keV n + <sup>93</sup>Nb

In this sample case, we demonstrate the difference between the various models for the width fluctuation correction in compound nucleus reactions, as discussed extensively in Ref. [119]. As sample case, we take 10 keV neutrons incident on <sup>93</sup>Nb and we ask for various compound nucleus models to calculate cross sections and angular distributions (**outangle y**), and to put the result for the elastic scattering angular distribution on a separate file, called *nn000.010ang.L00*. Since the GOE calculation (**widthmode 3**) is rather time-consuming, we reduce the number of bins to 20 for all cases. We wish to check whether the flux is conserved in the compound nucleus model for the various WFC models, so we set **outcheck y**. This means that for each set of quantum numbers, unitarity is checked by means of Eq. (13.17).

This sample case illustrates the capabilities of TALYS to simulate photonuclear reactions. We calculate the ( $\gamma, n$ ) reaction on <sup>90</sup>Zr as a function of incident energy, with default model parameters, and compare the result to experimental data. The following input file is used

```
#
# n-Nb093-HF
#
# General
#
projectile n
element nb
mass 93
```

Model	$\sigma^{comp-el}$
Hauser-Feshbach	2410.89 mb
Moldauer	2617.22 mb
HRTW	2752.25 mb
GOE	2617.12 mb

Table 13.1: Compound elastic cross section for 4 different compound nucleus models for 10 keV neutrons incident on  $^{93}\text{Nb}$ .

```
energy 0.01
#
# Parameters
#
bins 20
widthmode 0
#
# Output
#
outcheck y
outangle y
fileelastic y
```

This only new output block, i.e. not discussed before, is

```
+++++++ CHECK OF FLUX CONSERVATION OF TRANSMISSION COEFFICIENTS ++++++
Hauser-Feshbach model
```

```
Parity-- J= 3.0  j= 1.5  l= 1  T(j,l)= 7.98822E-03  Sum over outgoing channels
Parity-- J= 4.0  j= 0.5  l= 1  T(j,l)= 4.54100E-03  Sum over outgoing channels
Parity-- J= 4.0  j= 1.5  l= 1  T(j,l)= 7.98822E-03  Sum over outgoing channels
Parity-- J= 5.0  j= 0.5  l= 1  T(j,l)= 4.54100E-03  Sum over outgoing channels
Parity-- J= 5.0  j= 1.5  l= 1  T(j,l)= 7.98822E-03  Sum over outgoing channels
Parity-- J= 6.0  j= 1.5  l= 1  T(j,l)= 7.98822E-03  Sum over outgoing channels
.....
```

in which the aforementioned unitarity is checked. For the Moldauer model the above input file has **widthmode 1** For the HRTW model the above input file has **widthmode 2**. For the GOE model the above input file has **widthmode 3**.

Table 13.1 lists the obtained compound nucleus elastic cross section for the 4 cases.

Fig. 13.1 displays the elastic angular distribution for the 4 models. Results like these made us conclude in Ref. [119] that Moldauer's model, which is closest to the exact GOE result, is the one to use in practical applications, especially when considering the long calculation time of the GOE model. Obviously, this sample case can be extended to one with various incident energies, so that the differences between excitation functions can be studied, see also Ref. [119].

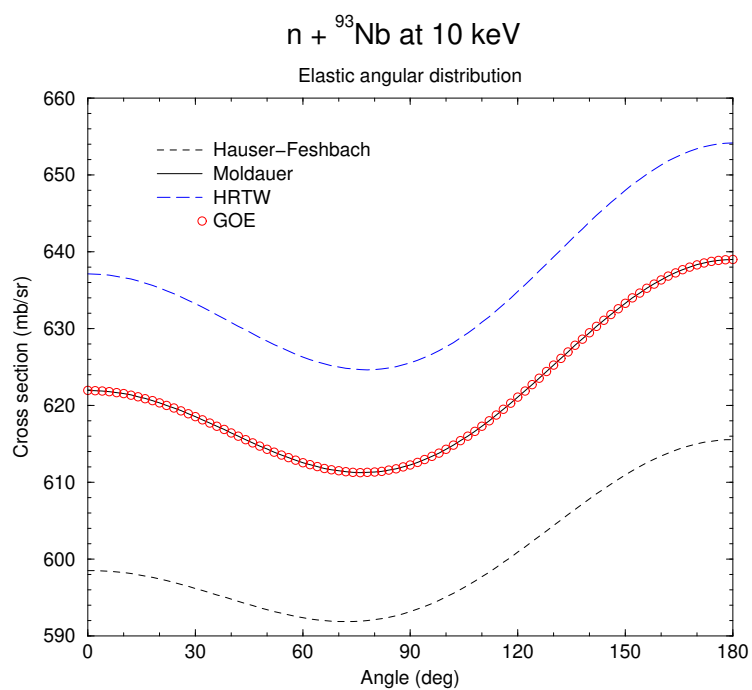


Figure 13.1: Total elastic angular distribution for 4 different compound nucleus models for 10 keV neutrons incident on  ${}^{93}\text{Nb}$ .





## 14. Multiple emission

At incident energies above approximately the neutron separation energy, the residual nuclides formed after the first binary reaction are populated with enough excitation energy to enable further decay by particle emission or fission. This is called multiple emission. We distinguish between two mechanisms: multiple compound (Hauser-Feshbach) decay, which comes important as soon as the incident energy is higher than the neutron separation energy and multiple pre-equilibrium decay, which becomes important above 30-40 MeV of incident energy.

### 14.1 Multiple Hauser-Feshbach decay

This is the conventional way, and for incident energies up to several tens of MeV a sufficient way, to treat multiple emission. It is assumed that pre-equilibrium processes can only take place in the binary reaction and that secondary and further particles are emitted by compound emission.

After the binary reaction, the residual nucleus may be left in an excited discrete state  $i'$  or an excited state within a bin  $i'$  which is characterized by excitation energy  $E_x'(i')$ , spin  $I'$  and parity  $\Pi'$ . The population of this state or set of states is given by a probability distribution for Hauser-Feshbach decay  $P^{\text{HF}}$  that is completely determined by the binary reaction mechanism. For a binary neutron-induced reaction to a discrete state  $i'$ , i.e. when  $E_x'(i')$ ,  $I'$  and  $\Pi'$  have unique values, the residual population is given by

$$P^{\text{HF}}(Z', N', E_x'(i'), I', \Pi') = \sigma_{n,k'}^i(E^{\text{tot}}, I, \Pi \rightarrow E_x'(i'), I', \Pi'), \quad (14.1)$$

where the non-elastic reaction cross section for a discrete state  $\sigma_{n,k'}^i$  was defined in Section 4.2.3 and where the ejectile  $k'$  connects the initial compound nucleus  $(Z_C, N_C)$  and the residual nucleus  $(Z', N')$ . For binary reactions to the continuum, the residual population of states characterised by

$(I', \Pi')$  per  $E_{x'}(i')$  bin is given by the sum of a pre-equilibrium and a compound contribution

$$P^{\text{HF}}(Z', N', E_{x'}(i'), I', \Pi') = \int dE_{k'} \frac{d\sigma^{\text{comp, cont}}}{dE_{k'}}(E^{\text{tot}}, I, \Pi \rightarrow E_{x'}(i'), I', \Pi') \\ + \mathcal{P}^{\text{pre}}(Z', N', p_{\pi}^{\text{max}} + 1, h_{\pi}^{\text{max}} + 1, p_{\nu}^{\text{max}} + 1, h_{\nu}^{\text{max}} + 1, E_{x'}(i')), \quad (14.2)$$

where the integration range over  $dE_{k'}$  corresponds exactly with the bin width of  $E_{x'}(i')$  and  $\mathcal{P}^{\text{pre}}$  denotes the population entering the compound stage after primary preequilibrium emission. The expression for  $\mathcal{P}^{\text{pre}}$  will be given in Eq. (14.6) of the next section. Once the first generation of residual nuclides/states has been filled, the picture can be generalized to tertiary and higher order multiple emission.

In general, the population  $P^{\text{HF}}$  before decay of a level  $i'$  or a set of states  $(I', \Pi', E_{x'}(i'))$  in bin  $i'$  of a nucleus  $(Z', N')$  in the reaction chain is proportional to the feeding, through the ejectiles  $k'$ , from all possible mother bins  $i$  with an energy  $E_x(i)$  in the  $(Z, N)$  nuclides, i.e.

$$P^{\text{HF}}(Z', N', E_{x'}(i'), I', \Pi') = \sum_{I, \Pi} \sum_{k'} \sum_i [P^{\text{HF}}(Z, N, E_x(i), I, \Pi) \\ + \mathcal{P}^{\text{pre}}(Z, N, p_{\pi}^{\text{max}} + 1, h_{\pi}^{\text{max}} + 1, p_{\nu}^{\text{max}} + 1, h_{\nu}^{\text{max}} + 1, E_x(i))] \\ \times \frac{\Gamma_{k'}(E_x(i), I, \Pi \rightarrow E_{x'}(i'), I', \Pi')}{\Gamma^{\text{tot}}(E_x(i), I, \Pi)}. \quad (14.3)$$

The appearance of the indices  $p_{\pi}^{\text{max}}$  indicates that only the reaction population that has not been emitted via the (multiple) pre-equilibrium mechanism propagates to the multiple compound stage. Similar to Eq. (13.14) the decay widths are given by

$$\Gamma_{k'}(E_x(i), I, \Pi \rightarrow E_{x'}(i'), I', \Pi') = \frac{1}{2\pi\rho(E_x(i), I, \Pi)} \sum_{j'=|J-I'|}^{J+I'} \sum_{l'='=|j'-s'|}^{j'+s'} \delta_{\pi}(\alpha') \langle T_{\alpha' l' j'}^J(E_{d'}) \rangle. \quad (14.4)$$

Again, the term  $2\pi\rho$  (compound nucleus level density) of the decay width (14.4) falls out of the multiple emission equation (14.3) and therefore does not need to be calculated in practice. The total decay width is

$$\Gamma^{\text{tot}}(E_x(i), I, \Pi) = \sum_{k''} \sum_{l''=\text{mod}(J+s, 1)}^{J+l_{\text{max}}} \sum_{\Pi''=-1}^1 \sum_{i''} \Gamma_{k''}(E_x(i), I, \Pi \rightarrow E_{x''}(i''), I'', \Pi''). \quad (14.5)$$

In sum, the only differences between binary and multiple compound emission are that width fluctuations and angular distributions do not enter the model and that the initial compound nucleus energy  $E^{\text{tot}}$  is replaced by an excitation energy bin  $E_x$  of the mother nucleus. The calculational procedure, in terms of sequences of decaying bins, was already explained in Chapter 4.

**Options 14.1.1** With **maxZ** (p. 288) and **maxZ** (p. 288) you can set how many residual nuclides are taken into account.

## 14.2 Multiple pre-equilibrium emission

At high excitation energies, resulting from high incident energies, the composite nucleus is far from equilibrated and it is obvious that the excited nucleus should be described by more degrees of freedom than just  $E_x$ ,  $J$  and  $\Pi$ . In general, we need to keep track of the particle-hole configurations that are excited throughout the reaction chain and thereby calculate multiple pre-equilibrium emission up to any order. This is accomplished by treating multiple pre-equilibrium emission

within the exciton model. This is the default option for multiple pre-equilibrium calculations in TALYS (selected with the keyword **mpreeqmode 1**). TALYS contains, furthermore, an alternative more approximative model for multiple pre-equilibrium emission (**mpreeqmode 2**), called the *s-wave transmission coefficient method*. Both approaches are discussed below.

### 14.2.1 Multiple pre-equilibrium emission within the exciton model

We introduce the pre-equilibrium population  $\mathcal{P}^{\text{pre}}(Z, N, p_\pi, h_\pi, p_\nu, h_\nu, E_x(i))$  which holds the amount of the reaction population present in a unique  $(Z, N)$  nucleus,  $(p_\pi, h_\pi, p_\nu, h_\nu)$  exciton state and excitation energy bin  $E_x(i)$ . A special case is the pre-equilibrium population for a particular exciton state after binary emission, which can be written as

$$\begin{aligned} & \mathcal{P}^{\text{pre}}(Z', N', p_\pi - Z_{k'}, h_\pi, p_\nu - N_{k'}, h_\nu, E_{x'}(i')) = \\ & = \sigma^{\text{CF}}(Z_C, N_C, E^{\text{tot}}) W_{k'}(Z_C, N_C, E^{\text{tot}}, p_\pi, h_\pi, p_\nu, h_\nu, E_{k'}) \\ & \times \tau(Z_C, N_C, E^{\text{tot}}, p_\pi, h_\pi, p_\nu, h_\nu) P(Z_C, N_C, E^{\text{tot}}, p_\pi, h_\pi, p_\nu, h_\nu), \end{aligned} \quad (14.6)$$

where  $Z_C$  ( $N_C$ ) again is the compound nucleus charge (neutron) number and  $Z_{k'}$  ( $N_{k'}$ ) corresponds to the ejectile charge (neutron) number. The residual excitation energy  $E_{x'}(i')$  is linked to the total energy  $E^{\text{tot}}$ , the ejectile energy  $E_{k'}$ , and its separation energy  $S(k')$  by  $E_{x'}(i') = E^{\text{tot}} - E_{k'} - S(k')$ . This  $\mathcal{P}^{\text{pre}}$  represents the feeding term for secondary pre-equilibrium emission. Note that for several particle-hole configurations this population is equal to zero.

In general, the pre-equilibrium population can be expressed in terms of the mother nucleus, excitation energy bins, and particle-hole configurations from which it is fed. The residual population is given by a generalization of Eq. (12.1), in which  $\sigma^{\text{CF}}(Z_C, N_C, E^{\text{tot}})$  is replaced by the population of the particle-hole states left after the previous emission stage  $\mathcal{P}^{\text{pre}}(Z, N, p_\pi^0, h_\pi^0, p_\nu^0, h_\nu^0, E_x(i))$ . Since several combinations of emission and internal transitions may lead to the same configuration, a summation is applied over the ejectiles treated in multiple pre-equilibrium (neutrons and protons), over the  $(p_\pi^0, h_\pi^0, p_\nu^0, h_\nu^0)$  configurations with which the next step is started and over the mother excitation energy bins:

$$\begin{aligned} \mathcal{P}^{\text{pre}}(Z', N', p'_\pi, h'_\pi, p'_\nu, h'_\nu, E_{x'}(i')) & = \sum_{k'=n,p} \sum_{p_\pi^0=1}^{p_\pi^{\text{max}}} \sum_{h_\pi^0=1}^{h_\pi^{\text{max}}} \sum_{p_\nu^0=1}^{p_\nu^{\text{max}}} \sum_{h_\nu^0=1}^{h_\nu^{\text{max}}} \\ & \sum_i \mathcal{P}^{\text{pre}}(Z, N, p_\pi^0, h_\pi^0, p_\nu^0, h_\nu^0, E_x(i)) \\ & W_k(Z, N, p_\pi, h_\pi, p_\nu, h_\nu, E_x(i), E_{k'}) \tau(Z, N, p_\pi, h_\pi, p_\nu, h_\nu, E_x(i)) \\ & \times P(Z, N, p_\pi, h_\pi, p_\nu, h_\nu, E_x(i)), \end{aligned} \quad (14.7)$$

where the mother and daughter quantities are related by

$$\begin{aligned} Z & = Z' + Z_{k'}, \\ N & = N' + N_{k'}, \\ p_\pi & = p'_\pi + Z_{k'}, \\ h_\pi & = h'_\pi, \\ p_\nu & = p'_\nu + N_{k'}, \\ h_\nu & = h'_\nu, \\ E_x & = E_{x'}(i') + E_{k'} + S_{k'}. \end{aligned} \quad (14.8)$$

In the computation, we thus need to keep track of every possible  $(Z', N', p'_\pi, h'_\pi, p'_\nu, h'_\nu, E_{x'}(i'))$  configuration, which is uniquely linked to a mother exciton state  $(Z, N, p_\pi, h_\pi, p_\nu, h_\nu, E_x(i))$  through

the ejectile characterized by  $(Z_{k'}, N_{k'}, E_{k'})$ . The term  $P(Z, N, p_\pi, h_\pi, p_\nu, h_\nu, E_x(i))$  represents the part of the pre-equilibrium cross section that starts in  $(Z, N, p_\pi^0, h_\pi^0, p_\nu^0, h_\nu^0, E_x(i))$  and survives emission up to a new particle-hole state  $(Z, N, p_\pi, h_\pi, p_\nu, h_\nu, E_x(i))$ . Again (see Eq. (12.16)),

$$P(Z, N, p_\pi^0, h_\pi^0, p_\nu^0, h_\nu^0, E_x(i)) = 1, \quad (14.9)$$

and the calculation for each newly encountered  $(Z, N, p_\pi, h_\pi, p_\nu, h_\nu, E_x(i))$  configuration proceeds according to Eq. (12.14).

The part of  $\mathcal{P}^{\text{pre}}$  that does not feed a new multiple pre-equilibrium population automatically goes to the multiple Hauser-Feshbach chain of Eq. (14.3).

The final expression for the multiple pre-equilibrium spectrum is very similar to Eq. (12.1)

$$\begin{aligned} \frac{d\sigma_k^{\text{MPE}}}{dE_{k'}} &= \sum_{p_\pi^0=1}^{p_\pi^{\text{max}}} \sum_{h_\pi^0=1}^{h_\pi^{\text{max}}} \sum_{p_\nu^0=1}^{p_\nu^{\text{max}}} \sum_{h_\nu^0=1}^{h_\nu^{\text{max}}} \\ &\sum_i \mathcal{P}^{\text{pre}}(Z, N, p_\pi^0, h_\pi^0, p_\nu^0, h_\nu^0, E_x(i)) \\ &\sum_{p_\pi=p_\pi^0}^{p_\pi^{\text{max}}} \sum_{h_\pi=h_\pi^0}^{h_\pi^{\text{max}}} \sum_{p_\nu=p_\nu^0}^{p_\nu^{\text{max}}} \sum_{h_\nu=h_\nu^0}^{h_\nu^{\text{max}}} W_k(Z, N, p_\pi, h_\pi, p_\nu, h_\nu, E_x(i), E_{k'}) \\ &\times \tau(Z, N, p_\pi, h_\pi, p_\nu, h_\nu, E_x(i)) P(Z, N, p_\pi, h_\pi, p_\nu, h_\nu, E_x(i)) \end{aligned} \quad (14.10)$$

### 14.2.2 Multiple pre-equilibrium emission with the s-wave transmission coefficient method

Apart from the exciton model, TALYS offers another, slightly faster, method to determine multiple pre-equilibrium emission by Chadwick et al. [121, 122]. Within this approach the multiple pre-equilibrium spectrum is given by the following expression:

$$\begin{aligned} \frac{d\sigma_k^{\text{MPE}}}{dE_{k'}} &= \sum_{p_\pi^0=1}^{p_\pi^{\text{max}}} \sum_{h_\pi^0=1}^{h_\pi^{\text{max}}} \sum_{p_\nu^0=1}^{p_\nu^{\text{max}}} \sum_{h_\nu^0=1}^{h_\nu^{\text{max}}} \\ &\sum_i \mathcal{P}^{\text{pre}}(Z, N, p_\pi^0, h_\pi^0, p_\nu^0, h_\nu^0, E_x(i)) \\ &\frac{1}{p_\pi^0 + p_\nu^0} \frac{\omega(Z_{k'}, h_\pi^0, N_{k'}, h_\nu^0, E_{k'} + S_{k'}) \omega(p_\pi^0 - Z_{k'}, h_\pi^0, p_\nu^0 - N_{k'}, h_\nu^0, E_x(i) - E_{k'} - S_{k'})}{\omega(p_\pi^0, h_\pi^0, p_\nu^0, h_\nu^0, E_x(i))} \\ &\times T_s(E_{k'}) \end{aligned} \quad (14.11)$$

In this approach each residual particle-hole configuration created in the primary pre-equilibrium decay may have one or more excited particles in the continuum. Each of these excited particles can either be emitted or captured. The emission probability is assumed to be well represented by the s-wave transmission coefficient  $T_s(E_{k'})$ .

## 15. Fission

The probability that a nucleus fissions can be estimated by TALYS on both phenomenological and microscopic grounds. Cross sections for (multi-chance) fission can be calculated. For this, various nuclear quantities are required.

### 15.1 Level densities for fission barriers

The level density formulae given in Section 8 for the ground state of the nucleus can all be applied for the fission barriers. In general, only the ingredients for a few level density expressions change as compared to the non-fissile case. In TALYS, two methods for fission level densities are programmed. The level densities are used in the calculation of fission transmission coefficients.

#### 15.1.1 Explicit treatment of collective effects

Eq. (8.89) for the rotational enhancement also holds for inner barriers with neutron number  $N \leq 144$ , which are all assumed to be axially symmetric (specified by the keyword **axtype 1**). Inner barriers with  $N > 144$ , e.g. the Am-isotopes, are taken to be axially asymmetric (**axtype 3**), and in that case the rotational enhancement is

$$K_{rot}(E_x) = K_{rot}^{asym}(E_x, \beta_2) = \max([\sqrt{\frac{\pi}{2}} \sigma_{\perp}^2 (1 - \frac{2\beta_2}{3}) \sigma_{\parallel} - 1] f(E_x) + 1, 1). \quad (15.1)$$

For outer barriers, we apply an extra factor of 2 to  $K_{rot}$ , due to the mass asymmetry. For all fission barriers, the default parameters for the damping function Eq. (8.90) are  $U_f^{bar} = 45$  MeV,  $C_f^{bar} = 5$  MeV.

The shell correction is also different: For the inner barrier,  $\delta W = 1.5$  MeV for an axially symmetric barrier, and 2.5 MeV otherwise. For the other barriers, we take 0.6 MeV in general.

### 15.1.2 Effective treatment of collective effects

Despite being of a more “effective” nature than the approach described above, this method (invoked with **colldamp y**) has been more successful in the description of fission cross sections, see e.g. [123]. An essential aspect is that the damping of collective effects are taken into account in a phenomenological way through the level density parameter  $a$ . The asymptotic level density parameter  $\tilde{a}$ , see Eq. (8.13) is damped from its effective limit  $A/8$  to its intrinsic limit  $A/13$  as follows

$$\tilde{a}^{\text{eff}} = \frac{A}{13} f(E_x) + \tilde{a}(1 - f(E_x)) \quad (15.2)$$

where

$$f(E_x) = \frac{1}{1 + \exp\left(-\frac{E_x - E_x^{g.s.}}{d_{col}^{g.s.}}\right)}, \quad (15.3)$$

with the same values as mentioned below Eq. (8.90). Next, the resulting  $a(E_x)$  is used in all equations. After this, an extra rotational enhancement needs only to be taken into account for tri-axial barriers (**axtype 2**). Instead of Eq. (8.89), this is taken as

$$K_{rot}(E_x) = \left(\frac{U}{a_{\text{eff}}}\right)^{1/4} (1 - f(E_x)) + f(E_x), \quad (15.4)$$

where  $a_{\text{eff}} = 8a/13$  and

$$f(E_x) = \frac{1}{1 + \exp\left(-\frac{1}{2}(E_x - 18)\right)}, \quad (15.5)$$

For barriers other than tri-axial,  $K_{rot} = 1$ . There is no vibrational enhancement in this model. For the shell correction, for all barriers  $\delta W = \frac{2}{3} |\delta W^{g.s.}|$ . Finally, the spin cut-off parameter (8.28) is multiplied by  $\left(1 + \frac{\beta_2}{3}\right)$  as done in Eq. (8.86) for the perpendicular spin cut-off parameter.

## 15.2 Fission transmission coefficients

For fission, the default model implemented in TALYS is based on the transition state hypothesis of Bohr and the Hill-Wheeler expression. This yields transmission coefficients that enter the Hauser-Feshbach model to compete with the particle and photon transmission coefficients.

### 15.2.1 Transmission coefficient for one fission barrier

The Hill-Wheeler expression gives the probability of tunneling through a barrier with height  $B_f$  and width  $\hbar\omega_f$  for a compound nucleus with excitation energy  $E_x$ . It reads

$$T_f(E_x) = \frac{1}{1 + \exp\left[-2\pi \frac{(E_x - B_f)}{\hbar\omega_f}\right]} \quad (15.6)$$

For a transition state with excitation energy  $\varepsilon_i$  above the top of the same barrier, one has

$$T_f(E_x, \varepsilon_i) = \frac{1}{1 + \exp\left[-2\pi \frac{(E_x - B_f - \varepsilon_i)}{\hbar\omega_f}\right]} \quad (15.7)$$

which means that the barrier is simply shifted up by  $\varepsilon_i$ .

For a compound nucleus with excitation energy  $E_x$ , spin  $J$ , and parity  $\Pi$ , the total fission transmission coefficient is the sum of the individual transmissions coefficients for each barrier through which the nucleus may tunnel, and thus reads in terms of the previously introduced  $T_f(E_x, \varepsilon_i)$

$$T_f^{J,\Pi}(E_x) = \sum_i T_f(E_x, \varepsilon_i) f(i, J, \Pi) + \int_{E_{th}}^{E_x} \rho(\varepsilon, J, \Pi) T_f(E_x, \varepsilon) d\varepsilon \quad (15.8)$$

The summation runs over all discrete transition states on top of the barrier and  $E_{th}$  marks the beginning of the continuum. In this equation,  $f(i, J, \Pi) = 1$  if the spin and parity of the transition state equal that of the compound nucleus and 0 otherwise. Moreover,  $\rho(\varepsilon, J, \Pi)$  is the level density of fission channels with spin  $J$  and parity  $\Pi$  for an excitation energy  $\varepsilon$ . The main difference with the usually employed expressions is that the upper limit in the integration is finite. This expression also enables to define the number of fission channels by replacing  $T_f(E_x, \varepsilon_i)$  by 1 in Eq. (15.8). This is needed for width fluctuation calculations where the fission transmission coefficient is treated as a continuum transmission coefficient.

### 15.3 Transmission coefficient for multi-humped barriers

For double humped barriers, the generally employed expression is based on an effective transmission coefficient  $T_{eff}$  defined by

$$T_{eff} = \frac{T_A T_B}{T_A + T_B} \quad (15.9)$$

where  $T_A$  and  $T_B$  are the transmission coefficients for barrier  $A$  and  $B$  respectively, calculated with Eq. (15.8).

If a triple humped barrier needs to be considered, the expression for  $T_{eff}$  reads

$$T_{eff} = \frac{T_{AB} T_C}{T_{AB} + T_C} \quad (15.10)$$

where  $T_{AB}$  is given by Eq. (15.9). Consequently, the expression used in TALYS reads

$$T_{eff} = \frac{T_A T_B T_C}{T_A T_B + T_A T_C + T_B T_C} \quad (15.11)$$

For any number of barrier, the effective number of fission channels is calculated as in the case for one barrier [119].

### 15.4 Class II/III states

Class II (resp. III) states may be introduced when double (resp. triple) humped barriers are considered. In the particular situation where the excitation energy  $E_{CN}$  of the compound nucleus is lower than the barrier heights, fission transmission coefficients display resonant structures which are due to the presence of nuclear excited levels (Class II) in the second, or in the third (Class III) well of the potential energy surface. When such resonant structures occur, the expression for the effective fission transmission coefficient has to be modified (generally enhanced).

The way this resonant effect is determined depends on the number of barriers that are considered.

### 15.4.1 Double humped fission barrier

In the case where two barriers occur, the effective fission transmission coefficient  $T_{eff}$  can be written as

$$T_{eff} = \frac{T_A T_B}{T_A + T_B} \times F_{AB}(E_{CN}) \quad (15.12)$$

where  $F_{AB}(E_{CN})$  is a factor whose value depends on the energy difference between the excitation energy of the nucleus and that of the a class II state located in the well between barrier  $A$  and  $B$ . It has been shown [124] that the maximum value of  $F_{AB}(E)$  reaches  $\frac{4}{T_A + T_B}$  and gradually decreases over an energy interval defined as the width  $\Gamma_{II}$  of the class II state with excitation energy  $E_{II}$ . This is accounted for using the empirical quadratic expression

$$F_{AB}(E) = \frac{4}{T_A + T_B} + \left( \frac{E - E_{II}}{0.5\Gamma_{II}} \right)^2 \times \left( 1 - \frac{4}{T_A + T_B} \right) \quad (15.13)$$

if  $E_{II} - 0.5\Gamma_{II} \leq E \leq E_{II} + 0.5\Gamma_{II}$  and  $F_{AB} = 1$  otherwise.

Theoretically, this expression is valid for the tunneling through a single double humped barrier whereas in realistic situations, both  $T_A$  and  $T_B$  are obtained from a summation over several transition states. One may thus have large  $T_A$  and  $T_B$  values so that Eq. 15.13 may give  $F_{AB}(E) \leq 1$ . Such a situation can only occur for high enough excitation energies for which the individual Hill-Wheeler contributions in Eq. (15.8) are large enough. However, in TALYS, we only consider class II states with excitation energies lower than the height of the first barrier. Consequently, the resonant effect can only occur if the compound nucleus energy  $E_{CN}$  is (i) lower than the top of the first barrier and (ii) close to a resonant class II state ( $E_{II} - 0.5\Gamma_{II} \leq E_{CN} \leq E_{II} + 0.5\Gamma_{II}$ ). With such requirements, the individual Hill-Wheeler terms are clearly small, and  $T_A + T_B \ll 1$ .

### 15.4.2 Triple humped fission barrier

If three barriers  $A$ ,  $B$  and  $C$  are considered, the situation is more complicated. In this case, three situations can occur depending on the positions of the class II and class III states. Indeed the enhancement can be due either to a class II state or to a class III state, but on top of that, a double resonant effect can also occur if both a class II and a class III state have an excitation energy close to the compound nucleus energy. For any situation, the enhancement is first calculated for the first and the second barrier giving the transmission coefficient

$$T_{eff}^{AB} = T_{AB} \times F_{AB} \quad (15.14)$$

with  $F_{AB}$  given by Eq. (15.13) as in the previous case.

Next, the eventual coupling with a class III state with energy  $E_{III}$  of width  $\Gamma_{III}$  is accounted for by generalizing Eq. (15.12) writing

$$T_{eff}^{ABC} = \frac{T_{eff}^{AB} T_C}{T_{eff}^{AB} + T_C} \times F_{ABC}(E_{CN}) \quad (15.15)$$

where  $F_{ABC}(E_{CN})$  is given by generalizing Eq. (15.13) writing

$$F_{ABC}(E) = \frac{4}{T_{eff}^{AB} + T_C} + \left( \frac{E - E_{III}}{0.5\Gamma_{III}} \right)^2 \times \left( 1 - \frac{4}{T_{eff}^{AB} + T_C} \right) \quad (15.16)$$



if  $E_{III} - 0.5\Gamma_{III} \leq E \leq E_{III} + 0.5\Gamma_{III}$  and  $F_{ABC} = 1$  otherwise.

## 15.5 Fission barrier parameters

In TALYS several options are included for the choice of the fission barrier parameters:

1. Experimental parameters [6]: collection of a large set of actinide fission barrier heights and curvatures for both the inner and outer barrier based on a fit to experimental data, compiled by V. Maslov. Moreover, this compilation contains head band transition states.
2. Mamdouh parameters [125]: set of double-humped fission barrier heights for numerous isotopes derived from Extended Thomas-Fermi plus Strutinsky Integral calculations.
3. Rotating-Finite-Range Model (RFRM) by Sierk [126]: single-humped fission barrier heights are determined within a rotating liquid drop model, extended with finite-range effects in the nuclear surface energy and finite surface-diffuseness effects in the Coulomb energy.
4. Rotating-Liquid-Drop Model (RLDM) by Cohen *et al* [127].

In the current version of TALYS, the dependence on angular momentum of the fission barriers is discarded. If LDM barriers are employed in the calculation, they are corrected for the difference between the ground-state and fission barrier shell correction energy:

$$B_f^{LDM}(T) = B_f^{LDM}(0) - (\delta W_{groundstate} - \delta W_{barrier}) * g(T) \quad (15.17)$$

This correction gradually disappears with temperature due to the washing out of the shell effects [128]:

$$g(T) = \begin{cases} 1 & \text{for } T < 1.65\text{MeV}, \\ g(T) = 5.809 \exp(-1.066 T) & \text{for } T \geq 1.65\text{MeV}. \end{cases} \quad (15.18)$$

Shell corrections on top of the fission barrier are generally unknown. They obviously play an important role for the level density as well. Default values are adopted: for subactinides  $\delta W_{barrier} = 0$  MeV, for actinides  $\delta W_{barrier,inner} = 2.5$  MeV and  $\delta W_{barrier,outer} = 0.6$  MeV [6].

## 15.6 WKB approximation

As an alternative to the Hill-Wheeler approach, it is also possible to use the WKB approximation to calculate fission transmission coefficients. We use an implementation by Mihaela Sin and Roberto Capote and refer to Ref. [129] for the full details of this method. It can be invoked with the keyword **fismodel 5**.

### Sample case 15.1 Fission cross sections: $n + {}^{232}\text{Th}$ : WKB approach

This sample case shows the WKB approximation to calculate fission transmission coefficients as outlined in Ref. [129] and Ref. [Goriely2011]. It can be invoked with the keyword **fismodel 5**.

We use the following input file,

```
#
# n-Th232-fis
#
# General
#
projectile n
element Th
```

```
mass      232
ltarget   000
energy    energies
maxlevelstar 30
partable  y
bins      40
ejectiles g n
#
# do not use best parameters from database
#
best      n
#
# set multi-preequilibrium switch lower for actinides
#
multipreeq 6.
#
# output of extra channels
#
channels  y
filechannels y
#
# reduce output for activation-only calculation
#
outspectra n
outangle  n
ddxmode  0
outdiscrete n
maxrot   4
strength 9
strengthm1 3
ngfit    y
upbend   y
ldmodel  5
fismodel 5
fispartdamp y
hbstate  n
class2   n
ecissave y
eciscalc y
inccalc  y
outdiscrete y
riplrisk y
#
# Current variables of TASMAN Run: 2509
#
```

```

ctable      90 233      1.35146 1
ptable      90 233 -1.94808E+00 1
ctable      90 233 -9.41013E-01 2
ptable      90 233      1.01594 2
ctable      90 233      3.16960 3
ptable      90 233 -1.14439E+00 3
betafiscor  90 233      1.04349
vfiscor     90 233      0.71170
ctable      90 232      0.32630 1
ptable      90 232      2.77492 1
ctable      90 232      8.34473 2
ptable      90 232 -6.59883E+00 2
ctable      90 232      0.69898 3
ptable      90 232      0.75241 3
betafiscor  90 232      0.50505
vfiscor     90 232      0.84379
ctable      90 231  5.77646E-02 1
ptable      90 231 -2.67866E+00 1
ctable      90 231 -3.61946E+00 2
ptable      90 231 -2.25936E+00 2
ctable      90 231 -2.85851E+00 3
ptable      90 231      2.66753 3
betafiscor  90 231      1.58614
vfiscor     90 231      1.62104
ctable      90 230 -3.18402E+00 1
ptable      90 230 -4.20179E+00 1
ctable      90 230      1.13153 2
ptable      90 230 -5.43220E-01 2
ctable      90 230      1.01315 3
ptable      90 230 -2.80603E+00 3
betafiscor  90 230      1.66138
vfiscor     90 230      1.71450

```

The above parameters are the usual ones to be adjusted to get good agreement with experimental data: fission parameters and level density parameters on top of the barriers. The resulting file *fission.tot* is plotted together with experimental data in Fig. 15.1.

## 15.7 Fission fragment properties

Since TALYS-1.6, two models for fission fragments are available, the temperature dependent Brosa model, and the GEF model.

### 15.7.1 GEF model

A fission model has been developed model by Schmidt-Jurado [130] It is based on the statistical population of states in the fission valleys at the moment of dynamical freeze-out, which is specific to

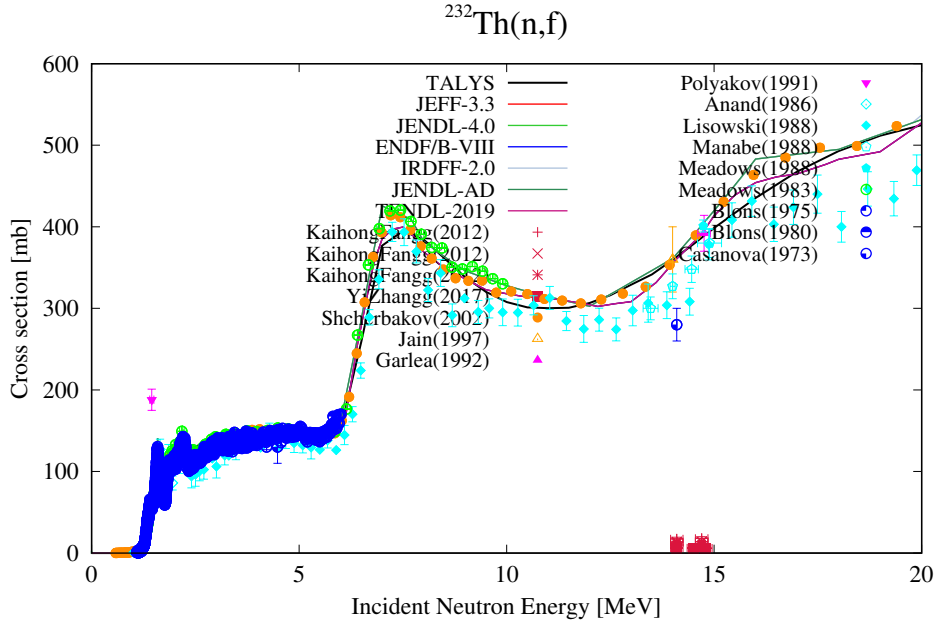


Figure 15.1: Neutron induced fission cross section of  $^{232}\text{Th}$  compared with experimental data. Calculation with fismodel 5 (WKB)

each collective degree of freedom. Three fission channels are considered. The separability principle governs the interplay of macroscopic and microscopic effects. The newly discovered energy-sorting mechanism determines the division of intrinsic excitation energy between the fragments at scission and the creation of a strong even-odd effect at large mass asymmetry. This model gives a new insight into several dynamical times. Most parameters are fixed from independent sources, only about 20 parameters have specifically been adjusted. Since the parameters of the model are closely related to physical properties of the systems, valuable conclusions on the fission process can be deduced. The good reproduction of measured data and the high predictive power of the code make it useful for applications in nuclear technology and complement the use of purely empirical models.

### 15.7.2 Temperature-dependent Brosa model

The description of fission fragment and product yields follows the procedure outlined in Ref. [131]. The Hauser-Feshbach formalism gives a fission cross section per excitation energy bin for each fissioning system. The fission fragment masses and charges are, subsequently, determined per given excitation energy bin  $E_x$ , in a fissioning system FS, characterised by  $(Z_{FS}, A_{FS}, E_x)$ , for which the fission cross section exceeds some minimum value. The total fragment mass distribution is given by a sum over all contributing bins weighted with the corresponding fission cross sections:

$$\sigma(A_{FF}) = \sum_{Z_{FS}, A_{FS}, E_x} \sigma_f(Z_{FS}, A_{FS}, E_x) Y(A_{FF}; Z_{FS}, A_{FS}, E_x), \quad (15.19)$$

where  $Y(A_{FF}; Z_{FS}, A_{FS}, E_x)$  is the relative yield of a fission fragment with mass  $A_{FF}$  originating from a fissioning system. Combining this expression with the result of a fission fragment charge distribution calculation yields the final production cross section of a fission fragment with mass

$A_{FF}$  and charge  $Z_{FF}$ :

$$\sigma_{prod}(Z_{FF}, A_{FF}) = \sum_{Z_{FS}, A_{FS}, E_x} \sigma_f(Z_{FS}, A_{FS}, E_x) Y(A_{FF}; Z_{FS}, A_{FS}, E_x) \times Y(Z_{FF}; A_{FF}, Z_{FS}, A_{FS}, E_x) \quad (15.20)$$

$Y(Z_{FF}; A_{FF}, Z_{FS}, A_{FS}, E_x)$  is the relative yield of a fission fragment with charge  $Z_{FF}$  given its mass  $A_{FF}$  and the fissioning system characterised by  $(Z_{FS}, A_{FS}, E_x)$ .

In general, an excitation energy distribution is connected to these fission *fragment* production cross sections. In theory, this could be used to deduce the fission *product* yields through a full evaporation calculation of the fission fragments. This is not yet possible in TALYS. Instead a rather crude approximation, outlined later in this section, is adopted to estimate the number of post-scission neutrons emitted from each fragment. This procedure leads to relative yields for the fission product masses  $Y(A_{FP}; Z_{FS}, A_{FS}, E_x)$  and charges  $Y(Z_{FP}; A_{FP}, Z_{FS}, A_{FS}, E_x)$  and, hence, to expressions similar to Eq. (15.19) and Eq. (15.20) for the final fission products.

### 15.7.3 Fission fragment mass distribution

The fission fragment mass distribution is determined with a revised version of the multi-modal random neck-rupture model (MM-RNRM). The original model has been developed by Brosa to calculate properties of fission fragments at zero temperature [132]. However, fission calculations within TALYS require fragment mass distributions up to higher temperatures.

In the recent version of the model [131] the temperature is added to the calculation of the potential energy landscape of the nucleus. A search for the fission channels in deformation space yields the superlong (SL), standard I (ST I), and standard II (ST II) fission barriers and prescission shapes as a function of temperature. In this manner, the incorporated melting of shell effects naturally gives rise to the vanishing of the asymmetric fission modes ST I and ST II with increasing excitation energies. The obtained temperature-dependent fission barrier and prescission shape parameters serve as input for the fragment mass distribution computations in TALYS.

Each mass distribution is a sum over contributions of the three dominant fission modes FM:

$$Y(A_{FF}; Z_{FS}, A_{FS}, E_x) = \sum_{FM=SL, STI, STII} W_{FM}(Z_{FS}, A_{FS}, E_x) Y_{FM}(A_{FF}; Z_{FS}, A_{FS}, E_x), \quad (15.21)$$

where  $W_{FM}(Z_{FS}, A_{FS}, E_x)$  denotes the weight of a fission mode, and  $Y_{FM}(A_{FF}; Z_{FS}, A_{FS}, E_x)$  is the corresponding mass distribution.  $Y(A_{FF}; Z_{FS}, A_{FS}, E_x)$  can then be substituted in Eq. (15.19) to determine the full fission fragment mass distribution for the reaction under consideration. An analogous expression can be written down for  $Y(A_{FP}; Z_{FS}, A_{FS}, E_x)$ .

Each mass distribution calculation is started by determining the relative contributions of the different fission modes. These are evaluated with the Hill-Wheeler penetrability through inverted parabolic barriers using temperature-dependent barrier parameters and, consequently (the reader is referred to [131] for a detailed explanation), ground-state level densities:

$$T_{f,FM}(Z_{FS}, A_{FS}, E_x) = \int_0^\infty d\varepsilon \rho_{gs}(Z_{FS}, A_{FS}, \varepsilon) \frac{1}{1 + \exp\left[\frac{2\pi(B_{f,FM}(Z_{FS}, A_{FS}, T(\varepsilon)) + \varepsilon - E_x)}{\hbar\omega_{f,FM}(Z_{FS}, A_{FS}, T(\varepsilon))}\right]}. \quad (15.22)$$

All actinides encounter a double-humped barrier on their way to fission. The effective transmission coefficient can be expressed in terms of the transmission coefficient through the first and second barrier denoted by  $T_f^A$  and  $T_f^B$  respectively by Eq. (15.9). Since, however, the theoretical inner

barrier is much lower than the outer barrier, the relative contribution  $W_{FM}(Z_{FS}, A_{FS}, E_x)$  of the three fission modes may simply be determined by an equation of the following form:

$$W_{SL}(Z_{FS}, A_{FS}, E_x) = \frac{T_{f,SL}^B}{T_{f,SL}^B + T_{f,STI}^B + T_{f,STII}^B}. \quad (15.23)$$

Equivalent formulas hold for  $W_{STI}(Z_{FS}, A_{FS}, E_x)$  and  $W_{STII}(Z_{FS}, A_{FS}, E_x)$ .

For subactinides it is not possible to calculate the competition between symmetric and asymmetric fission modes, since the computed fission channels exhibit rather broad and strangely shaped outer barriers, which makes a parabola fit to these barriers impossible. Hence, the Hill-Wheeler approach cannot be applied. Fortunately, the SL barriers are much lower than the ST barriers. Therefore, in all these calculations the asymmetric fission modes are simply discarded and the dominant symmetric SL mode is solely taken into account for subactinides.

The RNRM is employed to calculate the mass distribution per fission mode. An extensive description of the RNRM may be found in [132]. Here it is merely attempted to communicate its main ideas. In this model, the fission process is regarded as a series of instabilities. After the passage over the barriers, a neck starts to form. If this neck becomes flat its rupture may happen anywhere, which means that the point of future constriction can shift over the neck. This motion of the dent is called the shift instability. In the instant that the Rayleigh instability starts to deepen the dent, the position of the asymmetry is frozen and rupture is taking place. The RNRM translates the effect of both mechanisms into measurable quantities.

In order for the shift instability to do its work, a perfectly flat neck is required. Hence, the so-called flat-neck parameterisation is introduced: (see Fig. 15.2):

$$\rho(\zeta) = \begin{cases} (r_1^2 - \zeta^2)^{1/2} & -r_1 \leq \zeta \leq \zeta_1 \\ r + a^2 c \left( \cosh\left(\frac{\zeta - z + l - r_1}{a}\right) - 1 \right) & \zeta_1 \leq \zeta \leq \zeta_2 \\ (r_2^2 - (2l - r_1 - r_2 - \zeta)^2)^{1/2} & \zeta_2 \leq \zeta \leq 2l - r_1. \end{cases} \quad (15.24)$$

The radius of the nucleus is given by  $\rho$  as a function of a parameter  $\zeta$  in terms of several parameters: the semilength  $l$ , the neck radius  $r$ , the position  $z$  of the dent, the curvature  $c$ , the extension of the neck  $a$ , the radii of the spherical heads  $r_1$  and  $r_2$ , and the transitional points  $\zeta_1$  and  $\zeta_2$ . By requiring continuity and differentiability of the shape, volume conservation and a minimal value of  $c$  for a really flat neck, only  $(l, r, z)$  remain as independent parameters. Subsequently, the neck radius is eliminated by the Rayleigh criterion, which relates the total length  $2l$  of the prescission shape to the neck radius  $r$  by  $2l = 11r$ . The value of  $z$  can be transformed into the heavy fragment mass  $A_h$  by:

$$A_h = \frac{3A}{4r_{cn}^3} \int_{-l}^z \rho^2(\zeta) d\zeta, \quad (15.25)$$

where  $r_{cn}$  is the compound nucleus radius. The actual values of  $A_h$  and  $l$  originate from the channel searches and are called the prescission shape parameters. They have been stored in the structure database and are input to the RNRM calculations.

One last ingredient is missing for the computation of the mass distribution, namely the surface tension:

$$\gamma_0 = 0.9517 \left( 1 - 1.7828 \left( \frac{N-Z}{A} \right)^2 \right) \text{MeV fm}^{-2}. \quad (15.26)$$

This is taken from the LDM by Myers and Swiatecki [27].

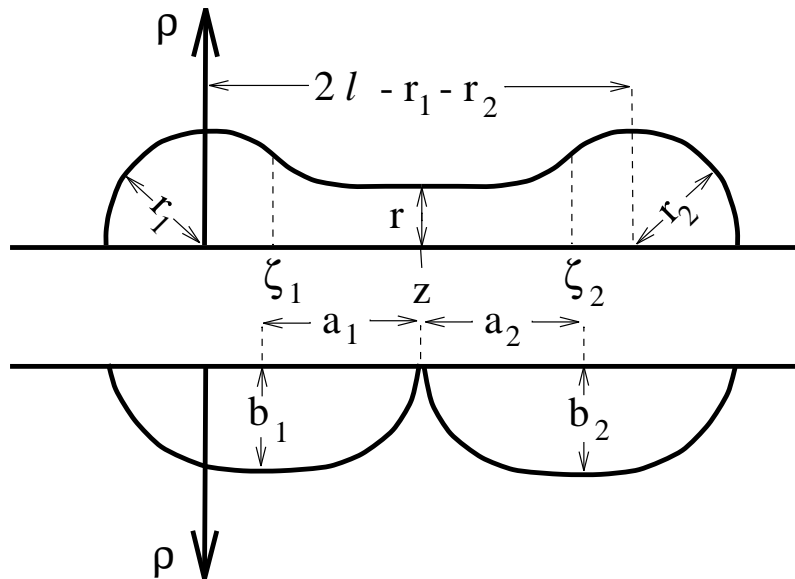


Figure 15.2: The upper part illustrates the flat-neck representation. The lower part contains the embedded spheroids parameterisation.

Fluctuations amplified by the shift instability alter the shape slightly and enable the rupture of the nucleus to take place at another point than the most probable point  $z$ . In order to determine the fission-fragment mass distribution, the probability of cutting the neck at an arbitrary position  $z_r$  has to be calculated. This probability is given by the change in potential energy from  $z_r$  to  $z$ :  $E(z_r) - E(z)$ . This is replaced by the energy to cut the nucleus at the two positions:  $E_{cut}(z_r) - E_{cut}(z)$ , with  $E_{cut}(z_r) = 2\pi\gamma_0\rho^2(z_r)$ . The rupture probability is now proportional to the Boltzmann factor:

$$y(A_{FF}) \propto \exp\left(\frac{-2\pi\gamma_0(\rho^2(z_r) - \rho^2(z))}{T}\right). \quad (15.27)$$

The fragment mass number  $A_{FF}$  can be computed according to the analogue of Eq. (15.25):

$$A_{FF}(z_r) = \frac{3A}{4r_{cn}^3} \int_{-l}^{z_r} \rho^2(\zeta) d\zeta. \quad (15.28)$$

The theoretical yield is finally determined with the following relation in which  $Y(A_{FF})$  stands for the normalized fission fragment mass yield:

$$Y_{FM}(A_{FF}; Z_{FS}, A_{FS}, E_x) = y(A_{FF}) + y(A - A_{FF}). \quad (15.29)$$

In Eq. (15.27) the temperature of the scissioning nucleus must be provided. All calculations of the PES and the crossing of the fission barriers have been isothermal. However, for the RNRM the loss and gain of excitation energy in crossing the barrier is taken into account into a new excitation energy and temperature at scission:

$$E_x^{scission} = E_x^{groundstate} + F_{def,scission}. \quad (15.30)$$

The new excitation energy has two components: the original excitation energy in the ground state  $E_x^{groundstate}$  and the deformation energy at scission  $F_{def,scission}$ .  $F_{def,scission}$  is positive for actinides and becomes negative in the subactinide region. The new excitation energy is related to a new temperature  $T_{scission}$ . However, a new prescission temperature corresponds to a different prescission shape with a somewhat different value for  $F_{def,scission}$ . Therefore, the temperature  $T_{scission}$  has to be determined in a self-consistent manner together with the final prescission shape. If a prescission shape has a high temperature or a very long neck, the mass distribution will be broad. Low temperatures and short necks result in a narrow mass distribution.

#### 15.7.4 Post-scission neutron multiplicities

The mass distribution calculated above belongs to the primary fission fragments. Most fragments, however, are highly excited directly after their creation. They take their share of total excitation energy available at scission (15.30). Moreover, they are strongly deformed, which manifests itself in an extra amount of excitation energy set free when this deformation relaxes towards the ground-state deformation of the fragments by the strong surface tension. The superfluous excitation energy is released during the process of post-scission neutron and gamma emission. The neutron emission is responsible for a shift of the pre-neutron emission mass distribution to somewhat smaller masses.

The total excitation energy in a newly created fragment with mass  $A_{FF}$  results from:

$$E_x^{fragment}(A_{FF}) = E_{def,fragment}(A_{FF}) + \frac{A_{FF}}{A} E_x^{scission}. \quad (15.31)$$

$E_{def,fragment}(A_{FF})$  denotes the deformation energy of the fragment, and the second term contains the portion of the thermal energy at scission of the whole fissioning system picked up by the fragment. The assumption is that the fragment receives a share proportional to its mass.

For the calculation of  $E_{def,fragment}(A_{FF})$  another shape parameterisation is employed: the embedded spheroids (see Fig. 15.2). The newborn fragments are modeled as two contacting spheroids with major axes  $a_1$  and  $a_2$ , which are linked to  $2l$  and  $z_r$  by:

$$a_1 = \frac{1}{2}(r_1 + z_r), \quad a_2 = l - \frac{1}{2}(r_1 + z_r). \quad (15.32)$$

The minor axes  $b_1$  and  $b_2$  follow from volume conservation:

$$b_1^2 = \frac{3}{4a_1} \int_{-r_1}^{z_r} \rho^2 d\zeta, \quad b_2^2 = \frac{3}{4a_2} \int_{z_r}^{2l-r_1} \rho^2 d\zeta. \quad (15.33)$$



The energy difference of the spheroidally deformed and the spherical fragment  $E_{def,fragment}(A_{FF})$  is given by:

$$E_{def,fragment}(A_{FF}, \varepsilon) = E_{surf}^{sph}(A_{FF}) \left( \frac{\arcsin(\varepsilon) + \varepsilon(1 - \varepsilon^2)^{1/2}}{2\varepsilon(1 - \varepsilon^2)^{1/6}} - 1 \right) + E_{Coul}^{sph}(A_{FF}) \left( \frac{(1 - \varepsilon^2)^{1/3}}{2\varepsilon} \ln \left( \frac{1 + \varepsilon}{1 - \varepsilon} \right) - 1 \right). \quad (15.34)$$

The eccentricity is defined as:

$$\varepsilon_i = \left( 1 - \left( \frac{b_i}{a_i} \right)^2 \right)^{1/2}, \quad (15.35)$$

and  $E_{surf}^{sph}(A_{FF})$  and  $E_{Coul}^{sph}(A_{FF})$  represent the LDM surface and the Coulomb energy of a spherical nucleus obtained from Ref. [133].

The neutron multiplicity  $\nu_{FM}(A_{FF})$  for a fragment with mass  $A_{FF}$  is now derived by finding the root of the following relation:

$$E_x^{fragment}(A_{FF}) = \sum_{n=1}^{\nu_{FM}(A_{FF})} (S_n + \eta_n) + E_\gamma. \quad (15.36)$$

The separation energy  $S_n$  is calculated from the mass formula [133]. The average kinetic energy of the neutrons is taken to be  $\frac{3}{2}$  times the fragment temperature, and the energy carried off by  $\gamma$ -rays  $E_\gamma$  is approximately half the separation energy of the first non-evaporated neutron.

The final fission product mass yield per fission mode is given by:

$$Y_{FM}(A_{FP}; Z_{FS}, A_{FS}, E_x) = Y_{FM}(A_{FF} - \nu_{FM}(A_{FF})) + Y_{FM}(A - A_{FF} - \nu_{FM}(A - A_{FF})). \quad (15.37)$$

### 15.7.5 Fission fragment charge distribution

Unfortunately, the MM-RNRM only yields information on the mass yields of the fission fragments. Predictions of charge distributions are performed in TALYS within a scission-point-model-like approach (Wilkins *et al.* [134]). Corresponding to each fission fragment mass  $A_{FF}$  a charge distribution is computed using the fact that the probability of producing a fragment with a charge  $Z_{FF}$  is given by the total potential energy of the two fragment system inside a Boltzmann factor:

$$Y_{FM}(Z_{FF}; A_{FF}, Z_{FS}, A_{FS}, E_x) = \frac{\exp\left(\frac{-E(Z_{FF}, A_{FF}, Z, A)}{T}\right)}{\sum_{Z_{FF_i}} \exp\left(\frac{-E(Z_{FF_i}, A_{FF}, Z, A)}{T}\right)} \quad (15.38)$$

In the original scission-point model this potential energy is integrated over all deformation space. Within the MM-RNRM, however, fission channel calculations already define the deformation at the scission point. Furthermore, a strong coupling between the collective and single particle degrees of freedom is assumed near the scission point. This means that the nucleus is characterised by a single temperature  $T$ .

The potential energy for the creation of one fragment with  $(Z_{FF}, A_{FF})$  and a second fragment with  $(Z - Z_{FF}, A - A_{FF})$  consists of a sum over the binding energy of the deformed fragments by the droplet model without shell corrections and their mutual Coulomb repulsion energy as well as the nuclear proximity repulsion energy:

$$E(Z_{FF}, A_{FF}, Z, A) = B(Z_{FF}, A_{FF}) + B(Z - Z_{FF}, A - A_{FF}) + E_{Coul} + V_{prox}. \quad (15.39)$$

The single constituents of this formula are defined by the following relations:

$$B(Z_{FF}, A_{FF}) = -a_1 \left[ 1 - \kappa \left( \frac{A_{FF} - 2Z_{FF}}{A_{FF}} \right)^2 \right] A_{FF} \quad (15.40)$$

$$+ a_2 \left[ 1 - \kappa \left( \frac{A_{FF} - 2Z_{FF}}{A_{FF}} \right)^2 \right] A_{FF}^{\frac{2}{3}} f(shape) \\ + \frac{3}{5} \frac{e^2}{r_0} \frac{Z_{FF}^2}{A_{FF}^{\frac{1}{3}}} g(shape) - \frac{\pi^2}{2} \frac{e^2}{r_0} \left( \frac{d}{r_0} \right)^2 \frac{Z_{FF}^2}{A_{FF}} \quad (15.41)$$

$$E_{Coul} = e^2 Z_{FF} (Z - Z_{FF}) S_{form} / (a_1 + a_2) \quad (15.42)$$

$$V_{prox} = -1.7817 \frac{4\pi\gamma_0 b_1^2 b_2^2}{a_1 b_2^2 + a_2 b_1^2} \quad (15.43)$$

The parameters in the binding energy formula are taken from Ref. [133]. The function  $f(shape)$  is the form factor for the Coulomb term whereas  $g(shape)$  denotes the form factor for the surface energy.  $S_{form}$  is the form factor for the Coulomb interaction energy between two spheroids.

The fission product charge distribution is obtained from the calculated fission fragment charge distribution by shifting the corresponding fragment masses  $A_{FF}$  to the fission product masses  $A_{FP}$  with the aid of the post-scission neutron multiplicity  $\nu_{FM}(A_{FF})$ :

$$Y_{FM}(Z_{FP}; A_{FP}, Z_{FS}, A_{FS}, E_x) = Y_{FM}(Z_{FF}; A_{FF} - \nu_{FM}(A_{FF}), Z_{FS}, A_{FS}, E_x). \quad (15.44)$$

Since proton evaporation of the fission fragments is neglected, the charge distribution connected with  $A_{FF}$  becomes simply linked to the fission product mass  $A_{FP}$ .

## 15.8 Implementation, options and sample cases

**Implementation 15.8.1** The fission parameters are stored in the *talys/structure/fission/* directory. There are various subdirectories: *barrier/*, *states/*, *gef*, *brosa/*, *mamdouh/*, and *hfbpath/*.

First, the experimental fission parameter set can be found in directory *barrier/*. The directory *states/* includes headband transition states for even-even, even-odd, odd-odd, and odd-even nuclides. The parameters are the result of an extensive fit to many experimental fission cross sections, compiled by V. Maslov for the RIPL library. As an example we show the available parameters for the uranium isotopes which are present in the file *barrier/U.bar*. We have stored Z, A, a parameter specifying the symmetry of the inner barrier, height of the inner barrier, curvature of the inner barrier, a parameter specifying the symmetry of the outer barrier, height of the outer barrier, curvature of the outer barrier, the pairing correlation function at the barrier (which is not used in the calculation) and the nuclear symbol with the format (2i4,a5,2f8.3,a5,2f8.3,f9.4,15x,i4,a2).

92	231	S	4.40	0.70	MA	5.50	0.50	0.869	231U
92	232	S	4.90	0.90	MA	5.40	0.60	0.848	232U
92	233	S	4.35	0.80	MA	5.55	0.50	0.946	233U
92	234	S	4.80	0.90	MA	5.50	0.60	0.889	234U
92	235	S	5.25	0.70	MA	6.00	0.50	0.803	235U
92	236	S	5.00	0.90	MA	5.67	0.60	0.833	236U
92	237	GA	6.40	0.70	MA	6.15	0.50	0.809	237U
92	238	GA	6.30	1.00	MA	5.50	0.60	0.818	238U
92	239	GA	6.45	0.70	MA	6.00	0.50	0.816	239U

In *states*, we have stored files with default head band and class II-states for even-even, even-odd, odd-odd, and odd-even nuclides. An example of a file containing the head band transition states is given below for *states/hbstates.ee*. This file is used for even-even nuclides. We loop over the two barriers. On the first line one finds the barrier number, the number of head band states and the energy at which the continuum starts with the format(2i4,f8.3). Next we loop over the transition states, in which we read the level number, level energy in MeV, spin and parity with the format (i4,f11.6,f6.1,i5).

```

1  8  0.800
1  0.000000  0.0  1
2  0.500000  2.0  1
3  0.400000  0.0  -1
4  0.400000  1.0  -1
5  0.500000  2.0  1
6  0.400000  2.0  -1
7  0.800000  0.0  1
8  0.800000  0.0  1
2  4  0.500
1  0.000000  0.0  1
2  0.500000  2.0  1
3  0.200000  0.0  -1
4  0.500000  1.0  -1

```

We also include the possibility to incorporate the effect of class II states in the calculation. For this purpose we take four parameter sets with class II states for even-even, even-odd, odd-odd, and odd-even nuclides. As an example we show the file *states/class2states.ee*. This file contains the well number, and the number of class II states. Subsequent lines contain the level number, the level energy in MeV, spin and parity with the format (i4,f8.3,f9.1,i5).

```

1  10
1  2.700      0.0  1
2  3.400      0.0  -1
3  4.100      1.0  -1
4  4.800      2.0  -1
5  5.000      1.0  1
6  5.200      0.0  1
7  5.400      0.0  -1
8  5.500      1.0  -1
9  5.600      2.0  -1
10 5.700      1.0  1

```

The directory *brosa/* contains three subdirectories: *barrier/*, *groundstate/*, and *prescission/*. In *brosa/barrier/* temperature-dependent fission barrier parameters per fission mode can be found. They are the results of calculations performed within the Brosa model. The extension in the filename reveals the fission mode: *sl* for superlong, *st* for standard I, and *st2* for standard II. As an example, below are the superlong parameters for several U isotopes taken from *brosa/barrier/U.sl*. Each line gives Z, A, temperature T in MeV,  $B_F$  in MeV,  $\hbar\omega$  in MeV, and the barrier position in terms of the distance between the fragment centers d in fm. The format is (2i4,4f15.5).

92 240	0.00000	11.40470	3.92020	10.37450
92 240	0.30000	11.23440	4.18630	10.34980
92 240	0.60000	11.64780	4.25390	10.44170
92 240	0.90000	9.58680	2.58900	10.35970
92 240	1.20000	7.88030	2.36520	10.53400
92 240	1.60000	5.69890	1.58010	10.43420
92 240	2.00000	4.17750	1.07010	10.28590
92 240	2.50000	2.76320	0.47300	9.69080
92 240	3.00000	1.84230	0.25520	8.57080
92 238	0.00000	10.88450	4.67530	10.27930
92 238	0.30000	10.97680	4.87110	10.27090
92 238	0.60000	10.59680	4.44100	10.29590
92 238	0.90000	9.31080	3.69600	10.37940
92 238	1.20000	7.55620	2.32260	10.40990
92 238	1.60000	5.46180	1.51320	10.33670
92 238	2.00000	3.98050	0.82730	10.15860
92 238	2.50000	2.74050	0.43440	9.43930
92 238	3.00000	1.87030	0.34600	8.62610

The ground state energies as a function of temperature are stored in *brosa/barrier/groundstate/*. Each line has the same format: Z, A, T in MeV, and  $E_{groundstate}$  in MeV (2i4,2f15.5). An example is included for U isotopes, see *brosa/barrier/groundstate/U.fis*.

92 240	0.00000	-1811.82104
92 240	0.30000	-1813.38599
92 240	0.60000	-1818.06006
92 240	0.90000	-1825.59497
92 240	1.20000	-1836.26599
92 240	1.60000	-1855.76294
92 240	2.00000	-1881.23901
92 240	2.50000	-1921.08105
92 240	3.00000	-1969.60205
92 238	0.00000	-1800.57996
92 238	0.30000	-1802.16504
92 238	0.60000	-1806.85205
92 238	0.90000	-1814.34302
92 238	1.20000	-1825.23596
92 238	1.60000	-1844.68799
92 238	2.00000	-1870.17004
92 238	2.50000	-1910.02905
92 238	3.00000	-1958.52405

The third directory, *brosa/barrier/precission/*, contains parameters that fix the precission shape of the nucleus in each fission mode. They mark the end of the fission path found in the same fission channel calculations that resulted in the Brosa barrier parameters. Again each line has the same format: Z, A, T in MeV,  $A_h$  the heavy fragment mass in amu,  $l$  the nucleus half length in fm, and  $E_{precission}$  in MeV (2i4,4f15.5). An example is included for precission shape parameters in the superlong mode of U isotopes, see *brosa/barrier/precission/U.sl*,

92 240	0.00000	120.45900	21.29700	-1840.84900
92 240	0.30000	120.83730	21.48670	-1826.35022

92 240	0.60000	120.83670	20.74300	-1831.38477
92 240	0.90000	120.30430	20.21950	-1840.12256
92 240	1.20000	120.65520	20.13060	-1854.43555
92 240	1.60000	120.83060	19.56120	-1874.16174
92 240	2.00000	120.74340	19.33610	-1902.75671
92 240	2.50000	120.62800	18.38600	-1945.59497
92 240	3.00000	120.78300	18.45000	-1998.50598
92 238	0.00000	120.00730	21.40630	-1815.46497
92 238	0.30000	119.64770	21.36200	-1816.33203
92 238	0.60000	119.50730	20.50950	-1821.51123
92 238	0.90000	119.27930	20.50120	-1830.13428
92 238	1.20000	119.47550	19.91770	-1842.70630
92 238	1.60000	123.00390	19.23990	-1864.13489
92 238	2.00000	120.49580	19.24370	-1892.63281
92 238	2.50000	119.62200	18.37100	-1935.14502
92 238	3.00000	119.37600	18.38000	-1988.42395

In directory *gef*, masses and shell correction energies for the GEF model are stored.

The Mamdouh fission parameter set can be found in directory *mamdouh/*. This parameter set has been derived from Extended Thomas-Fermi plus Strutinsky Integral calculations and comprise double-humped fission barrier heights and curvatures for numerous isotopes. As an example we show the available parameters for the various U isotopes which are present in the file *mamdouh/U.bar*. We have stored Z, A, height of the inner barrier, height of the outer barrier in MeV and the nuclear symbol with the format (i3,i4,2(24x,f8.2),5x,i3,a2).

92 230	1.24	0.00	0.00	3.80	1.77	-0.02	0.35	3.90	230U
92 231	1.24	0.01	0.00	4.10	1.83	0.05	0.50	4.30	231U
92 232	1.25	0.02	0.00	4.20	1.83	0.05	0.53	4.20	232U
92 233	1.28	0.05	0.00	4.70	1.84	0.06	0.53	4.40	233U
92 234	1.28	0.03	0.00	4.80	1.83	0.06	0.53	4.40	234U
92 235	1.28	0.04	0.00	5.40	1.63	0.01	0.53	4.10	235U
92 236	1.29	0.04	0.00	5.20	1.64	0.01	0.53	4.00	236U
92 237	1.28	0.04	0.00	5.70	1.63	0.01	0.53	4.30	237U
92 238	1.29	0.03	0.00	5.70	1.91	0.07	0.47	4.90	238U
92 239	1.29	0.03	0.00	6.10	1.81	-0.07	0.35	5.50	239U
92 240	1.29	0.03	0.00	6.00	1.90	-0.04	0.53	6.30	240U
92 241	1.30	0.04	0.00	6.30	1.91	-0.03	0.55	5.70	241U
92 242	1.30	0.04	0.00	5.90	1.90	-0.04	0.53	6.00	242U

If we use the WKB approximation to calculate fission transmission coefficients, **fismodel 5**, we need potential energy curves. They can be found in the directory *hfbpath/*. As an example, we show the data for  $^{235}\text{U}$  as present in the file *hfbpath/U.fis*. We have stored Z, A, number of fission width values, total energy (not used) with the format (3i4,f12.3). Next, we give the values for the fission width and height with the format (f10.3,20x,f10.3),

92 235 100	-1781.682			
0.269	0.000	0.152	0.000	
0.298	0.000	0.180	0.332	
0.326	0.000	0.189	1.047	
0.355	0.000	0.171	1.987	
0.383	0.000	0.216	2.846	

0.412	0.000	0.199	3.921
0.440	0.000	0.194	4.695
0.470	0.000	0.158	5.002
0.501	0.000	0.136	5.391
0.531	0.000	0.156	5.390
0.563	0.000	0.173	5.186

### Sample case 15.2 Fission yields for $^{238}\text{U}$

In this sample case we compute fission fragment/product mass/isotope yields. The fission fragment mass yield curve is determined for neutrons on  $^{238}\text{U}$  at incident energies of 1.6 and 5.5 MeV (these incident energies are given in the file *energies*). The following input file is used:

```
#
# n-U238-fy
#
# General
#
projectile n
element    U
mass      238
energy    energies
partable  y
bins      40
channels  y
filechannels y
autorot   y
best      y
ecissave  y
massdis   y
ffevaporation y
maxlevelstar 30
```

The long list of adjustable parameters, not visible here but automatically taken from *talys/structure/best/U238/z092a238n.best* through the **best y** keyword, is mainly needed to give a decent (though not yet perfect) description of the total fission cross section. For completeness, we show the results for  $^{238}\text{U}$  we obtained for incident energies of 1, 1.6, 3.5, 5.5, 7.5 and 10 MeV. The TALYS results for the pre-neutron emission mass yields can be found in *yield001.600.fis* and *yield005.500.fis* and are given in the upper plot of Fig. 15.3. The two other plots show a comparison of the normalized yields with experimental data [135].

Since we have added the keyword *ffevaporation y* to the input, we have also calculated the fission product isotope yields. Fig. 15.4 contains the result for the production of the fission products  $^{115}\text{Cd}$  and  $^{140}\text{Ba}$ . The left plot shows the cumulative yield (obtained after adding the calculated independent yields of all beta-decay precursors). The normalized cumulative yields are compared to experimental data [136, 137] in the other two plots. ■

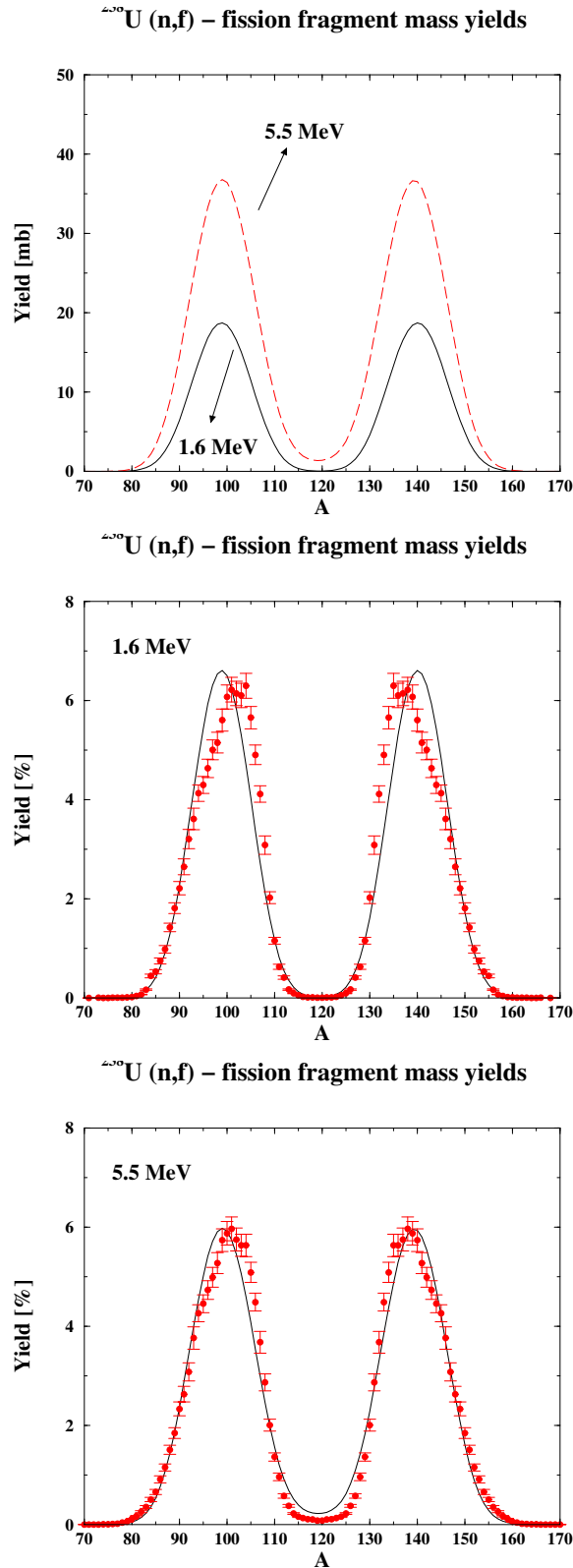


Figure 15.3: Fission fragment mass yield curves as function of the mass number  $A$ , produced by 1.6 MeV and 5.5 MeV neutrons on  $^{238}\text{U}$ . The upper curve shows the results as they are produced by TALYS and the other two plots contain the comparison with experimental data in terms of normalized yields [135].

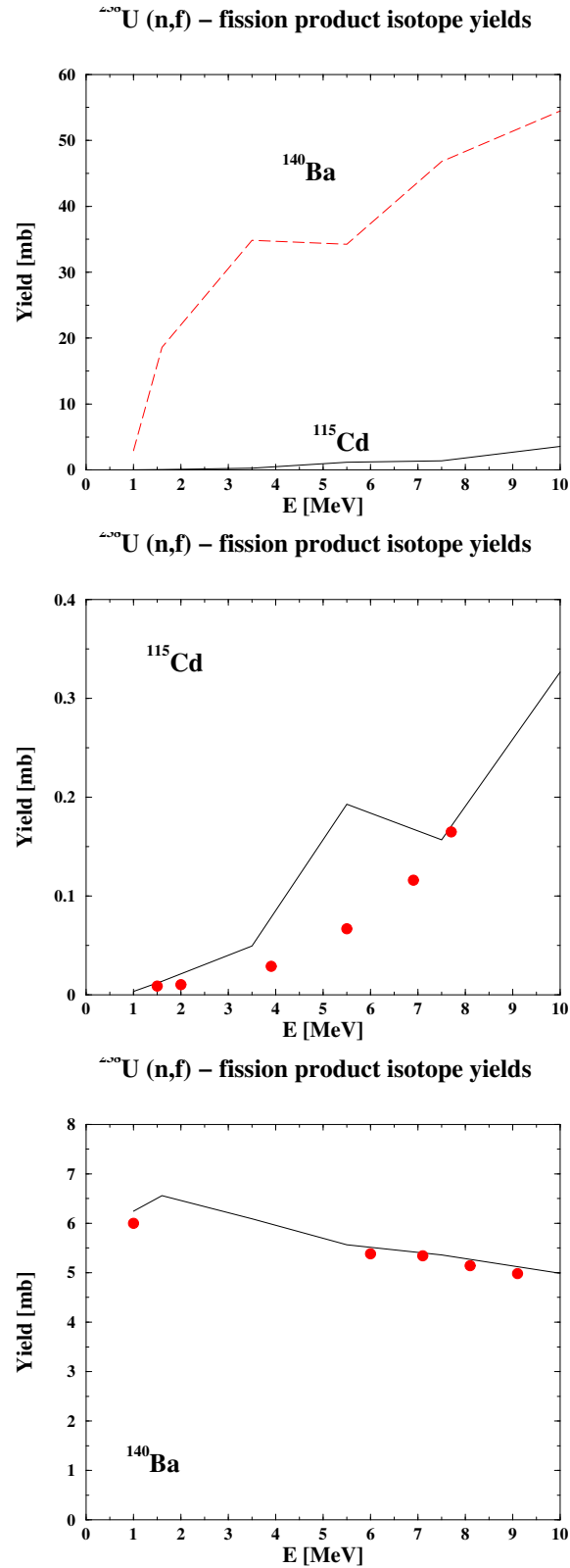


Figure 15.4: Fission product isotope yields produced by neutrons on  $^{238}\text{U}$  as function of the mass number  $A$ . The upper plot shows the results for  $^{115}\text{Cd}$  and  $^{140}\text{Ba}$  as they are produced by TALYS and the other two plots contain the comparison with experimental data in terms of normalized yields [136, 137].



**Sample case 15.3 Neutron multiplicities and fission yields for  $n + {}^{242}\text{Pu}$** 

TALYS contains an implementation of several versions of the GEF code by Karl-Heinz Schmidt and Beatriz Jurado. Part of GEF has been translated by Vasily Simutkin and Michail Onegin into a Fortran subroutine. In this sample case, the prompt neutron multiplicity as a function of mass,  $\bar{\nu}(A)$ , number of neutrons,  $P(\nu)$ , and the total average,  $\bar{\nu}$ , are calculated, as well as the pre- and post-neutron fission yields as a function of  $N$  and  $A$ . The following input file is used

```
#
# n-Pu242-fy
#
# General
#
projectile n
element    Pu
mass      242
energy    n0-20.grid
#
# Parameters and models
#
maxrot 2
fymodel 2
best y
#
# Output
#
channels y
filechannels y
ecissave y
inccalc y
eciscalc y
massdis y
# In a second run, you may use
# eciscalc n
# inccalc n
# for efficient variations for this input file
```

Note the comment given at the end of the input file. Note also that a hardwired energy grid is used through *n0-20.grid*. The prompt average neutron multiplicity  $\bar{\nu}$  is given in the output file *nubar.tot*. Fig. 15.5 shows a comparison with experimental data and some of the world nuclear data libraries. Fig. 15.6 presents the distribution of  $\bar{\nu}(A)$  as a function of fission product mass, for an incident energy of 1 MeV. This is given in output file *nuA001.000.fis*. Fig. 15.7 presents the distribution of  $\nu$  as a function of number of neutrons, for an incident energy of 1 MeV. This is given in output file *Pnu001.000.fis*. ■

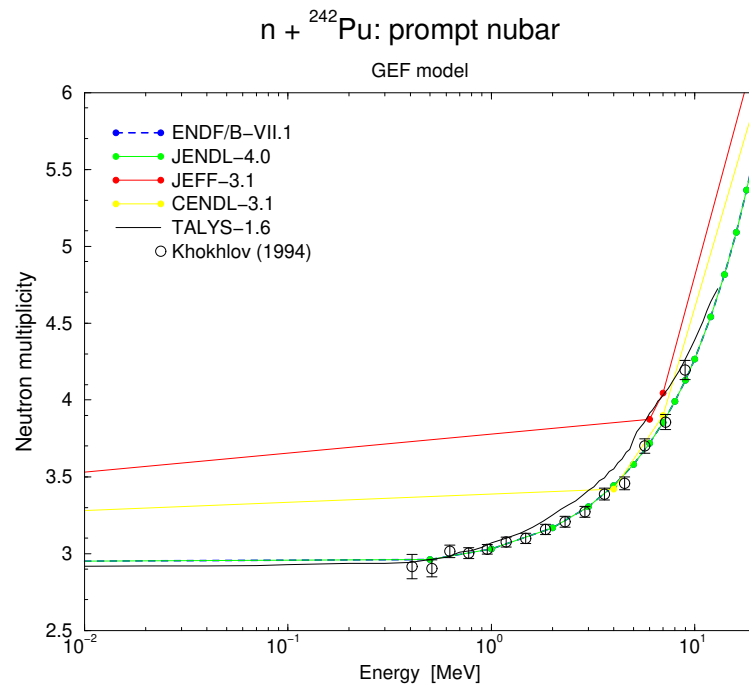


Figure 15.5: Average prompt neutron multiplicity  $\bar{\nu}$  for  $n + {}^{242}\text{Pu}$ , with the GEF model, compared with experimental data and nuclear data libraries.

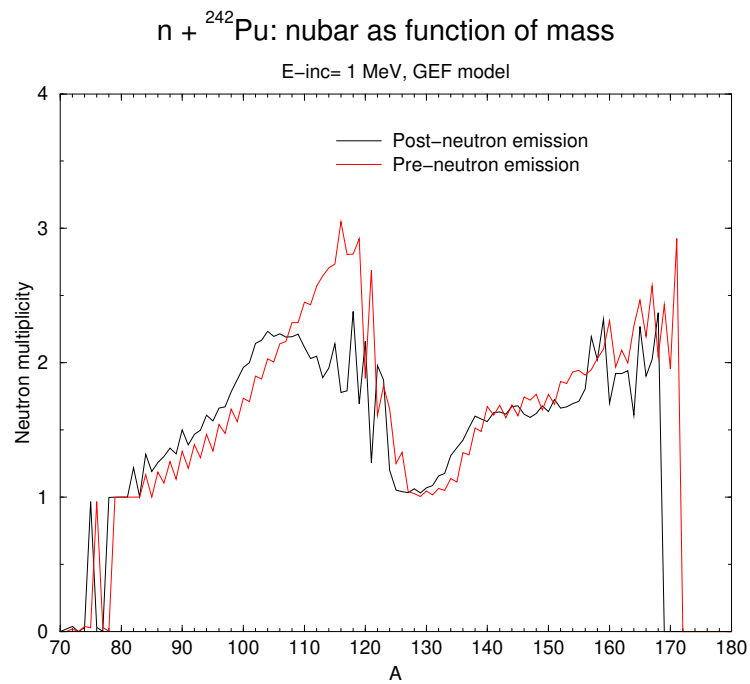


Figure 15.6: Average prompt neutron multiplicity  $\bar{\nu}(A)$ , as a function of fission product mass, for  $n + {}^{242}\text{Pu}$ , with the GEF model.

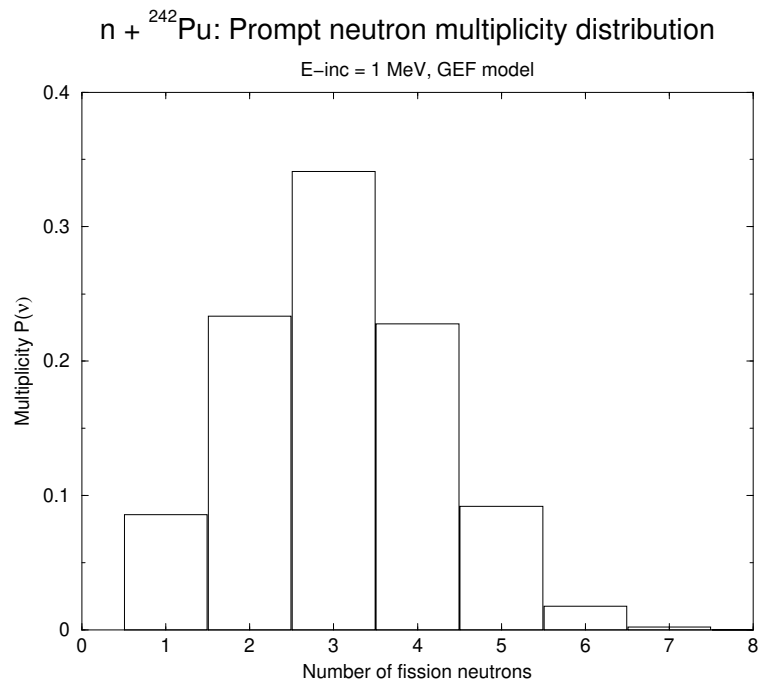


Figure 15.7: Prompt neutron multiplicity distribution  $P(\nu)$ , as a function of the number of neutrons, for  $n + {}^{242}\text{Pu}$ , with the GEF model.

#### Sample case 15.4 Fission yields and neutron observables $n + {}^{239}\text{Pu}$

Since 2020, TALYS has the option to start a calculation with a pre-defined fission fragment (FF) distribution and to explicitly calculate the neutron and gamma decay from that, leading to fission product yields (FPY) and various fission neutron observables. In these sample cases for thermal neutrons on  ${}^{239}\text{Pu}$ , 3 different FF models are compared with regards to the resulting FPY and neutron distributions. The prompt neutron multiplicity as a function of mass,  $\bar{\nu}(A)$ , number of neutrons,  $P(\nu)$ , the total average,  $\bar{\nu}$ , and the prompt fission neutron spectrum, are calculated, as well as the fission product yields as a function of  $Z$  and  $A$ . The following input file for the GEF model is used

```
projectile n
element Pu
mass 239
energy 2.53e-8
ejectiles g n
massdis y
fymodel 4
ffmodel 1
ldmodel 5
Rfiseps 1.e-9
outspectra y
bins 100
channels y
```

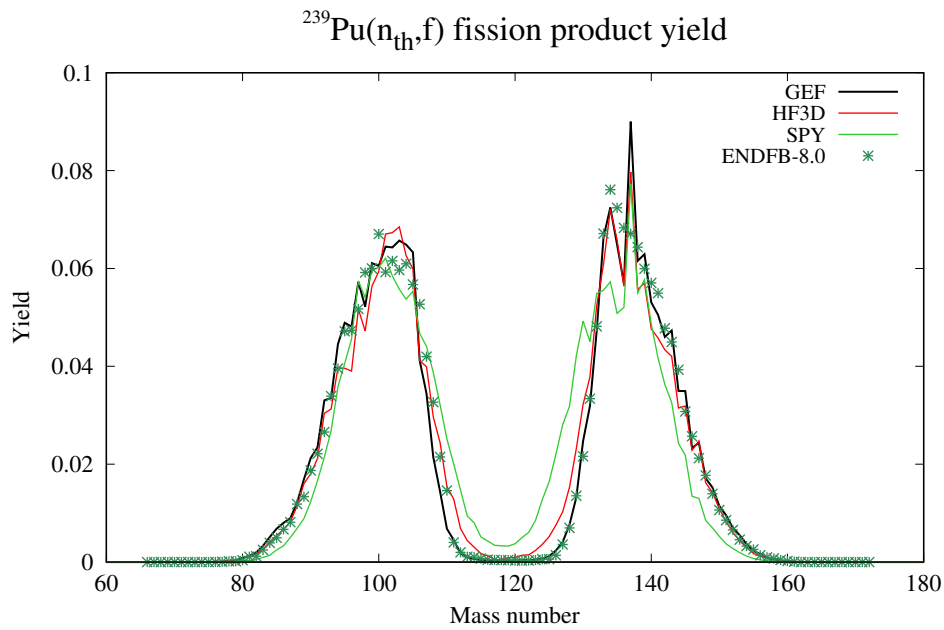


Figure 15.8: Fission product yields for  $n_{th} + ^{239}\text{Pu}$ , for 3 FF models.

maxchannel 8

and similarly for the HF3D model (**ffmodel 2**) and SPY model (**ffmodel 3**). The FY calculation can be somewhat time consuming so the sample cases are done only for thermal neutron energy. Not however that the usual incident energy file can also be used to obtain e.g.  $\bar{\nu}$  as a function of incident energy. In Fig. 15.8 the FPY results for the 3 models are plotted together with the FPY from ENDF/B-VIII.0. These FPY as a function of  $A$  are given in file *yieldA1.00E-06.fis*. Fig. 15.9 presents the distribution of  $\bar{\nu}(A)$  as a function of fission product mass. This is given in output file *nunA1.00E-06.fis*. Fig. 15.10 presents the distribution of  $\nu$  as a function of number of neutrons, for an incident energy of 1 MeV. This is given in output file *Pnun1.00E-06.fis*. The prompt fission neutron spectrum of Fig. 15.11 is given in file *pfns1.00E-06.fis*. In the addition to the latter case, in *n-Pu239-fy-pfns* we have also done a TALYS run without explicit FY evaporation, which means the phenomenologicla model of Iwamoto model for PFNS is automatically used. ■

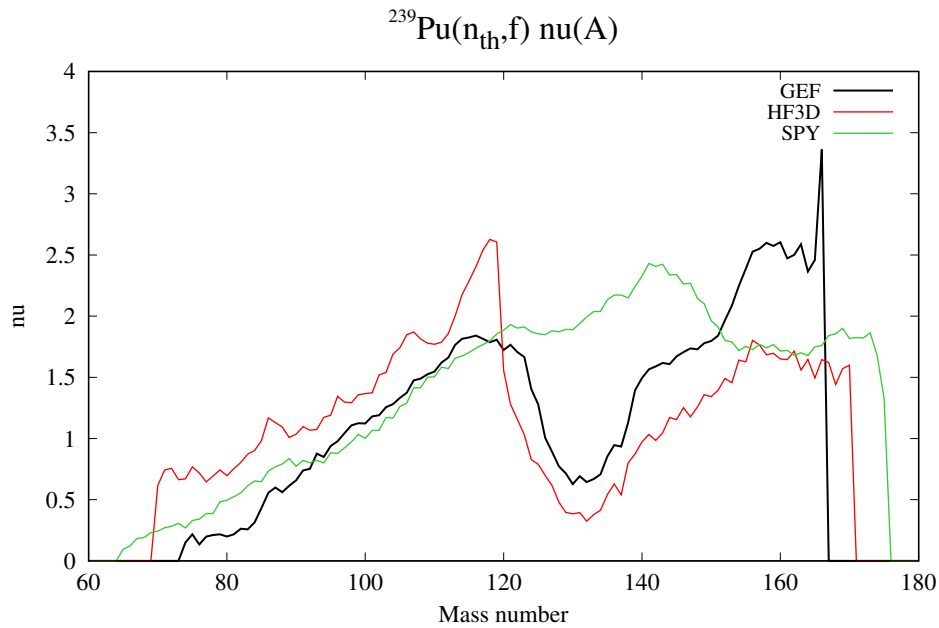


Figure 15.9: Prompt neutron multiplicity  $\bar{\nu}(A)$ , as a function of fission product mass, for  $n_{th} + ^{239}\text{Pu}$ , for 3 FF models.

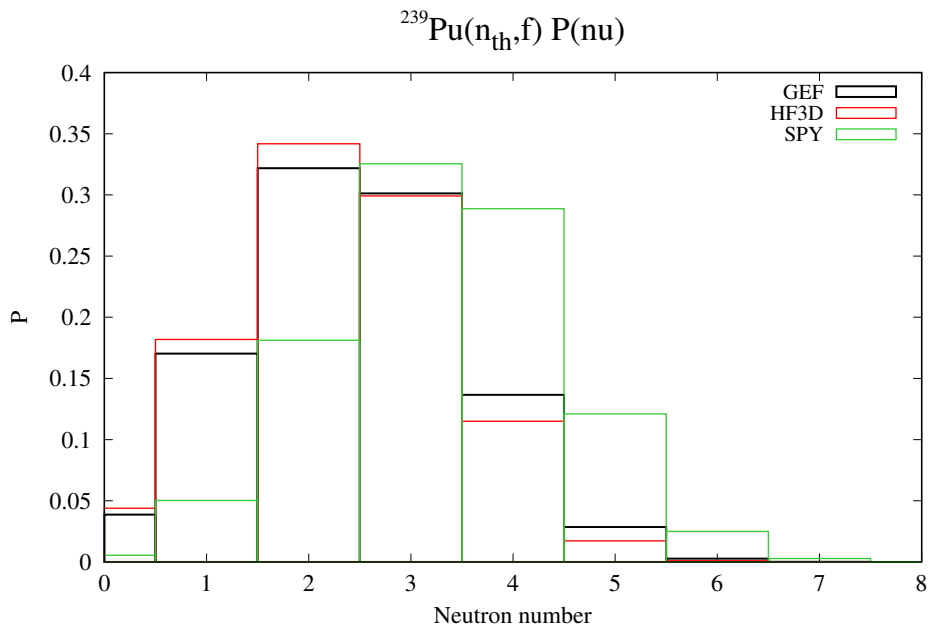


Figure 15.10: Prompt neutron multiplicity distribution  $P(\nu)$ , for  $n_{th} + ^{239}\text{Pu}$ , for 3 FF models.

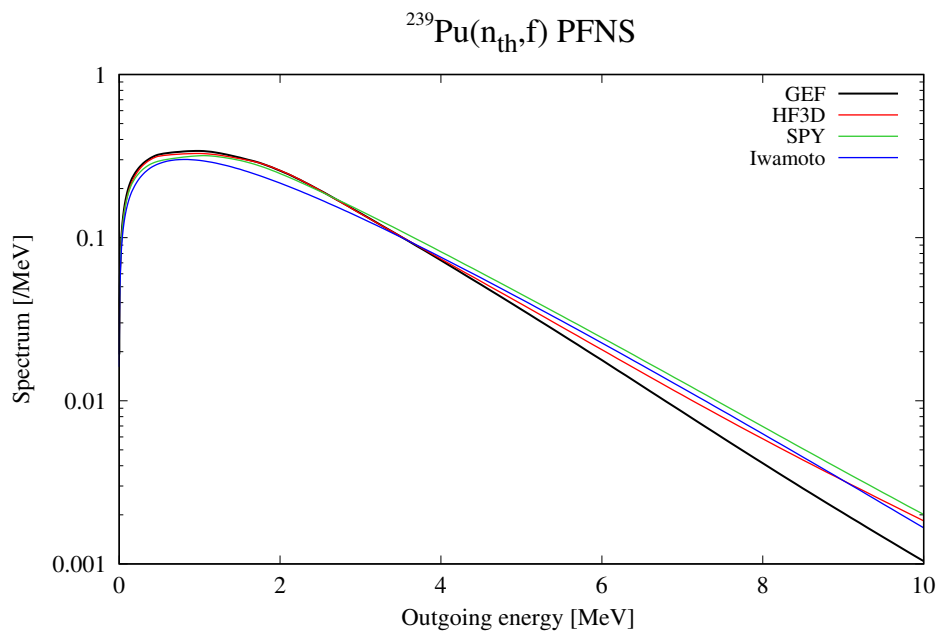


Figure 15.11: Prompt fission neutron spectrum, for  $n_{th} + ^{239}\text{Pu}$ , for 3 FF models.

## 16. Thermal reactions

The Introduction states that TALYS is meant for the analysis of data in the 1 keV - 200 MeV energy region. The lower energy of 1 keV should not be taken too literally. More accurate is: above the resolved resonance range. The start of the unresolved resonance range differs from nucleus to nucleus and is related to the average resonance spacing  $D_0$  or, equivalently, the level density at the binding energy. Generally, the starting energy region is higher for light nuclides than for heavy nuclides. Only beyond this energy, the optical and statistical models are expected to yield reasonable results, at least for the non-fluctuating cross sections. The lower energies are the domain of R-matrix theory, which describes the resonances. Nevertheless, it would be useful to have a first-order estimate of the non-threshold reactions, not only for the obvious neutron capture channel, but also for the exothermal  $(n, p)$ ,  $(n, \alpha)$  and fission channels. The fact that a nuclear model calculation in TALYS is only performed down to about 1 keV should not prevent us to give at least an estimate of the  $1/v$ -like behaviour of the excitation function down to  $10^{-5}$  eV (the lower energy limit in ENDF-6 files). In collaboration with J. Kopecky, we constructed a method that provides this.

### 16.1 Capture channel

First, we decide on the lower energy of validity of a TALYS nuclear model calculation  $E_L$ . Somewhat arbitrarily, we set as default  $E_L = D_0$  when we wish to construct evaluated data libraries, where  $D_0$  is taken from the nuclear model database or, if not present, derived from the level density.  $E_L$  can also be entered as an input keyword (**Elow**). Next, we determine the neutron capture cross section at the thermal energy  $E_{th} = 2.53 \cdot 10^{-8}$  MeV, either from the experimental database, see Chapter 5, or, if not present, from the systematical relation [138]

$$\sigma_{n,\gamma}(E_{th}) = 1.5 \times 10^{-3} a (S_n - \Delta)^{3.5} \text{ mb} \quad (16.1)$$

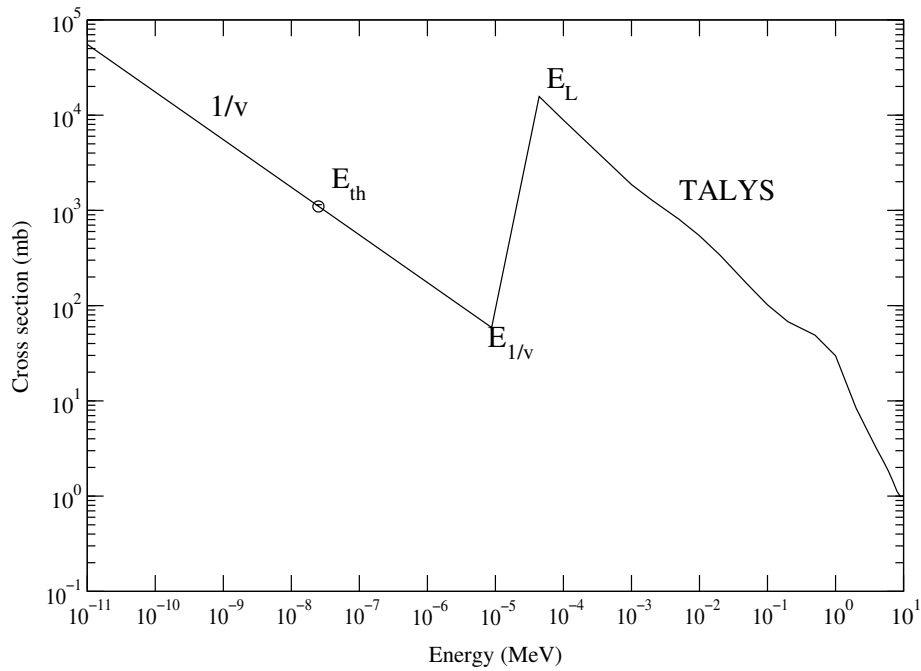


Figure 16.1: Capture cross section at low energies. The origin of the various energy regions are indicated.

with  $a$  the level density parameter at the separation energy  $S_n$  and  $\Delta$  the pairing energy. We assign a  $1/v$ , i.e.  $1/\sqrt{E}$ , dependence to the cross section from  $10^{-5}$  eV to an upper limit  $E_{1/v}$  which we set, again arbitrarily, at  $E_{1/v} = 0.2E_L$ . The  $1/v$  line obviously crosses  $\sigma_{n,\gamma}(E_{th})$  at  $E_{th}$ . The points at  $E_{1/v}$  and  $E_L$  are connected by a straight line. The resulting capture cross section then looks like Fig. 16.1. In reality, the region between  $E_{1/v}$  and  $E_L$  is filled with resolved resonances, the only feature we did not try to simulate.

## 16.2 Other non-threshold reactions

For other reactions with positive Q-values, such as  $(n,p)$  and  $(n,\alpha)$ , only a few experimental values at thermal energy are available and a systematical formula as for  $(n,\gamma)$  is hard to construct. If we do have a value for these reactions at thermal energy, the same method as for capture is followed. If not, we assume that the ratio between the gamma decay width and e.g. the proton decay width is constant for incident energies up to  $E_L$ . Hence, we determine  $R_p = \sigma_{n,p}/\sigma_{n,\gamma}$  at  $E_L$ , and since we know the thermal  $(n,\gamma)$  value we can produce the  $(n,p)$  excitation function down to  $10^{-5}$  eV by multiplying the capture cross section by  $R_p$ . A similar procedure is applied to all other non-threshold reactions.

## 16.3 Implementation, options and sample cases

**Implementation 16.3.1** In `talys/structure/thermal/`, the thermal cross sections for  $(n,\gamma)$ ,  $(n,p)$ ,  $(n,\alpha)$  and  $(n,f)$  cross sections are stored. The values come from Mughabghab [139] and Kopecky, who has compiled this database for use in the EAF library. In TALYS, we use this to determine cross sections for non-threshold reactions at low energies. For each isotope, we read  $Z$ ,  $A$ , target state (ground state or isomer), final state (ground state or isomer), the thermal



$(n, \gamma)$  cross section, its error, the thermal  $(n, p)$  cross section, its error, the thermal  $(n, \alpha)$  cross section, its error, the thermal  $(n, f)$  cross section, its error, nuclear symbol. The format is (4i4,8e9.2,7x,i4,a2). As an example, below are the values for the Fe-isotopes from *Fe.ther*.

26	54	0	0	2.25E+03	1.80E+02		1.00E-02	...	54Fe
26	55	0	0	1.30E+04	2.00E+03		1.70E+05	...	55Fe
26	56	0	0	2.59E+03	1.40E+02			...	56Fe
26	57	0	0	2.48E+03	3.00E+02			...	57Fe
26	58	0	0	1.32E+03	3.00E+01			...	58Fe
26	59	0	0	1.30E+04	3.00E+03		1.00E+04	...	59Fe

### Sample case 16.1 Low energy resonance data for $n + {}^{89}\text{Y}$

TALYS can read in resonance parameters, from various possible sources, and call the RECENT code of Red Cullen's PREPRO package, in TALYS included as a subroutine. Low energy pointwise resonance cross sections will then be added to channels like total, elastic, fission and capture.

#### Case a: Resonance input file: Standard

The input file is

```
#
# n-Y089-RRR-standard
#
# General
#
projectile n
element Y
mass 89
energy n0-2.grid
#
# Model
#
resonance y
#
# Output
#
channels y
filechannels y
```

where *n0-2.grid* is the built-in TALYS energy grid for neutrons between 0 and 2 MeV.

#### Case b: Resonance input file: Temperature broadening

The input file is

```
#
# n-Y089-RRR-temp
#
```

```

# General
#
projectile n
element Y
mass 89
energy n0-2.grid
#
# Model
#
resonance y
Tres 1200.
#
# Output
#
channels y
filechannels y

```

#### Case c: Resonance input file: Group structure

The input file is

```

#
# n-Y089-RRR-group
#
# General
#
projectile n
element Y
mass 89
energy n0-2.grid
#
# Model
#
resonance y
group y
#
# Output
#
channels y
filechannels y

```

The results are depicted in Fig. 16.2. Note the discontinuity between the end of the resonance range and the statistical model energy range of TALYS. Obviously, this needs to be studied case by case. ■

**Sample case 16.2** Unresolved resonance range parameters:  $n + {}^{136}\text{Ba}$

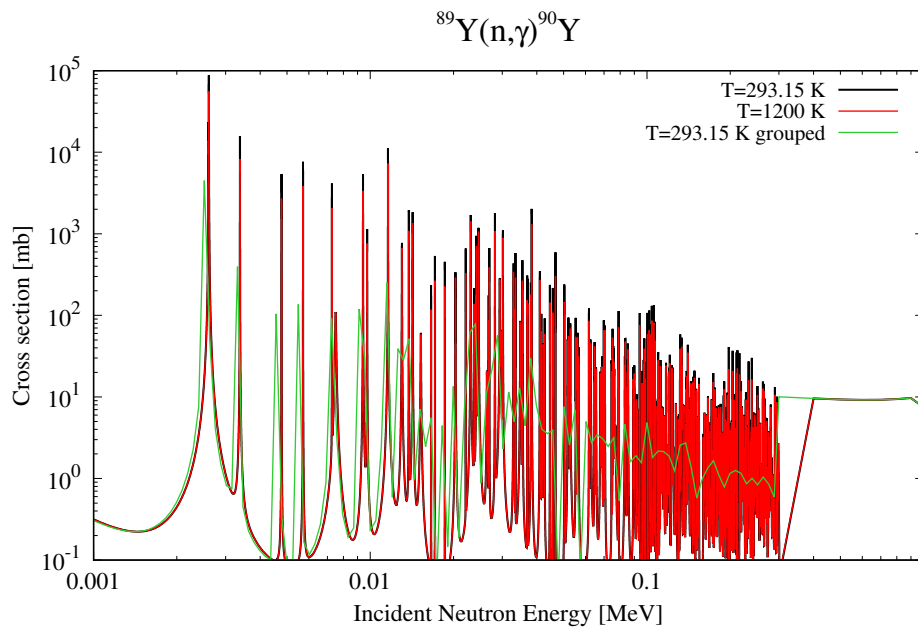


Figure 16.2:  $^{89}\text{Y}(n,\gamma)$  cross section, at 293.15 K, 1200 K, and grouped.

With TALYS, parameters for the unresolved resonance range (URR) can be produced. The following input file is used

```
#
# n-Ba136-URR
#
# General
#
projectile n
element   ba
mass     136
energy   energies
#
# Output
#
urr y
```

where the line **urr y** results in a set of URR output files which are described page 552. The top of the general output file *urr.dat* looks as follows:

```
#
# Resonance parameters for Z= 56 A= 136 (136Ba) Target spin= 0.0
# Thermal capture cross section= 6.70000E+02 mb   Sn= 6.90561E+00 MeV
#
# Einc[MeV]= 1.00000E-03
# Rprime[fm]= 4.39731E+00
# 1   J       D(1) [eV]   D(1,J) [eV]   S(1)         S(1,J)   Gx(1,J) [eV]  Gn(1,J) [e
```

```
0 0.5 1.20855E+03 1.20855E+03 1.80190E+00 1.80190E+00 0.00000E+00 2.17768E-
1 0.5 4.17392E+02 1.20855E+03 9.29906E-01 1.16255E+00 0.00000E+00 1.40500E-
1 1.5 4.17392E+02 6.37595E+02 9.29906E-01 8.52357E-01 0.00000E+00 5.43458E-
#
# Einc[MeV]= 2.00000E-03
# Rprime[fm]= 4.38034E+00
# l J D(l)[eV] D(l,J)[eV] S(l) S(l,J) Gx(l,J)[eV] Gn(l,J)[e
0 0.5 1.20703E+03 1.20703E+03 1.78817E+00 1.78817E+00 0.00000E+00 2.15838E-
```

## 17. Populated initial nucleus

Usually, a TALYS calculation will concern a projectile with a certain incident energy and a target, either in its ground or an excited state. For various reasons, we have introduced the possibility to start the decay from an initial population, i.e. an excited nucleus with a population distributed over excitation energy. An example of an interesting application is the neutron spectrum from fission fragments. One could calculate the fragment distribution from fission, e.g. as described in Section 15.7.3 or from empirical methods, and assume a population per excitation energy and spin, of the excited light and heavy fission fragment (models for such distributions exist, see e.g. Ref. [140]). This distribution can then be the starting point for a TALYS calculation. The initial population enters the Hauser-Feshbach scheme and the compound nucleus calculation proceeds as usual. The emitted neutrons can be recorded as well as the path from fission fragment to fission product. All relevant nuclear structure quantities are available since we simulate the process by a photon-induced reaction, the only difference being that we do not excite a single compound nucleus energy but directly fill the continuum bins and discrete levels according to our specified starting population. TALYS can be used in this mode by simply specifying **projectile 0**, (p. 278), and providing an energy file as input, see **energy** (p. 281). The initial population can be provided at two levels of detail. A full excitation energy-spin-parity population can be given, which is then interpolated on the internal excitation energy scheme of TALYS. Alternatively, only the total population per excitation energy can be given, after which the spin-parity-dependent population is determined by multiplying it with the spin distribution of Eq. (8.8). For the fission neutron spectrum example mentioned above, one could loop over all fission fragments (by writing a clever script), sum the results, and obtain the fission neutron spectrum. There are more applications for this feature, such as coupling a high-energy intranuclear cascade code with TALYS, the latter taking care of the low energy evaporation part including all its quantum-mechanical conservation rules.



## 18. Astrophysical reaction rates

A complete calculation of astrophysical reaction rates is possible with TALYS. In stellar interiors, nuclides not only exist in their ground states but also in different thermally excited states and a thermodynamic equilibrium holds locally to a very good approximation. Therefore, most of nuclear astrophysics calculations have made use of nuclear reaction rates evaluated within the statistical model [141]. The assumption of a thermodynamic equilibrium combined with the compound nucleus cross sections for the various excited states then allows to produce Maxwellian-averaged reaction rates, which is important input for stellar evolution models. Calculation of stellar reaction rates is obviously not new, but TALYS provides some features which automatically makes the extension to reaction rate calculations very worthwhile. In contrast with existing dedicated astrophysical reaction rate codes, the present Chapter shows that we provide the inclusion of pre-equilibrium reaction mechanism, the detailed competition between all open channels, the inclusion of multi-particle emission (neglected in most astrophysics codes), the inclusion of detailed width fluctuation corrections, the inclusion of parity-dependent level densities, the inclusion of coupled channel description for deformed nuclei, and the coherent inclusion of fission channel. Different prescriptions are also used when normalizing nuclear models on available experimental data, such as level densities on s-wave spacings or E1 resonance strength on photoabsorption data.

The energies of both the targets and projectiles, as well as their relative energies  $E$ , obey Maxwell-Boltzmann distributions corresponding to the temperature  $T$  at that location (or a black-body Planck spectrum for photons). The astrophysical rate is obtained by integrating the cross section given by Eq. (13.2) over a Maxwell-Boltzmann distribution of energies  $E$  at the given temperature  $T$ . In addition, in hot astrophysical plasmas, a target nucleus exists in its ground as well as excited states. In a thermodynamic equilibrium situation, the relative populations of the various levels of nucleus with spins  $I^\mu$  and excitation energies  $E_x^\mu$  obey a Maxwell-Boltzmann distribution. Hence, in the formulae to follow, it is understood that the definition of the incident  $\alpha$  channel, see below Eq. (13.2), now includes an explicit superscript  $\mu$  to distinguish between the

excited states. The effective stellar rate of  $\alpha \rightarrow \alpha'$  in the entrance channel at temperature  $T$  taking due account of the contributions of the various target excited states is finally expressed as

$$N_A \langle \sigma v \rangle_{\alpha\alpha'}^*(T) = \left( \frac{8}{\pi m} \right)^{1/2} \frac{N_A}{(kT)^{3/2} G(T)} \int_0^\infty \sum_\mu \frac{(2I^\mu + 1)}{(2I^0 + 1)} \times \sigma_{\alpha\alpha'}^\mu(E) E \exp\left(-\frac{E + E_x^\mu}{kT}\right) dE, \quad (18.1)$$

where  $k$  is the Boltzmann constant,  $m$  the reduced mass of the  $\alpha$  channel,  $N_A$  the Avogadro number, and

$$G(T) = \sum_\mu (2I^\mu + 1) / (2I^0 + 1) \exp(-E_x^\mu / kT) \quad (18.2)$$

the  $T$ -dependent normalized partition function. Reverse reactions can also be estimated making use of the reciprocity theorem [141]. In particular, the stellar photodissociation rates are classically derived from the radiative capture rates by

$$\lambda_{(\gamma,\alpha)}^*(T) = \frac{(2I + 1)(2j + 1)}{(2I' + 1)} \frac{G_I(T)}{G_{I'}(T)} \left( \frac{AA_a}{A'} \right)^{3/2} \left( \frac{kT}{2\pi\hbar^2 N_A} \right)^{3/2} \times N_A \langle \sigma v \rangle_{(\alpha,\gamma)}^* e^{-Q_{\alpha\gamma}/kT}, \quad (18.3)$$

where  $Q_{\alpha\gamma}$  is the Q-value of the  $I^0(\alpha, \gamma)I'^0$  capture channel. Note that, in stellar conditions, the reaction rates for targets in thermal equilibrium are usually believed to obey reciprocity since the forward and reverse channels are symmetrical, in contrast to the situation which would be encountered for targets in their ground states only [141]. In TALYS, the total stellar photodissociation rate is determined from

$$\lambda_{(\gamma,j)}^*(T) = \frac{\sum_\mu (2J^\mu + 1) \lambda_{(\gamma,\alpha)}^\mu(T) \exp(-E_x^\mu / kT)}{\sum_\mu (2J^\mu + 1) \exp(-E_x^\mu / kT)}, \quad (18.4)$$

where the photodissociation rate  $\lambda_{(\gamma,\alpha)}^\mu$  of state  $\mu$  with excitation energy  $E_x^\mu$  is given by

$$\lambda_{(\gamma,\alpha)}^\mu(T) = \int_0^\infty c n_\gamma(E, T) \sigma_{(\gamma,\alpha)}^\mu(E) dE, \quad (18.5)$$

where  $c$  is the speed of light,  $\sigma_{(\gamma,j)}^\mu(E)$  the photodisintegration cross section at energy  $E$ , and  $n_\gamma$  the stellar  $\gamma$ -ray distribution well described by the black-body Planck spectrum at the given temperature  $T$ .

**Options 18.0.1** In TALYS, if **astro y**, an appropriate incident energy grid for astrophysical calculations is made which overrules any incident energy given by the user. Inclusion of ground state only, or particular excited states can be steered with **astrogs** (p. 643), **nonthermlev** (p. 647) and **astroex** (p. 646). With **astroE** (p. 644) the energy of Maxwellian average can be chosen, which one usually wants to put at 30 keV. Also the temperature can be set with **astroT** (p. 645).

**Implementation 18.0.1** The implementation of astrophysical reaction rates takes place in *astro.f*.



### Sample case 18.1 Astrophysical reaction rates : $n + {}^{187}\text{Os}$

With TALYS, astrophysical reaction rates can be calculated. As sample case, we took the work done in Ref.[142] where the  ${}^{187}\text{Os}(n,\gamma)$  was studied for the derivation of the age of the galaxy withing the Re-Os cosmochronology.

#### Case a: ${}^{187}\text{Os}(n,\gamma)$ cross section

First, the calculated  ${}^{187}\text{Os}(n,\gamma)$  was compared with experimental data, using the following input file

```
#
# n-0s187-astro-ng
#
# General
#
projectile n
element    os
mass      187
energy    energies
channels  y
filechannels y
#
# Parameters
#
best y
```

The results are given in Fig.18.1.

#### Case b: ${}^{187}\text{Os}(n,\gamma)$ astrophysical reaction rate

Next, the astrophysical reaction rates for neutrons on  ${}^{187}\text{Os}$  are computed with the following input file

```
#
# n-0s187-astro-rate
#
# General
#
projectile n
element    os
mass      187
energy    1.
#
# Parameters
#
ldmodel    5
asys      n
strength   2
```

```
gnorm 0.27
astro y
astrog n
partable y
```

which produces various output files in which the reaction rates as function of temperature are given. The  $(n, \gamma)$  rate as given in the output file is

```
8. Thermonuclear reaction rates
Reaction rate for Z= 76 A=188 (188Os)
  T          G(T)          Rate
0.0001 1.00000E+00 5.46242E+08
0.0005 1.00000E+00 4.96512E+08
0.0010 1.00000E+00 4.45882E+08
0.0050 1.00000E+00 3.52813E+08
0.0100 1.00002E+00 3.01755E+08
0.0500 1.20827E+00 2.03467E+08
0.1000 1.64630E+00 1.87256E+08
0.1500 1.95860E+00 1.82416E+08
0.2000 2.21894E+00 1.80436E+08
0.2500 2.48838E+00 1.79690E+08
0.3000 2.79031E+00 1.79474E+08
0.4000 3.49874E+00 1.79716E+08
0.5000 4.31540E+00 1.80165E+08
0.6000 5.20597E+00 1.80375E+08
0.7000 6.15215E+00 1.80187E+08
0.8000 7.14759E+00 1.79580E+08
0.9000 8.19300E+00 1.78579E+08
1.0000 9.29313E+00 1.77221E+08
.....
```

The same numbers can be found in the separate file *astrorate.g*. In *astrorate.tot* the rates for all reactions can be found.

### Sample case 18.2 Maxwellian averaged cross section at 30 keV: $n + {}^{138}\text{Ba}$

In this sample case, the Maxwellian averaged cross section (MACS) at 30 keV is calculated. The following input file is used

```
#
# n-Ba138-MACS
#
# General
#
projectile n
element    ba
mass      138
energy    1.
#
```

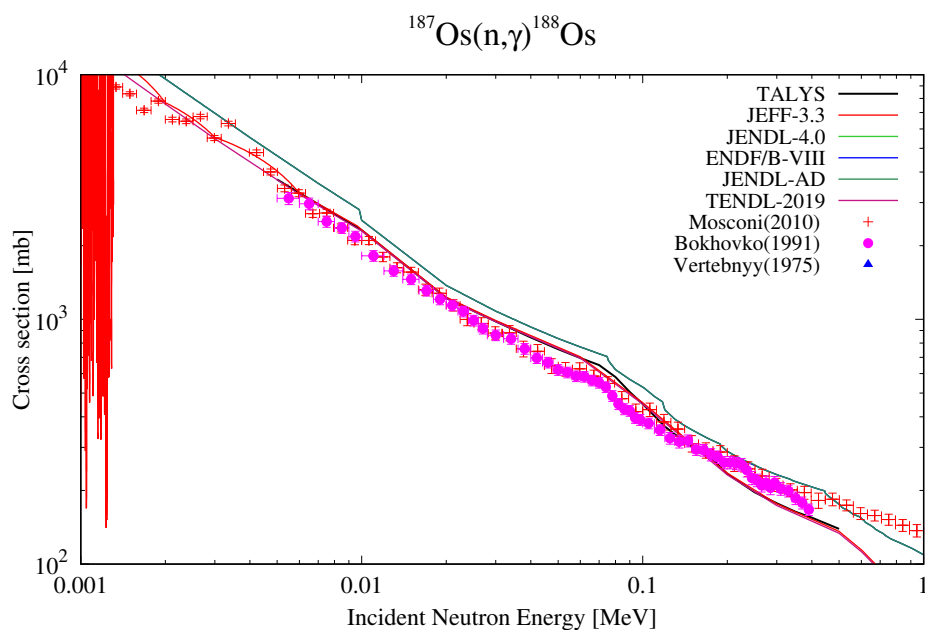


Figure 18.1:  $^{187}\text{Os}(n,\gamma)$  cross section.

```
# Output
#
astroE 0.030
```

where the line **astroE 0.03** determines the energy in MeV for which the average is calculated. Note that the line **energy 1.** is irrelevant, but must nevertheless be given. The file *astrorate.g* looks as follows:

```
# Reaction rate for 138Ba(n,g) at <E>= 0.03000 MeV (Excited States Contribution : n)
#   T      Rate      MACS
0.3481 5.49792E+05 3.79831E+00
```

in which the temperature, reaction rate and MACS is given. For this case, the deviation from a normal 30 keV calculation (which can be obtained by removing the **astroE** line and using **energy 0.03**) is small, since the latter is 3.75 mb. ■



## 19. Medical isotope production

This section describes the formalism to calculate the production yield of a radioactive nucleus by a nuclear reaction. Starting from the general formalism we will introduce realistic approximations and derive simpler equations that hold in basically all cases of interest. In TALYS we have implemented only analytical solutions to the production and depletion equations. This may be generalized in future versions.

### 19.1 Production and depletion of isotopes

The most general situation is that of the irradiation of a piece of material, consisting of many different isotopes, which were either present at the start of the irradiation, have been formed, or will be formed during the irradiation, by either the primary flux of particles, by secondary particles, or by radioactive decay. If we have a total of  $K$  different isotopes (theoretically  $K$  may stand for the entire nuclide chart) and the number of each isotope  $k$  is  $N_k$ , then the temporal development of such a system is described by  $K$  differential equations:

$$\begin{aligned}\frac{dN_1(t)}{dt} &= \sum_{k=1, k \neq 1}^K \Lambda_{k \rightarrow 1} N_k(t) - \Lambda_1 N_1(t) \\ &\dots \\ \frac{dN_i(t)}{dt} &= \sum_{k=1, k \neq i}^K \Lambda_{k \rightarrow i} N_k(t) - \Lambda_i N_i(t) \\ &\dots \\ \frac{dN_K(t)}{dt} &= \sum_{k=1, k \neq K}^K \Lambda_{k \rightarrow K} N_k(t) - \Lambda_K N_K(t)\end{aligned}\tag{19.1}$$

In each equation, the first term is a feeding term. In the most general case, various parent nuclides may contribute to the formation of isotope  $i$ , hence the summation over parent nuclides  $k$ , with

$\Lambda_{k \rightarrow i}$  the partial formation rate for any possible parent isotope  $k$  to isotope  $i$ . Each partial formation rate can be expressed as

$$\Lambda_{k \rightarrow i} = \lambda_{k \rightarrow i} + R_{k \rightarrow i} \quad (19.2)$$

with  $\lambda_{k \rightarrow i}$  the (partial) radioactive decay rate and  $R_{k \rightarrow i}$  the (partial) nuclear reaction rate for any possible parent isotope  $k$  to isotope  $i$ . The second terms of Eq. (19.1) are loss terms due to radioactive decay and nuclear reactions from isotope  $i$  to any other isotope. Here the total depletion rate (we interchange “depletion” with “formation” whenever appropriate) for isotope  $i$  is

$$\Lambda_i = \lambda_i + R_i \quad (19.3)$$

where the total decay rate for isotope  $i$  is

$$\lambda_i = \sum_{k=1, k \neq i}^K \lambda_{i \rightarrow k}$$

and the total nuclear reaction rate for isotope  $i$  is

$$R_i = \sum_{k=1, k \neq i}^K R_{i \rightarrow k} \quad (19.4)$$

The entire loop over  $k$  may run over isotopes in their ground or isomeric states. Theoretically, the sum over reaction rates could include secondary particles (neutrons, photons, alpha particles etc.) formed after the first interaction of the incident beam with the material, over the entire outgoing energy spectrum. Since the number of isotopes  $i$ ,  $N_i(t)$  may appear simultaneously in many equations, due to its possible formation, or depletion, by many different nuclear reactions, it is clear that such a coupled system can only be solved by complicated mathematical and computational techniques. In fact, the most exact simulation would involve a Monte Carlo 3D transport calculation in which all primary and secondary particles are taken into account, including complete cross section libraries for all possible particles, coupled with an activation code that keeps track of the nuclide inventory. If we neglect such thick target transport issues, a system of equations like (19.1) is often solved by methods developed by Bateman [143] for radioactive decay and later generalized for source terms by Rubinson [144].

Fortunately, the situation is often not as complex as sketched above, since very reasonable approximations can be introduced into Eq. (19.1), certainly for charged-particle induced reactions. First, let us start with the common case of the irradiation of a target which contains of only one natural element at the start of the irradiation. This set of equations can be then be separated into a linear combination of contributions by each target *isotope*. Hence, we solve Eq. (19.1) one by one for each target isotope and add these contributions at the end to get the answer for the target material. Then, for such a mono-isotopic target  $T$  we have at the start of the irradiation

$$\begin{aligned} t = 0 : N_T &= N_T(0) \\ N_i &= 0 \\ &\dots \\ N_K &= 0 \end{aligned} \quad (19.5)$$

Since in practice our target isotope is not radioactive, the loss terms reduce to  $\Lambda_T = R_T$  in Eq. (19.1) for  $N_T$ . Next, if a substantial part of the target is converted into other isotopes, beam particles

may interact with atoms other than the original target atoms. However, this is only a concern for irradiation times of the order of months or longer. For most practical applications, like medical isotope production, we can often assume that the burn-up of the target is small and that the target composition does not change much during the irradiation (this will be confirmed by some of our sample cases). Hence, in addition we will assume there are no nuclear reaction or radioactive decay feeding terms for the *target* isotope  $T$ .

This means that the target isotope is described by the following simple equation

$$\frac{dN_T(t)}{dt} = -R_T N_T(t) \quad (19.6)$$

which basically assumes that the only thing that happens to the target material is burn-up through nuclear reactions.

For the isotopes that are produced during the irradiation we can make similar reasonable assumptions. We assume there is no loss of a *produced* isotope of interest through nuclear reactions with beam particles, i.e. we assume nuclides are not hit twice (again, this assumption becomes less accurate at very long irradiation times). This is equivalent to stating that the isotopes of interest are only produced from nuclear reactions on the target isotopes or from decay of other products formed during the irradiation. Also, consistent with the assumption for the target isotope that nuclides are not hit twice, we have  $\Lambda_i = \lambda_i$  for the depletion term, and assume that no other nuclear reactions lead to the isotope of interest. Thus for the produced isotopes  $i$  we obtain

$$\frac{dN_i(t)}{dt} = \sum_{k=1, k \neq i}^K \lambda_{k \rightarrow i} N_k(t) + R_{T \rightarrow i} N_T(t) - \lambda_i N_i(t) \quad (19.7)$$

Nuclides often have only one or a few radioactive decay modes. Usually, only beta decay to the ground state or an isomer needs to be considered, although in some cases alpha decay may occur as well. If we neglect alpha decay, the summation in the first term above reduces to only one term. Thus, Eq. (19.7) reduces to

$$\frac{dN_i(t)}{dt} = \lambda_{p \rightarrow i} N_p(t) + R_{T \rightarrow i} N_T(t) - \lambda_i N_i(t) \quad (19.8)$$

where we now use the subscript  $p$  for the parent isotope which decays to isotope  $i$ . The parent isotope itself is described by the equation,

$$\frac{dN_p(t)}{dt} = R_{T \rightarrow p} N_T(t) - \lambda_p N_p(t) \quad (19.9)$$

where for simplicity we have left out a possible feeding term  $\lambda_{g \rightarrow p} N_g(t)$  from its own parent (i.e. the grandparent of isotope  $i$ ). It neglects radioactive decay to channels produced by multiple proton emission, e.g.  $^{120}\text{Te}(p,2p)^{119}\text{Sb}$  and the contribution from its feeding channel  $^{120}\text{Te}(p,2n)^{119}\text{I} + 2\beta^+$ . Although the (p,2p) channel is generally smaller and therefore often not relevant for most practical medical isotope production routes, it would be safer that at higher energies feeding from radioactive decay is taken into account. For that a more general solution of Eq. (19.1) would have to be implemented, similar to solutions for neutron-induced problems where processes like e.g. multiple fission yield decay need to be taken into account. For most charged-particle induced reactions of interest, the current approximation is justified.

If we now use the boundary conditions  $N_T(t=0) = N_T(0)$  and  $N_p(t=0) = N_i(t=0) = 0$ , the solutions to these equations are given by:

$$\begin{aligned} N_T(t) &= N_T(0) e^{-R_T t}, \\ N_p(t) &= N_T(0) \frac{R_{T \rightarrow p}}{\lambda_p - R_T} [e^{-R_T t} - e^{-\lambda_p t}], \\ N_i(t) &= N_T(0) \frac{R_{T \rightarrow i}}{\lambda_i - R_T} [e^{-R_T t} - e^{-\lambda_i t}] + N_T(0) \frac{R_{T \rightarrow p} \lambda_{p \rightarrow i}}{\lambda_p - R_T} \left[ \frac{e^{-R_T t} - e^{-\lambda_i t}}{\lambda_i - R_T} - \frac{e^{-\lambda_p t} - e^{-\lambda_i t}}{\lambda_i - \lambda_p} \right] \end{aligned} \quad (19.10)$$

This is derived in the Appendix.

By setting the derivative of  $N_i(t)$  to zero, the irradiation time for which a maximal yield is obtained can be easily derived:

$$t_{max} = \frac{\ln(\lambda_i/R_T)}{\lambda_i - R_T} \quad (19.11)$$

The production of  $i$  is restrained by its  $\beta$ -decay on the one hand and by the decreasing number of available target atoms on the other hand. Hence, this observable  $t_{max}$  depends on the decay constant  $\lambda_i$  and the total production rate  $R_T$ .

Now that the analytical formulae for production are set with Eq. (19.10), the remaining ingredients needed to evaluate Eqs.(19.10) and (19.11) are:  $N_1(0), \dots, N_I(0)$ ,  $R$ , and  $\lambda$ , and this will be done in the next two sections.

## 19.2 Initial condition and stopping power

The number of target atoms at  $t=0$ ,  $N_i(0)$ , for  $1 \leq i \leq I$  equals:

$$N_i(0) = \frac{N_A}{A} B_i \rho V_{tar}, \quad (19.12)$$

where  $N_A = 6.022 \cdot 10^{23}$  is Avogadro's number,  $A$  is the mass number,  $B_i$  is the abundance of isotope  $i$  with  $\sum_i^I B_i = 1$ ,  $\rho$  the mass density in  $[\text{g}/\text{cm}^3]$ , and  $V_{tar}$  the active target volume in  $[\text{cm}^3]$ .  $V_{tar}$  is given by the product of the beam surface  $S_{beam}$  in  $[\text{cm}^2]$  and the effective target thickness in  $[\text{cm}]$ , which can be expressed in terms of the stopping power  $\frac{dE}{dx}$ ,

$$V_{tar} = S_{beam} \int_{E_{back}}^{E_{beam}} \left( \frac{dE}{dx} \right)^{-1} dE, \quad (19.13)$$

where  $E_{beam}$  denotes the incident beam energy and  $E_{back}$  is the average projectile energy available at the backside of the target. If the projectiles travels through the target, the average projectile beam energy will decrease. The amount of energy loss inside the target is determined by the target thickness and the stopping power. The integration limits  $E_{beam}$  and  $E_{back}$  are fixed by the requested projectile energy range inside the target, which is determined by the cross section as function of projectile energy (excitation function). This formula neglects the spreading of the beam inside the target.

The stopping power describes the average energy loss of projectiles in the target by atomic collisions as a function of their energy in  $[\text{MeV}/\text{cm}]$ . We use the Bethe-Bloch formula [145]:

$$\frac{dE}{dx} = 0.1535 \rho \frac{Z}{A} \frac{z_p^2}{\beta^2} \left[ \ln \left( \frac{2m_e \gamma^2 v^2 W_{max}}{I^2} \right) - 2\beta^2 \right], \quad \text{MeV}/\text{cm} \quad (19.14)$$



with  $Z$  the target charge number, and  $z_p$  the projectile charge number, while  $\beta$  represents a beam particle traveling at a relative velocity

$$\beta = \frac{v}{c} = \sqrt{\frac{E_{beam}(E_{beam} + 2m_0c^2)}{(E_{beam} + m_0c^2)^2}}, \quad (19.15)$$

with rest mass  $m_0$ . Further,  $m_e$  is the electron mass, and  $\gamma = \frac{1}{\sqrt{1-\beta^2}}$ . The maximum energy transfer in a single collision,  $W_{max}$  is given by

$$W_{max} = 2m_e c^2 (\beta\gamma)^2, \quad (19.16)$$

if the incident particle is much heavier than the electron mass. For the mean excitation potential  $I$  a semi-empirical formula is adopted:

$$\frac{I}{Z} = 9.76 + 58.8Z^{-1.19} \text{ eV}. \quad (19.17)$$

This expression is claimed to be tested only if  $Z \geq 13$ , but we use it for all values.

### 19.3 Nuclear reaction and decay rates

When the full beam hits the target (i.e., assuming that the beam diameter is smaller than the target dimensions), the production rate in [atoms/s] of isotope  $i$  through the nuclear reaction on the target isotope  $T$  is given by the following expression

$$R_{T \rightarrow i} = \frac{I_{beam}}{z_p q_e V_{tar}} \frac{1}{\int_{E_{back}}^{E_{beam}} \left(\frac{dE}{dx}\right)^{-1} \sigma_i^{rp}(E) dE} \quad (19.18)$$

where  $I_{beam}$  is the beam current in [A] and  $q_e$  is the electron charge. The factor  $I_{beam}/(z_p q_e)$  corresponds to the number of projectiles impinging on the target per [s]. The residual production cross section of  $i$  in [mb] is denoted by  $\sigma_i^{rp}(E)$ .

Analogously, the production rate in [ $s^{-1}$ ] of all reaction channels, from the target, is given by:

$$R_T = \frac{I_{beam}}{z_p q_e V_{tar}} \frac{1}{\int_{E_{back}}^{E_{beam}} \left(\frac{dE}{dx}\right)^{-1} (\sigma_{non}(E) - \sigma_{in}(E)) dE} \quad (19.19)$$

where  $\sigma_{non}(E)$  is the non-elastic cross section and  $\sigma_{in}(E)$  is the inelastic cross section of  $i$  in [mb], which is nothing else than the residual production cross section  $\sigma_i^{rp}(E)$ . The difference  $\sigma_{non}(E) - \sigma_{in}(E)$  provides the probability to create an isotope different from the original target atom in the nuclear reaction.

The decay rate of  $i$  is given by the simple relation

$$\lambda_i = \frac{\ln 2}{T_i^{1/2}}, \quad (19.20)$$

where  $T_i^{1/2}$  is the half-life of isotope  $i$  in [s]. The  $\lambda_i$  in the loss term usually feeds only one or two different channels, namely by beta decay to the ground state or isomer of the daughter isotope  $d$ ,

$$\lambda_i = \lambda_{i \rightarrow d} = \lambda_{i \rightarrow d}^{g.s.} + \lambda_{i \rightarrow d}^{iso}. \quad (19.21)$$

With this, all ingredients of Eq. (19.10) are defined and we may calculate the nuclide inventory  $N_i(t)$  at any time during or after the irradiation process.

## 19.4 Radioactive activity

With  $N_i(t)$  known, the final expression for the activity of the produced isotope  $i$ ,  $A_i$ , as a function of the irradiation time  $t$  can now be given by

$$A_i(t) = \lambda_i N_i(t). \quad (19.22)$$

For small irradiation times the expression for  $N_i(t)$ , Eq. (19.10), behaves as:

$$N_i(t) = N_T(0) R_{T \rightarrow i} t, \quad (19.23)$$

and hence the activity of  $i$  as

$$A_i(t) = \lambda_i N_T(0) R_{T \rightarrow i} t, \quad (19.24)$$

Under these circumstances, the yield scales linearly with the irradiation time  $t$  and the production rate  $R_{T \rightarrow i}$ . Only then, the production yield expressed in [MBq/mAh] is a meaningful quantity, which can be used to determine the yield, given a certain irradiation time and beam current.

**Options 19.4.1** With **production** (p. 650), the calculation of medical isotope production is invoked. Characteristics of the accelerator can be set with **Ebeam** (p. 651), **Eback** (p. 652), **Ibeam** (p. 653), **Area** (p. 654) and **rho** (p. 655). Irradiation parameters are set with **Tirrad** (p. 658), **Tcool** (p. 659), **radiunit** (p. 656), and **yieldunit** (p. 657).

### Sample case 19.1 Medical isotope production with $p + {}^{100}\text{Mo}$

In this sample case, the activity of all the produced isotopes from the reaction  $p + {}^{100}\text{Mo}$  are given, for a  $150 \mu\text{A}$  proton accelerator, as a function of time. The following input file is used

```
#
# p-Mo100-medical
#
# General
#
projectile p
element Mo
mass 100
energy 8. 30. 0.5
#
# Spherical OMP
#
spherical y
#
# Medical isotope production
#
production y
Ibeam 0.15
Ebeam 24.
Eback 10.
```

Note the use of the **energy** keyword: Cross sections are calculated for incident energies

between 8 and 30 MeV with 0.5 MeV energy steps. The keyword **production y** means that activities of all radioactive products as a function of bombarding and/or decay time will be produced in tables. The incident energy of the particle beam is 24 MeV, and the required thickness of the target (which is given in the output) is such that at the back of the target the energy is 10 MeV. Other keywords relevant for isotope production, such as **Tcool**, see page 659, **Tirrad**, see page 658, **rho**, see page 655, **area**, see page 654, **radiounit**, see page 656, **yieldunit**, see page 657, have their default values in this sample case.

We like to point out an efficient option for medical isotope calculations. Suppose one is interested in varying several accelerator options, such as e.g. **Ebeam**. Once the first calculation, like the sample case above, has been done, one may rerun that case and add the extra line **reaction n** to the input file. Then, TALYS will *not* redo the nuclear reaction calculations, but use the various *rpZZZAAA.tot, etc.* cross section output files which are already available in the work directory. Obviously, **Ebeam** and **Eback** must always fall inside the energy grid for which the cross sections are calculated, in this case between 8 and 30 MeV.

The tail of the standard output files contains some general parameters:

#### Summary of isotope production

```
Maximal irradiation time : 0 years 1 days 0 hours 0 minutes 0 sec
Cooling time             : 0 years 1 days 0 hours 0 minutes 0 sec
Energy range            : 24.000 --> 10.000 MeV
Beam current            : 0.150 mA
Target material density : 10.220 g/cm^3
Target area             : 1.000 cm^2
Effective target thickness : 0.085 cm
Effective target volume  : 0.085 cm^3
Effective target mass    : 0.865 g
Number of target atoms   : 5.20730E+21
Number of incident particles: 9.36226E+14 s^-1
Produced heat in target  : 2.100 kW
```

More interesting are probably the files with the produced activity (in Bq or Ci) per isotope. These yields are in files *YZZZAAA.tot* where *ZZZ* is the charge number and *AAA* is the mass number in (i3.3) format. For produced isomers, and ground states, there are additional files *pZZZAAA.LMM*, where *MM* is the number of the isomer (ground state=0) in (i2.2) format. For the above sample case, *Y043099.L02*, i.e. the production of <sup>99m</sup>Tc looks as follows:

```
# Reaction: p + 100Mo Production of 99Tc Isomer= 1
# Half life           : 0 years 0 days 6 hours 0 minutes 36 seconds
# Maximum production at: 0 years 3 days 18 hours 0 minutes 32 seconds
# Initial production rate: 1.08493E-10 [s^-1] Decay rate: 3.20368E-05 [s^-1]
# # time points =100
# Time [h] Activity [GBq] #isotopes [ ] Yield [GBq/mAh] Isotopic frac.
0.5 3.16572E+01 9.88152E+14 4.22096E+02 0.11878
1.0 6.15404E+01 1.92093E+15 3.98443E+02 0.11551
1.5 8.97491E+01 2.80144E+15 3.76115E+02 0.11233
2.0 1.16377E+02 3.63261E+15 3.55039E+02 0.10925
2.5 1.41513E+02 4.41720E+15 3.35144E+02 0.10628
3.0 1.65240E+02 5.15783E+15 3.16363E+02 0.10342
3.5 1.87638E+02 5.85695E+15 2.98635E+02 0.10067
```

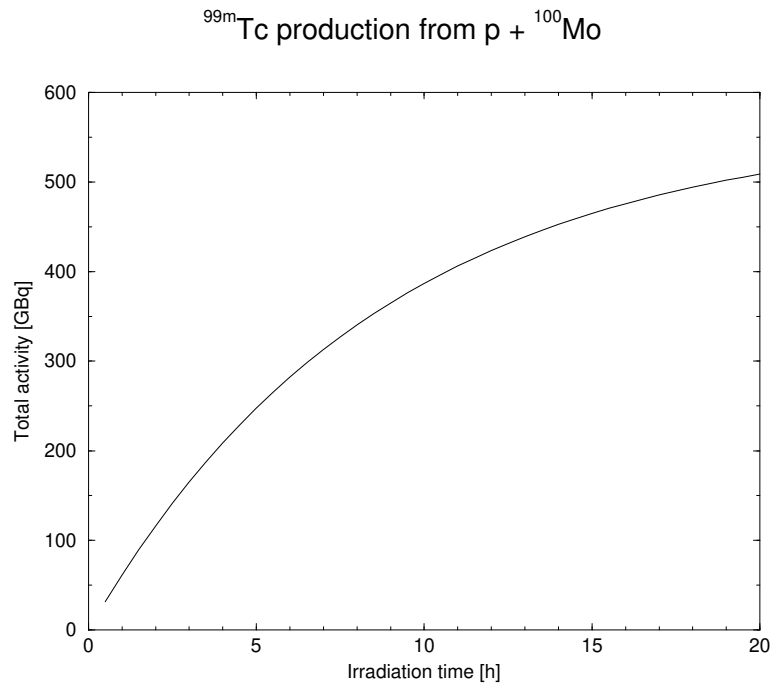


Figure 19.1: Total activity of  $^{99m}\text{Tc}$  produced by a 24 MeV proton accelerator of  $150\ \mu\text{A}$  and a  $^{100}\text{Mo}$  target with an energy of 10 MeV at the back of the target.

4.0	2.08780E+02	6.51689E+15	2.81900E+02	0.09801
4.5	2.28738E+02	7.13986E+15	2.66103E+02	0.09545
5.0	2.47577E+02	7.72791E+15	2.51191E+02	0.09298
5.5	2.65361E+02	8.28301E+15	2.37115E+02	0.09060
6.0	2.82148E+02	8.80701E+15	2.23828E+02	0.08831
6.5	2.97994E+02	9.30164E+15	2.11285E+02	0.08609
7.0	3.12953E+02	9.76855E+15	1.99445E+02	0.08396
7.5	3.27073E+02	1.02093E+16	1.88268E+02	0.08190
8.0	3.40402E+02	1.06253E+16	1.77718E+02	0.07991

where the final column denotes the fraction of the produced isotope relative to all isotopes of that element. Fig. 19.1 shows the  $^{99m}\text{Tc}$  production as a function of time. Note that the curve is not a straight line due to the decay of  $^{99m}\text{Tc}$  to the ground state. ■

## 20. Verification and validation

TALYS has been tested both formally ("computational robustness") and by comparison of calculational results with experimental data. This will be described in the present Chapter.

### 20.1 Robustness test with DRIP

One way to test a nuclear model code is to let it calculate results for a huge number of nuclides, and the whole range of projectiles and energies. We have written a little code DRIP, not included in the release, that launches TALYS for complete calculations for all nuclides, from dripline to dripline. Besides checking whether the code crashes, visual inspection of many curves simultaneously, e.g. (n,2n) excitation functions for 50 different targets, may reveal non-smooth behaviour. Various problems, not only in the TALYS source code itself, but also in the nuclear structure database, were revealed with this approach in the initial development stages.

### 20.2 Robustness test with MONKEY

Part III contains a description of the close to 400 keywords that can be used in a problem for TALYS. Some of them are flags which can only be set to **y** or **n**. Other keywords can be set to a few or several integer values and there are also keywords whose values may cover a quasi-continuous range between the minimum and maximum possible value that we allow for them. Strictly speaking, the total number of possible input files for TALYS is huge (though theoretically finite, because of the finite precision of a computer), and obviously only a small subset of them corresponds to physically meaningful cases. Indeed, as with many computer codes it is not too difficult to bring a TALYS calculation to a formally successful end, i.e. without crashing, with an input file that represents a completely unphysical case. Obviously, there is no way to prevent this - the user is supposed to have full responsibility for what she or he is doing - and we can never anticipate what

input files are made by other users. Nevertheless, to test the robustness of our code, we wrote a program called MONKEY which remotely simulates TALYS in the hands of an evil user: It creates an input file for TALYS, using *all* the keywords, with *random* values, and starts a run with it. Each keyword in this input file has a value that remains within the allowed range as specified in Part III. If TALYS is compiled with all checking options (i.e. array out-of-bounds checks, divisions by zero, other over/underflow errors, etc.) enabled and runs successfully for thousands of such random input files without crashing, we can at least say we have tested *computational* robustness to some extent, and that we *may* have probed every corner of the code. We realize that this is not a full-proof method to test TALYS formally on the Fortran level (if there exists such a test!), but MONKEY has helped us to discover a few bugs during development which otherwise would have come to the surface sooner or later. We think it is better that *we* find them first. The ideal result of this procedure would be that TALYS never crashes or stops without giving an appropriate error message. We emphasize that this test alone does obviously not guarantee any physical quality of the results. For that, much more important are the input files that *are* physically meaningful. These are discussed in the next Section.

### 20.3 Sample cases

With this manual, we hope to convince you that a large variety of nuclear reaction calculations can be handled with TALYS. To strengthen this statement, we have included many different sample cases throughout this tutorial. In each case, the TALYS input file, the relevant output and, if available, a graphical comparison with experimental data has been presented. The entire collection of sample cases serves as (a) verification of TALYS: our sample output files should coincide, apart from numerical differences of insignificant order, with the output files obtained on your computer, and (b) validation of TALYS: the results should be comparable to experimental data for a widely varying set of nuclear reactions.

It may take a while (about 6 hours on a Macbook Pro) before all sample cases have finished. Running the *verify* script included in the TALYS package will be worth the wait however, since a successful execution of all sample cases will put you on safer ground for your own calculations.

Below are the sample cases described in this tutorial and to be found in the *talys/samples* directory,

1. n-Nb093-14MeV: **All results for 14 MeV n +  $^{93}\text{Nb}$** , page 66
2. n-Pb208-xs: **Excitation functions:  $^{208}\text{Pb}(n,n')$ ,  $(n,2n)$ ,  $(n,p)$  etc**, page 75
3. p-Fe000-rp: **Residual production cross sections for protons on Fe**, page 79
4. n-Pb208-nngamma: **Gamma-ray intensities:  $^{208}\text{Pb}(n,n\gamma)$  and  $^{208}\text{Pb}(n,2n\gamma)$** , page 80
5. n-Bi209-spec: **Inelastic spectra at 20 MeV: Direct + Preeq + GR + Compound**, page 84
6. n-Th232-fis: **Fission cross sections: n +  $^{232}\text{Th}$** , page 91
7. n-Th232-fis-wkb: **Fission cross sections with WKB method: n +  $^{232}\text{Th}$** , page 227
8. n-Si028-recoil-exact: **Recoils: 20 MeV n +  $^{28}\text{Si}$** , page 97
9. n-Tc099-ld1: **Different level density models : n +  $^{99}\text{Tc}$** , page 136
10. g-Zr090-xs: **Photonuclear reactions: g +  $^{90}\text{Zr}$** , page, page 144
11. n-Y089-dircap: **Direct neutron capture for n +  $^{89}\text{Y}$** , page 145
12. n-Sn120-omp-KD03: **Different optical models : n +  $^{120}\text{Sn}$** , page 158
13. a-Ho165-omp1: **Different alpha-particle optical model potentials: alpha +  $^{165}\text{Ho}$** , page 164
14. n-Nb093-adjust: **Local parameter adjustment for n +  $^{93}\text{Nb}$** , page 168

15. n-Si028-cc: **Coupled-channels rotational model:  $n + {}^{28}\text{Si}$** , page 172
16. n-Ge074-vib: **Coupled-channels vibrational model:  $n + {}^{74}\text{Ge}$** , page 173
17. n-Pb208-omp: **Spherical optical model and DWBA:  $n + {}^{208}\text{Pb}$** , page 175
18. n-Bi209-preeq: **Continuum spectra at 63 MeV for  $\text{Bi}(n, xp) \dots \text{Bi}(n, x\alpha)$** , page 199
19. p-Zr090-preeq: **Pre-equilibrium angular dist. and multiple pre-equilibrium emission**, page 201
20. p-Bi209-500MeV: **Calculations up to 500 MeV for  $p + {}^{209}\text{Bi}$** , page 203
21. n-Nb093-WFC-HF: **Comparison of compound nucleus WFC models: 10 keV  $n + {}^{93}\text{Nb}$** , page 214
22. n-U238-fy: **Fission yields for  ${}^{238}\text{U}$** , page 238
23. n-Pu242-fy: **Neutron multiplicities and fission yields for  $n + {}^{242}\text{Pu}$** , page 241
24. n-Pu239-fy-gef: **Fission yields and neutron observables  $n + {}^{239}\text{Pu}$** , page 244
25. n-Y089-RRR-standard: **Low energy resonance data for  $n + {}^{89}\text{Y}$** , page 250
26. n-Ba136-URR: **Unresolved resonance range parameters:  $n + {}^{136}\text{Ba}$** , page 252
27. n-Os187-astro-ng: **Astrophysical reaction rates :  $n + {}^{187}\text{Os}$** , page 258
28. n-Ba138-MACS: **Maxwellian averaged cross section at 30 keV:  $n + {}^{138}\text{Ba}$** , page 259
29. p-Mo100-medical: **Medical isotope production with  $p + {}^{100}\text{Mo}$** , page 268







# Part Three: Reference Guide

<b>21</b>	<b>Four main keywords</b> .....	<b>277</b>
<b>22</b>	<b>Keywords for basic and numerical parameters</b> .....	<b>285</b>
<b>23</b>	<b>Keywords for masses and deformations</b> 329	
<b>24</b>	<b>Keywords for discrete levels</b> .....	<b>337</b>
<b>25</b>	<b>Keywords for level densities</b> .....	<b>351</b>
<b>26</b>	<b>Keywords for gamma emission</b> .....	<b>403</b>
<b>27</b>	<b>Keywords for optical model</b> .....	<b>447</b>
<b>28</b>	<b>Keywords for direct reactions and ECIS</b>	<b>517</b>
<b>29</b>	<b>Keywords for compound nucleus</b> ....	<b>547</b>
<b>30</b>	<b>Keywords for pre-equilibrium reactions</b> 565	
<b>31</b>	<b>Keywords for fission</b> .....	<b>603</b>
<b>32</b>	<b>Keywords for astrophysics</b> .....	<b>641</b>
<b>33</b>	<b>Keywords for medical isotope production</b> 649	
<b>34</b>	<b>Keywords for output</b> .....	<b>661</b>
<b>35</b>	<b>Keywords of TALYS</b> .....	<b>687</b>



In this part, all TALYS keywords will be described, one per page. The description of each keyword is as follows:

- Name of the keyword
- Explanation
- Examples
- Range of allowed values
- Default value
- Comments (optional), when we feel that some extra warnings or explanation for proper use is appropriate.

In each chapter, the keywords are roughly ordered by importance, rather than alphabetically. This is rather subjective of course, but it may make this part a bit more readable. Hence, it could be that the keywords near the end of certain Chapters are seldom used. In principle, all keywords may be referred to in the other parts of this tutorial, often in the Sections called *Implementation*, *Options* and *Sample cases*.



## **21. Four main keywords**

**projectile**

Eight different symbols can be given as **projectile**, namely **n, p, d, t, h, a, g** representing neutron, proton, deuteron, triton,  $^3\text{He}$ , alpha and gamma, respectively, and **0**, which is used if instead of a nuclear reaction (projectile + target) we start with an initial population of an excited nucleus.

**Examples**

**projectile n**

**projectile d**

**projectile 0**

**Range**

**projectile** must be equal to **n, p, d, t, h, a, g** or **0**.

**Default**

None.

---

## element

Either the nuclear symbol or the charge number  $Z$  of the target nucleus can be given. Possible values for **element** range from Li (3) to C4 (124).

### Examples

**element pu**

**element 41**

**element V**

**element B9**

### Range

$3 \leq \text{element} \leq 124$  or  $\text{Li} \leq \text{element} \leq \text{C4}$ .

### Default

None.

### Comments

- To accommodate target nuclides with  $Z > 110$  the element names are defined as follows: Rg(111), Cn(112), Nh(113), Fl(114), Mc(115), Lv(116), Ts(117), Og(118), B9(119), C0-4(120-124), which takes as a basis for  $Z = 100$  the symbol  $A0$ , etc. unless an official name has been assigned to it, which is currently the case for  $Z \leq 118$ . Clearly, if we ever need or wish to go beyond  $Z = 124$ , there are enough symbols left. Obviously, the symbols for  $Z$  above 118 will be changed as soon as official names are assigned to them.

**mass**

The target mass number  $A$ . The case of a natural element can be specified by **mass 0**. Then, a TALYS calculation for each naturally occurring isotope will be performed (see also the **abundance** keyword, p. 320), after which the results will be properly weighted and summed.

**Examples**

**mass 239**

**mass 0**

**Range**

**mass 0** or  $5 < \text{mass} \leq 339$ . The extra condition is that the target nucleus, i.e. the combination of **mass** and **element** must be present in the mass database, which corresponds to all nuclei between the proton and neutron drip lines.

**Default**

None.



## energy

The incident energy in MeV. The user has four possibilities: (1) A single incident energy is specified in the input as a real number, (2) A filename is specified, where the corresponding file contains a series of incident energies, with one incident energy per line. Any name can be given to this file, provided it starts with a character, and it is present in your working directory. Option (2) is helpful for the calculation of excitation functions or for the creation of nuclear data libraries. Option (2) is mandatory if **projectile 0**, i.e., if instead of a nuclear reaction we start with a population of an excited nucleus (see the **Special cases** below). There are an additional 2 options which allows to use a whole range of incident energies without having to make a file for this: (3) A filename that is predefined by TALYS. This contains a hardwired energy grid with for example for neutrons a finer energy grid at low energies and a progressively wider grid at high energies. These have the form *pE1-E2.grid* where *p* is the name of the projectile, and *E1*, *E2* are the start and end energy in integer form respectively. The *.grid* specifies that this concerns a predefined energy grid. An often used example is *n0-200.grid* which is the neutron energy grid of the TENDL library. More energies are used in the eV and keV range while at high energies automatically a course grid is used. With the file *n0-20.grid* exactly the same incident energy grid is used, but only up to the final energy of 20 MeV. Finally, option (4) is to give after the *energy* keyword 3 numbers: the starting energy, the end energy and the energy step. Next, an equidistant grid is made based on these 3 numbers.

### Examples

**energy 140.**

**energy 0.002**

**energy range** (user-defined file *range* with energies)

**energy n0-30.grid** (hardwired energy grid for incident neutrons between 0 and 20 MeV)

**energy p10-100.grid** (hardwired energy grid for incident protons between 10 and 100 MeV)

**energy 0.5 20. 0.5** (Incident energies: 0.5, 1, 1.5, ...20 MeV)

### Range

$10^{-11}$  MeV  $\leq$  **energy** < 1000 MeV or a *filename*, whereby the corresponding file contains at least 1 and a maximum of **numenin** incident energies, where **numenin** is an array dimension specified in *talys.cmb*. Currently **numenin=600**.

### Default

None.

### Using the four main keywords

To summarize the use of the four basic keywords, consider the following input file

```
projectile n
element   pd
mass      110
energy range
```

The file *range* looks e.g. as follows

```
0.1
0.2
0.5
1.
1.5
2.
5.
8.
10.
15.
20.
```

In the source code, the number of incident energies, here 11, is known as **numinc**. For the four-line input given above, TALYS will simulate all open reaction channels for  $n + {}^{110}\text{Pd}$  for all incident energies given in the file *range*, using defaults for all models and parameters. The most important cross sections will automatically be written to the output file.

### Special cases

There are two examples for which the **energy** keyword does not represent the incident energy of the projectile.

#### 1. Populated initial nucleus

If **projectile 0**, the user *must* give a filename for the **energy** keyword. This time however, the file does not consist of incident energies but of the excitation energy grid of the initial population, see Section 17. On the first line of the file we read the number of energies (lines), number of spins, and number of parities. The excitation energies that are read represent the middle values of the bins, and are followed by either (a) if the number of spins is zero, the total population in that bin, or (b) if the number of spins is not zero, the population per excitation energy bin and spin, using one column per spin. The values given in the second column represent either a histogram, if **popMeV n** (the default) or spectrum (**popMeV y**). If **popMeV y**, these spectral values are multiplied by the energy bin width and we obtain the population, which has a dimension of millibarns. If the population for only 1 parity is given, the population is equally distributed over both parities. Hence, if we let TALYS determine the spin-parity distribution (case (a)), an example of such a file is

```
      8    0    1
0.25 0.1342
0.75 0.2176
1.25 0.3344
1.75 0.6522
2.25 0.6464
2.75 0.2897
```

3.25 0.1154  
3.75 0.0653

and if we give the full spin-dependent population (case (b)), with equal parity distribution, we could e.g. have

	8	3	1
0.25	0.0334	0.0542	0.0112
0.75	0.0698	0.1043	0.0441
1.25	0.1131	0.2303	0.0971
1.75	0.1578	0.3333	0.1143
2.25	0.1499	0.3290	0.1212
2.75	0.1003	0.2678	0.0845
3.25	0.0844	0.1313	0.0661
3.75	0.0211	0.0889	0.0021

If in addition the population should be parity-dependent, one would have e.g.

	8	3	2
0.25	0.0134	0.0542	0.0112
0.75	0.0298	0.1043	0.0441
1.25	0.0531	0.2303	0.0971
1.75	0.0578	0.3333	0.1143
2.25	0.0599	0.3290	0.1212
2.75	0.0603	0.2678	0.0845
3.25	0.0444	0.1313	0.0661
3.75	0.0111	0.0889	0.0021
0.25	0.0234	0.0142	0.0008
0.75	0.0498	0.0943	0.0141
1.25	0.0666	0.1103	0.0471
1.75	0.0678	0.1333	0.0343
2.25	0.0299	0.1290	0.0512
2.75	0.0403	0.1678	0.0245
3.25	0.0344	0.1013	0.0361
3.75	0.0100	0.0489	0.0011

Note that in this case the energy restarts in the middle; first the energy grid for 3 spin values and parity -1 is given, then for parity +1.

## 2. Astrophysical reaction rates

If **astro y**, (p. 642) for astrophysical reaction rate calculations, the incident energy as given in the input is overruled by a hardwired incident energy grid that is appropriate for the calculation of reaction rates. However, to avoid unnecessary programming complications, the **energy** keyword *must* still be given in the input, and it can have any value. Hence if **astro y** one could e.g. give **energy 1**. in the input file. The adopted incident energies that overrule this value will be given in the output.

### Four main keywords: summary

The first four keywords are clearly the most important: they do *not* have a default value while they determine default values for some of the other keywords which, in other words, may be projectile-, energy-, element- or mass-dependent. All the keywords that follow now in this manual have default values and can hypothetically be left out of the input file if you are only interested in a minimal reaction specification. If you want to add keywords, you can enter them one by one in the format

that will be described below. Another way is to go to the top of the output file that is generated by the simple input file given above. You will find all the keywords with their values as adopted in the calculation, either user-specified or as defaults. All nuclear model parameters per nucleus are printed in the file *parameters.dat*, provided **partable y** was set in the input. You can copy this part and paste it into the input file, after which the values can be changed.

## **22. Keywords for basic and numerical parameters**

**bins**

The number of excitation energy bins in which the continuum of the initial compound nucleus is divided for further decay. The excitation energy region between the last discrete level and the total excitation energy for the initial compound nucleus is divided into **bins** energy bins. The resulting bin width then also determines that for the neighboring residual nuclei, in the sense that for any residual nucleus we ensure that the bins fit exactly between the last discrete level and the maximal possible excitation energy. For residual nuclides far away from the target, a smaller number of bins is automatically adopted. If **bins 0** a simple formula is invoked to take into account the fact that more bins are needed when the incident energy increases. Instead of a constant number of bins, it is then incident energy ( $E$ ) dependent. More specifically, **nbins 0** means

$$\text{Nbins} = 30 + 50 \frac{E^2}{(E^2 + 60^2)} \quad (22.1)$$

It is obvious that **bins** has a large impact on the computation time.

**Examples**

**bins 25**

**Range**

**bins = 0** or  $2 \leq \text{bins} \leq \text{numbins}$ , where **numbins** is specified in the file *talys.cmb*. Currently, **numbins=100**

**Default**

**bins 40**

**Ltarget**

The excited state of the target as given in the discrete level database. This keyword allows to compute cross sections for isomeric targets.

**Examples**

**Ltarget 2**

**Range**

$0 \leq \text{Ltarget} \leq \text{numlev}$ , where **numlev** is specified in the file *talys.cmb*. Currently, **numlev=30**

**Default**

**Ltarget 0**, i.e. the target is in its ground state.

## maxZ

The maximal number of protons away from the initial compound nucleus that is considered in a chain of residual nuclides. For example, if **maxZ 3**, then for a  $n + {}^{56}\text{Fe}$  ( $Z=26$ ) reaction, the V-isotopes ( $Z=23$ ) are the last to be considered for further particle evaporation in the multiple emission chain. **maxZ** is normally only changed for diagnostic purposes. For example, if you are only interested in the  $(n, n')$ ,  $(n, 2n)$ , ...  $(n, xn)$  residual production cross sections, or the associated discrete gamma ray intensities, **maxZ 0** is appropriate. (Note that in this case, the competition of emission of protons up to alpha particles is *still* taken into account for all the nuclides along the  $(n, xn)$  chain, only the decay of the residual nuclides associated with this charged-particle emission are not tracked further).

## Examples

**maxZ 6**

## Range

$0 \leq \text{maxZ} \leq \text{numZ}-2$ , where **numZ=2+2\*memorypar** is specified in the file *talys.cmb*, where **memorypar=5** for a computer with at least 256 Mb of memory.

## Default

Continue until all possible reaction channels are closed or until the maximal possible value for **maxZ** is reached (which rarely occurs). By default **maxZ=numZ-2** where the parameter **numZ** is specified in the file *talys.cmb*. This parameter should be large enough to ensure complete evaporation of all daughter nuclei.



## maxN

The maximal number of neutrons away from the initial compound nucleus that is considered in a chain of nuclides. For example, if **maxN 3**, then for a  $n + {}^{56}\text{Fe}$  ( $N=31$ ) reaction, residual nuclei with  $N=28$  ( $=31-3$ ) are the last to be considered for further particle evaporation in the multiple emission chain. **maxN** is normally only changed for diagnostic or economical purposes.

## Examples

**maxN 6**

## Range

$0 \leq \text{maxN} \leq \text{numN}-2$ , where **numN=4+4\*memorypar** is specified in the file *talys.cmb*, where **memorypar=5** for a computer with at least 256 Mb of memory.

## Default

Continue until all possible reaction channels are closed or until the maximal possible value for **maxN** is reached (which rarely occurs). By default **maxN=numN-2** where the parameter **numN** is specified in the file *talys.cmb*. This parameter should be large enough to ensure complete evaporation of all daughter nuclei.

**Liso**

Level number of isomer. This user assigned value can overrule the one determined from the discrete level file.

**Examples**

**Liso 2**

**Range**

$0 \leq \mathbf{Liso} \leq \mathbf{numlev}$ , where **numlev** is specified in the file *talys.cmb*.

**Default**

**Liso** not given.

**isomer**

Flag to run TASMANT for covariances or optimization of isomeric production cross sections.

**Examples**

**#isomer n**

**#isomer y**

**Range**

**y or n**

**Default**

**#isomer n**

**channels**

Flag for the calculation and output of all exclusive reaction channel cross sections, e.g.  $(n, p)$ ,  $(n, 2n)$ ,  $(n, 2npa)$ , etc. The **channels** keyword can be used in combination with the keywords **outspectra** and **outangle** (see next Section) to give the exclusive spectra and angular distributions.

**Examples**

**channels y**

**channels n**

**Range**

y or n

**Default**

**channels n**

---

**partable**

Flag to write the all the model parameters used in a calculation on a separate file, *parameters.dat*. This can be a very powerful option when one wishes to vary any nuclear model parameter in the input. The file *parameters.dat* has the exact input format, so it can be easily copied and pasted into any input file. This is helpful for a quick look-up of all the parameters used in a calculation. We have used this ourselves for automatic (random) TALYS-input generators for e.g. covariance data.

**Examples**

**partable y**  
**partable n**

**Range**

y or n

**Default**

**partable n**

**best**

Flag to use the set of adjusted nuclear model parameters that produces the optimal fit for measurements of all reaction channels of the nuclide under consideration. TALYS will look in the *talys/structure/best* directory for such a set, see Section 5.4. With the addition of the single line **best y** to the input file, TALYS will automatically use all detailed adjusted parameter fits that are stored in *talys/structure/best*.

**Examples**

**best y**  
**best n**

**Range**

y or n

**Default**

**best n**

## **ejectiles**

The outgoing particles that are considered in competing reaction channels. By default, all competing channels are included even if one is interested in only one type of outgoing particle. This is necessary since there is always competition with other particles, in e.g. Hauser-Feshbach and pre-equilibrium models, that determines the outcome for the particle under study. Furthermore, reaction Q-values automatically keep channels closed at low incident energies. However, for diagnostic or time-economical purposes, or cases where e.g. one is only interested in high-energy ( $p, p'$ ) and ( $p, n$ ) multi-step direct reactions, one may save computing time and output by skipping certain ejectiles as competing particles. For neutron-induced reactions on actinides up to 20 MeV, setting **ejectiles g n** is a rather good approximation that saves time. Also comparisons with computer codes that do not include the whole range of particles will be facilitated by this keyword.

### **Examples**

**ejectiles n**

**ejectiles g n p a**

### **Range**

**ejectiles** can be any combination of **g, n, p, d, t, h** and **a**.

### **Default**

Include all possible outgoing particles, i.e. **ejectiles g n p d t h a**

### **Comments**

**transpower**

A limit for considering transmission coefficients in the calculation. Transmission coefficients  $T_{lj}$  smaller than  $T_{0\frac{1}{2}} \times 10^{-transpower} / (2l + 1)$  are not used in Hauser-Feshbach calculations, in order to reduce the computation time.

**Examples**

**transpower 12**

**Range**

**$2 \leq \text{transpower} \leq 20$**

**Default**

**transpower 5**



**transeps**

A limit for considering transmission coefficients in the calculation. Transmission coefficients smaller than **transeps** are not used in Hauser-Feshbach calculations, irrespective of the value of **transpower**, in order to reduce the computation time.

**Examples**

**transeps 1.e-12**

**Range**

**$0. \leq \text{transeps} \leq 1.$**

**Default**

**transeps 1.e-8**

**xseps**

The limit for considering cross sections in the calculation, in mb. Reaction cross sections smaller than **xseps** are not used in the calculations, in order to reduce the computation time.

**Examples**

**xseps 1.e-10**

**Range**

**0. ≤ xseps ≤ 1000.**

**Default**

**xseps 1.e-7**

---

## **popeps**

The limit for considering population cross sections in the multiple emission calculation, in mb. Nuclides which, before their decay, are populated with a total cross section less than **popeps** are skipped, in order to reduce the computation time. From **popeps**, also the criteria for continuation of the decay per excitation energy bin (variable **popepsA**) and per  $(E_x, J, \Pi)$  bin (variable **popepsB**) in Hauser-Feshbach calculations are automatically derived.

### **Examples**

**popeps 1.e-6**

### **Range**

**$0. \leq \text{popeps} \leq 1000.$**

### **Default**

**popeps 1.e-3**

**popMeV**

Flag to use initial population per MeV instead of per histogram, in the case of an excited energy distribution to be used with **projectile 0**. This keyword is given for backward compatibility. It is now seen that it is more natural to input the absolute value of the initial population per continuum bin or level, as histograms.

**Examples**

```
popMeV y  
popMeV n
```

**Range**

y or n

**Default**

```
popMeV n
```

**Estop**

Incident energy above which TALYS stops. This keyword is needed to speed up some uncertainty applications involving random TALYS calculations. No matter which incident energies are given in the input file, TA

**Examples**

**Estop 50.**

**Range**

**$1.e - 5 \leq \text{Estop} \leq 1000.$**

**Default**

**Estop 1000.**

**endf**

Flag for the creation of various output files needed for the assembling of an ENDF-6 formatted file. Apart from the creation of various files that will be discussed for the following keywords, a file *endf.tot* is created which contains the reaction, elastic and total cross sections (calculated by ECIS) on a fine energy grid. Also a file *decay.X* will be created, with X the ejectile symbol in (a1) format, that contains the discrete gamma decay probabilities for the binary residual nuclides. This will be used for gamma-ray production from discrete binary levels. In addition to all the output detailed below, the continuum gamma-ray spectra per residual nucleus and incident energy will be stored in files *gamZZZAAAEYYY.YYY.tot*, where where ZZZ is the charge number and AAA is the mass number in (i3.3) format, and YYY.YYY is the incident energy in (f7.3) format. Setting **endf y** will automatically enable many of the “**file**” keywords given below. All these specific files will be written to output if **endfdetail y**, see the next keyword.

**Examples**

```
endf y
endf n
```

**Range**

y or n

**Default**

endf n

---

## **endfdetail**

Flag to use specific MT numbers per reaction channel. With **endfdetail**, the corresponding value read from the TALYS output file *endf.inf* can be overruled from **y** to **n**, i.e. even if detailed reaction channel information was provided by TALYS, you may now choose to use a lumped presentation using MT5. Of course, if the corresponding value in *endf.inf* is **n**, then the detailed information was not produced by TALYS, and **endfdetail** can no longer be set to **y**. If you want that, a TALYS run with keyword **endfdetail y** (the default for incident neutrons), should be performed first.

### **Examples**

**endfdetail n**

**endfdetail y**

### **Range**

**y** or **n**

### **Default**

The value of **endfdetail** is adopted from the TALYS output file *tefal.inf*.

**angles**

Number of emission angles for reactions to discrete states.

**Examples**

**angles 18**

**Range**

$1 \leq \text{angles} \leq \text{numang}$ , where **numang** is specified in the file *talys.cmb*. Currently, **numang=90**

**Default**

**angles 90**



**anglescont**

Number of emission angles for reactions to the continuum.

**Examples**

**anglescont 18**

**Range**

$1 \leq \text{anglescont} \leq \text{numangcont}$ , where **numangcont** is specified in the file *talys.cmb*. Currently, **numangcont=36**

**Default**

**anglescont 36**

**maxchannel**

Maximal number of outgoing particles in exclusive channel description, e.g. if **maxchannel 3**, then reactions up to 3 outgoing particles, e.g.  $(n, 2np)$ , will be given in the output. **maxchannel** is only active if **channels y**. We emphasize that, irrespective of the value of **maxchannel** and **channels**, all reaction chains are, by default, followed until all possible reaction channels are closed to determine cumulative particle production cross sections and residual production cross sections.

**Examples**

**maxchannel 2**

**Range**

$0 \leq \text{maxchannel} \leq 8$ ,

**Default**

**maxchannel 4**

**relativistic**

Flag for relativistic kinematics.

**Examples**

**relativistic y**

**relativistic n**

**Range**

**y or n**

**Default**

**relativistic y**

**reaction**

Flag to disable nuclear reaction calculation. This may be helpful if one is e.g. only interested in a level density calculation. A TALYS run will then last only a few hundreds of a second. (This keyword was used for the large scale level density analysis of [20]).

**Examples**

**reaction y**

**reaction n**

**Range**

**y or n**

**Default**

**reaction y**

**recoil**

Flag to include recoil information.

**Examples**

**recoil n**  
**recoil y**

**Range**

**y** or **n**

**Default**

**recoil** is read from the TALYS output file *endf.inf*. In other words, if *recoil y* was set in the TALYS input file, then this will automatically be used in TEFAL.

**labddx**

Flag for the calculation of double-differential cross sections in the LAB system. This is only active if **recoil y**. If **labddx n**, only the recoils of the nuclides are computed.

**Examples**

**labddx y**  
**labddx n**

**Range**

**y** or **n**

**Default**

**labddx n**

---

**anglesrec**

Number of emission angles for recoiling nuclides. This is only active if **recoil y**.

**Examples**

**anglesrec 4**

**Range**

$1 \leq \text{anglesrec} \leq \text{numangrec}$ , where **numangrec** is specified in the file *talys.cmb*. Currently, **numangrec=9**

**Default**

**anglesrec 9** if **labddx y**, **anglesrec 1** if **labddx n**

**maxenrec**

Number of emission energies for recoiling nuclides. This is only active if **recoil y**.

**Examples**

**maxenrec 4**

**Range**

$1 \leq \text{maxenrec} \leq \text{numenrec}$ , where **numenrec** is specified in the file *talys.cmb*. Currently, **numenrec=5\*memorypar**, where we suggest **memorypar=5** for a computer with at least 256 Mb of memory.

**Default**

**maxenrec 10**



---

## **recoilaverage**

Flag to consider only one average kinetic energy of the recoiling nucleus per excitation energy bin (instead of a full kinetic energy distribution). This approximation significantly decreases the calculation time. This is only active if **recoil y**.

### **Examples**

**recoilaverage y**

**recoilaverage n**

### **Range**

**y or n**

### **Default**

**recoilaverage n**

**channelenergy**

Flag to use the channel energy instead of the center-of-mass energy for the emission spectrum.

**Examples**

`channelenergy y`

`channelenergy n`

**Range**

y or n

**Default**

`channelenergy n`

**equidistant**

Flag for adopting an equidistant or non-equidistant (logarithmic) excitation energy grid for the continuum of each residual nucleus. The latter option makes the calculation more efficient since it uses more precision when it is needed.

**Examples**

**equidistant y**

**equidistant n**

**Range**

**y or n**

**Default**

**equidistant y**

## segment

The number of segments to divide the standard emission energy grid. The basic emission energy grid we use for spectra and transmission coefficient calculations is:

```
0.001, 0.002, 0.005    MeV
0.01, 0.02, 0.05      MeV
0.1- 2 MeV : dE= 0.1 MeV
2 - 4 MeV : dE= 0.2 MeV
4 - 20 MeV : dE= 0.5 MeV
20 - 40 MeV : dE= 1.0 MeV
40 - 200 MeV : dE= 2.0 MeV
200 - 300 MeV : dE= 5.0 MeV
above 300 MeV : dE=10.0 MeV
```

This grid is divided into a finer grid by subdividing each interval by **segment**.

## Examples

**segment 3**

### Range

$1 \leq \text{segment} \leq 4$ . Extra conditions:  $1 \leq \text{segment} \leq 3$  if the maximum incident energy is larger than 20 MeV,  $1 \leq \text{segment} \leq 2$  if the maximum incident energy is larger than 40 MeV, **segment=1** if the maximum incident energy is larger than 100 MeV. (These rules are imposed due to memory limitations).

### Default

**segment 1**

**Elow**

Lowest incident energy in MeV for which TALYS performs a full nuclear model calculation. Below this energy, cross sections result from inter- and extrapolation using the calculated values at **Elow** and tabulated values and systematics at thermal energy, see Section 16. This keyword should only be used in the case of several incident energies.

**Examples**

**Elow 0.001**

**Range**

$1.e-6 \leq \text{Elow} \leq 1.$

**Default**

**Elow**= $D_0$  for data files (**endf y**), **1.e-6** otherwise.

**maxZrp**

The maximal number of protons away from the initial compound nucleus before new residual evaporation.

**Examples**

**maxZrp 6**

**Range**

$0 \leq \text{maxZ} \leq \text{numZ}-2$ , where  $\text{numZ}=2+2*\text{memorypar}$  is specified as global parameter.

**Default**

**maxZrp=numZ-2**

**maxNrp**

The maximal number of neutrons away from the initial compound nucleus before new residual evaporation.

**Examples**

**maxNrp 6**

**Range**

$0 \leq \text{maxN} \leq \text{numN}-2$ , where **numN=2+2\*memorypar** is specified as global parameter.

**Default**

**maxNrp=numN-2**

## abundance

File with tabulated abundances. The **abundance** keyword is only active for the case of a natural target, i.e. if **mass 0**. By default, the isotopic abundances are read from the structure database, see Chapter 5. It can however be imagined that one wants to include only the most abundant isotopes of an element, to save some computing time. Also, **abundance** may be used to analyze experimental data for targets of a certain isotopic enrichment. On the input line, we read **abundance** and the filename. From each line of the file, TALYS reads  $Z$ ,  $A$  and the isotopic abundance with the format (2i4,f11.6). An example of an abundance file, e.g. *abnew*, different from that of the database, is

```
82 206 24.100000
82 207 22.100000
82 208 52.400000
```

where we have left out the “unimportant”  $^{204}\text{Pb}$  (1.4%). TALYS automatically normalizes the abundances to a sum of 1, leading in the above case to 24.44 % of  $^{206}\text{Pb}$ , 22.41 % of  $^{207}\text{Pb}$  and 53.14 % of  $^{208}\text{Pb}$  in the actual calculation.

## Examples

**abundance abnew**

## Range

**abundance** can be equal to any filename, provided it is present in the working directory.

## Default

If **abundance** is not given in the input file, abundances are taken from *talys/structure/abundance* and calculations for all isotopes are performed.



**micro**

Flag to use microscopic models for all nuclear structure ingredients.

**Examples**

**micro y**  
**micro n**

**Range**

**y or n**

**Default**

**micro n**

**ompenenergyfile**

File with incident energies to calculate the total, elastic and reaction cross section on a sufficiently precise energy grid for ENDF-6 file purposes. On the input line, we read **ompenenergyfile**, filename.

**Examples**

**ompenenergyfile energies.omp**

**Range**

**ompenenergyfile** can be equal to any filename, provided it starts with a character. The incident energies should be in the range  $10^{-11}$  MeV to  $< 250$  MeV, whereby the **ompenenergyfile** contains at least 1 and a maximum of **numen6** incident energies, where **numen6** is an array dimension specified in *talys.cmb*. Currently **numen6=10000**.

**Default**

**ompenenergyfile** is not given in the input file,

**bestend**

Flag to put the best set of parameters at end of input file, instead of at the beginning. With **bestend n**, the parameters can be overruled later in the input file, while this is no longer possible with **bestend y**. Obviously, this keyword is only relevant if **best y**.

**Examples**

**bestend y**

**bestend n**

**Range**

y or n

**Default**

**bestend n**

## **bestpath**

Keyword to enable to use your own collection of “best” TALYS input parameters. This directory should be created under *talys/structure/* after which the current keyword can be used with that directory name. In practice, the user can copy a file from the standard *talys/structure/best/* database to his/her own database and change it. In this way, changes in the “official” database are avoided. See Section 5.4 for the format

### **Examples**

#### **bestpath mybest**

For this keyword to work, information needs to be present in the *talys/structure/mybest/* with a format equal to that of the original *talys/structure/best/* directory, see Section 5.4.

### **Range**

**bestpath** can be equal to any filename, provided it starts with a character. Furthermore, the library must exist in the *talys/structure/* directory. This keyword is only active if **best y**.

### **Default**

**bestpath** is equal to *talys/structure/best/*.

## rescuefile

File with incident energy dependent normalization factors. If TALYS is not able to produce the “ultimate fit”, it is possible to invoke a “rescuefile” as a last resort. By giving a table of incident energies and normalization factors, the result of TALYS can be made exactly equal to that of experimental data, an evaluate data set, or any other data set. The contents of the file consist of a simple x-y table with x the incident energy and y the normalization factor. A rescuefile can be used for the reactions (n,n’), (n, $\gamma$ ), (n,f), (n,p), (n,d), (n,t), (n,h), (n,’alpha ), (n,2n) and (n,total). To invoke this, the ENDF-6 format for MT numbers representing a reaction channel is used, see the sample cases below. Moreover, a global multiplication factor can be applied. On the input line, we read **rescuefile**, reaction identifier (MT number), global multiplication factor (optional).

### Examples

```

rescuefile 0 rescue.001 1.01 (n,total)
rescuefile 4 rescue.004 (n,n’)
rescuefile 16 rescue.016 (n,2n)
rescuefile 18 rescue.018 0.995 (n,f)
rescuefile 102 rescue.102 (n, $\gamma$ )
rescuefile 103 rescue.103 1.02 (n,p)
rescuefile 104 rescue.104 (n,d)
rescuefile 105 rescue.105 (n,t)
rescuefile 106 rescue.106 (n,h)
rescuefile 107 rescue.107 (n, $\alpha$ )

```

### Range

**rescuefile** can be equal to any filename, provided it starts with a character.

### Default

no default.

**nulldev**

Path for the null device. The null device is a "black hole" for output that is produced, but not of interest to the user. Some ECIS output files fall in this category. To ensure compatibility with Unix, Linux, Windows and other systems a null device string is used, of which the default setting is given in *machine.f*. With this keyword, extra flexibility is added. If the null device is properly set in *machine.f*, this keyword is not needed. On the input line, we read **nulldev**, filename.

**Examples**

```
nulldev /dev/null  
nulldev nul
```

**Range**

**nulldev** can be equal to any appropriate filename, provided it starts with a character.

**Default**

The default is set in subroutine *machine.f*.

## **strucpath**

Path for the directory with nuclear structure information. With this keyword, extra flexibility is added. Nuclear structure databases other than the default can be invoked with this keyword. If the path name is properly set in *machine.f*, this keyword is not needed for standard use. On the input line, we read **strucpath**, filename.

### **Examples**

**strucpath /home/raynal/mon-structure/**

### **Range**

**strucpath** can be equal to any appropriate directory. The maximum length of the path is 60 characters.

### **Default**

The default is set in subroutine *machine.f*.





## **23. Keywords for masses and deformations**

**massmodel**

Model for nuclear masses. There are 4 theoretical mass models and an analytic formula, see Section 6. They are only used when no experimental mass is available or when **expmass n**.

**Examples**

**massmodel 0:** Duflo-Zuker formula

**massmodel 1:** Möller table

**massmodel 2:** Goriely HFB-Skyrme table

**massmodel 3:** HFB-Gogny D1M table

**Range**

$0 \leq \text{massmodel} \leq 3$

**Default**

**massmodel 2**

**expmass**

Flag for using the experimental nuclear mass when available. Use **expmass n** to overrule the experimental mass by the theoretical nuclear mass (i.e. the Audi-Wapstra values will not be used). This will then be done for all nuclides encountered in the calculation.

**Examples**

**expmass y**

**expmass n**

**Range**

**y** or **n**

**Default**

**expmass y**

**beta2**

Deformation parameter for moment of inertia for the ground state or fission barrier, see Eq. (8.86). On the input line, we read **beta2**, Z, A, value, fission barrier. If the number of the fission barrier is not given or is equal to 0, it concerns the ground state.

**Examples**

**beta2 90 232 0.3**

**beta2 94 239 1.1 2**

**Range**

$0. \leq \text{beta2} < 1.5$

**Default**

**beta2 Z A 0.** for the ground state, **beta2 Z A 0.6** for the first barrier, **beta2 0.8** for the second barrier, and **beta2 Z A 1.** for the third barrier, if present.

**massdir**

User made directory with nuclear masses. With this keyword, extra flexibility for nuclear masses is added. The directory should be placed under talys/structure/masses, and the individual files should have the same structure as the other mass tables. On the input line, we read **massdir**, directory name.

**Examples**

**massdir hfb69**

**Range**

**massdir** can be equal to any appropriate directory name. The maximum length of the directory name is 72 characters.

**Default**

**massdir** is not used, i.e. the masses are read from an existing mass table.

**massnucleus**

The mass of the nucleus in amu. Use **massnucleus** to overrule the value given in the mass table. On the input line, we read **massnucleus**,  $Z$ ,  $A$ , value.

**Examples**

**massnucleus 41 93 92.12345**

**massnucleus 94 239 239.10101**

**Range**

$A - 1 \leq \text{massnucleus} \leq A + 1$ , where  $A$  is the mass number.

**Default**

**massnucleus 0.**, i.e. the nuclear mass is read from the mass table.

---

**massexcess**

The mass excess of the nucleus in MeV. Use **massexcess** to overrule the value given in the mass table. On the input line, we read **massexcess**, *Z*, *A*, value.

**Examples**

**massexcess 41 93 -45.678**

**massexcess 94 239 39.98765**

**Range**

**-500. ≤ massexcess ≤ 500.,**

**Default**

**massexcess 0.,** i.e. the mass excess is read from the mass table.





## **24. Keywords for discrete levels**

**maxlevelstar**

The number of included discrete levels for the *target* nucleus that is considered in Hauser-Feshbach decay and the gamma-ray cascade.

**Examples**

**maxlevelstar 0**  
**maxlevelstar 12**

**Range**

$0 \leq \text{maxlevelstar} \leq \text{numlev}$ , where **numlev** is specified in a module. Currently, **numlev=30**

**Default**

**maxlevelstar 30**

## maxlevelsbin

The number of included discrete levels for the nuclides resulting from *binary* emission that is considered in Hauser-Feshbach decay and the gamma-ray cascade. On the input line we read **maxlevelsbin**, the symbol of the ejectile and the number of levels of the associated residual nucleus.

### Examples

```
maxlevelsbin a 8  
maxlevelsbin p 12
```

### Range

$0 \leq \text{maxlevelsbin} \leq \text{numlev}$ , where **numlev** is specified in a module. Currently, **numlev=30**

### Default

**maxlevelsbin g 10, maxlevelsbin n 10, maxlevelsbin p 10, maxlevelsbin d 5, maxlevelsbin t 5, maxlevelsbin h 5, maxlevelsbin a 10.**

### Comments

- The value for the default of the inelastic channel will always be overruled by the **maxlevelstar** keyword, for the target nucleus, or the value for its default.

**maxlevelsres**

The number of included discrete levels for all *residual* nuclides that is considered in Hauser-Feshbach decay and the gamma-ray cascade. This keyword is overruled by **maxlevelsbin** and **maxlevelstar** for specified nuclides.

**Examples**

**maxlevelsres 0**

**maxlevelsres 12**

**Range**

$0 \leq \text{maxlevelsres} \leq \text{numlev}$ , where **numlev** is specified in a module. Currently, **numlev=30**

**Default**

**maxlevelsres 10**

**Nlevels**

The number of included discrete levels for a *specific residual* nucleus that is considered in Hauser-Feshbach decay and the gamma-ray cascade. On the input line we read **Nlevels**, *Z*, *A*, and the number of levels.

**Examples**

**Nlevels 41 93 8**

**Range**

$0 \leq \mathbf{Nlevels} \leq \mathbf{numlev}$ , where **numlev** is specified in a module. Currently, **numlev=30**

**Default**

**Nlevels** has the value specified by the defaults of **maxlevelstar**, **maxlevelsbin**, and **maxlevelsres**.

**disctable**

Keyword to change the used discrete level database. The standard discrete level database from RIPL can be updated with theoretical levels calculated from the microscopic level densities from **ldmodel 5**, for nuclides with no or few experimental levels. Note that **disctable 3** gives the option to perform nuclear reaction calculations using purely theoretically estimated levels.

**Examples**

**disctable 1:** RIPL database with experimental levels + theoretical levels added until the 100th level

**disctable 2:** RIPL database with experimental levels only (the only option until TALYS-1.4)

**disctable 3:** Theoretical levels for all nuclides

**Range**

$1 \leq \text{disctable} \leq 3$

**Default**

**disctable 1**

---

## levelfile

File with discrete levels. The format of the file is exactly the same as that of the nuclear structure database *talys/structure/levels/*. In practice, the user can copy a file from this database to the working directory and change it. In this way, changes in the “official” database are avoided. Note that even if only changes for one isotope are required, the entire file needs to be copied if for the other isotopes the originally tabulated values are to be used. On the input line, we read **levelfile**, *Z*, filename.

## Examples

**levelfile 26 Fe.loc**

## Range

**levelfile** can be equal to any filename, provided it starts with a character.

## Default

If **levelfile** is not given in the input file, the discrete levels are taken from the *talys/structure/levels* database per nucleus.

## branch

Branching ratio for discrete level to overrule that given by the discrete level file. With this keyword one can change for example the "guessed" branching ratios put in the discrete level file, or even overwrite those which are supposed to be experimentally known. On the input line we read **branch**, Z, A, mother level, the number of branching levels Nbr, and next the daughter level and branching ratio for each level up to Nbr.

## Examples

**branch 38 85 7 2 2 0.7 0 0.3:** 70% decay of 7th level of  $^{85}\text{Sr}$  to level 2, and 30% to the ground state

**branch 38 87 17 1 1 1.:** 100% decay of 17th level of  $^{87}\text{Sr}$  to level 1.

## Range

$0 \leq \text{number of levels, mother level, daughter level} \leq \text{numlev}$ , where **numlev** is specified in a module.

## Default

**branch** is not given.

## Comment

If branching ratios are considered to permanently overrule those of the discrete level file, the keywords as given above can be stored in the talys/structure/levels/branch directory, after which a simple **bestbranch y** (p. 346) is enough to invoke all these adjusted branching ratios.



**electronconv**

Flag for the application of an electron-conversion coefficient on the gamma-ray branching ratios from the discrete level file.

**Examples**

**electronconv y**

**electronconv n**

**Range**

**y or n**

**Default**

**electronconv y**

**bestbranch**

Flag to use the set of adjusted branching ratios for discrete levels.

**Examples**

**bestbranch y**  
**bestbranch n**

**Range**

**y** or **n**

**Default**

**bestbranch y**

**Comment**

If branching ratios are considered to permanently overrule those of the discrete level file, the keywords as given by **branch** (p. 344) can be stored in the talys/structure/levels/branch directory, after which a simple **bestbranch y** is enough to invoke all these adjusted branching ratios.

**outlevels**

Flag for the output of discrete level information for each nucleus. All level energies, spins parities, branching ratios and lifetimes will be printed.

**Examples**

**outlevels y**

**outlevels n**

**Range**

y or n

**Default**

the same value as **outbasic**: **outlevels n**

## ngfit

Flag to use the set of adjusted nuclear model parameters that produces the optimal fit for  $(n,\gamma)$  cross sections in the fast neutron range. TALYS will look in the *talys/structure/gamma* directory for files ending in *.wtable* to have the optimal **wtable** parameter, obtained by automated adjustment for each level density model. With the addition of the single line **ngfit y** to the input file, TALYS will automatically use all detailed adjusted parameter fits that are stored in *talys/structure/gamma*.

### Examples

**ngfit y**  
**ngfit n**

### Range

y or n

### Default

**ngfit n**, and **ngfit y** if **best y**

---

**nnfit**

Flag to use the set of adjusted nuclear model parameters that produces the optimal fit for (n,n'), (n,2n) and (n,p) cross sections. TALYS will look in the *talys/structure/best* directory for the *nn.par* file to have the optimal parameters, obtained by automated adjustment. With the addition of the single line **nnfit y** to the input file, TALYS will automatically use all detailed adjusted parameter fits that are stored in *talys/structure/best/nn.par*.

**Examples**

```
nnfit y
nnfit n
```

**Range**

y or n

**Default**

**nnfit n**, and **nnfit y** if **best y**

**nafit**

Flag to use the set of adjusted nuclear model parameters that produces the optimal fit for  $(n, \alpha)$  cross sections. TALYS will look in the *talys/structure/best* directory for the *na.par* file to have the optimal parameters, obtained by automated adjustment. With the addition of the single line **nafit y** to the input file, TALYS will automatically use all detailed adjusted parameter fits that are stored in *talys/structure/best/na.par*.

**Examples**

**nafit y**  
**nafit n**

**Range**

y or n

**Default**

**nafit n**, and **nafit y** if **best y**

## **25. Keywords for level densities**

## ldmodel

Model for level densities. There are 3 phenomenological level density models and 3 options for microscopic level densities, all described in Chapter 8. Since TALYS-1.4, it is possible to choose a level density model per nuclide considered in the reaction.

### Examples

**ldmodel 1:** Constant Temperature + Fermi gas model (CTM)

**ldmodel 2:** Back-shifted Fermi gas Model (BFM)

**ldmodel 3:** Generalised Superfluid Model (GSM)

**ldmodel 4:** Skyrme-Hartree-Fock-Bogoluybov level densities from numerical tables

**ldmodel 5:** Gogny-Hartree-Fock-Bogoluybov level densities from numerical tables

**ldmodel 6:** Temperature-dependent Gogny-Hartree-Fock-Bogoluybov level densities from numerical tables

**ldmodel 2 41 93:** Back-shifted Fermi gas Model just for this particular nucleus

### Range

$1 \leq \text{ldmodel} \leq 6$

### Default

ldmodel 1

### Comments

- Note the available flexibility exemplified by the last sample case above. It is possible to use different level density models per nuclide in one and the same TALYS run. Hence, one can e.g. globally use **ldmodel 5** but deviate from that for several nuclides by giving input lines like the **ldmodel 2 41 93** given above in the same input file.



---

## **ldmodelCN**

Model for level density for the compound nucleus

### **Examples**

**ldmodelCN 1:** Constant Temperature + Fermi gas model (CTM)

**ldmodelCN 2:** Back-shifted Fermi gas Model (BFM)

etc. see definition of **ldmodel**

### **Range**

$1 \leq \text{ldmodelCN} \leq 6$

### **Default**

**ldmodelCN = ldmodel**

### **Comments**

**a**

The level density parameter **a** at the neutron separation energy in  $\text{MeV}^{-1}$ . On the input line, we read **a**, *Z*, *A*, value.

**Examples**

**a 41 93 11.220**

**a 94 239 28.385**

**Range**

**$1. \leq a \leq 100.$**

**Default**

**a** is read from the *talys/structure/density/* directory or, if not present, is calculated from systematics, see Eq. (8.12).

## **aadjust**

Multiplier to adjust the level density parameter  $a$ . On the input line, we read **aadjust**,  $Z$ ,  $A$  and value.

### **Examples**

```
aadjust 41 93 1.04
```

### **Range**

$0.5 \leq \text{aadjust} \leq 2$ .

### **Default**

**aadjust**  $Z A 1$ .

### **Comments**

- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

**alimit**

The asymptotic level density parameter  $\tilde{a}$ , for a particular nucleus, in  $\text{MeV}^{-1}$ , see Eq. (8.12). On the input line, we read **alimit**,  $Z$ ,  $A$ , value.

**Examples**

**alimit 41 93 10.8**

**alimit 94 239 28.010**

**Range**

**1.  $\leq$  alimit  $\leq$  100.**

**Default**

**alimit** is determined from the systematics given by Eq. (8.13).

**asys**

Flag to use all level density parameters from systematics by default, i.e. to neglect the connection between  $a$  and the discrete level scheme and  $D_0$ , even if experimental values is available for the latter.

**Examples**

`asys y`

**Range**

y or n

**Default**

`asys n`

**deltaW**

Shell correction of the mass in MeV, see Eq. (8.12). On the input line, we read **deltaW**,  $Z$ ,  $A$ , value, the ground state or fission barrier to which it applies (optional). If the fission barrier is not given or is equal to 0, TALYS assumes it concerns the ground state of the nucleus.

**Examples**

**deltaW 41 93 0.110**

**deltaW 94 239 -0.262 1**

**Range**

$-20. \leq \text{deltaW} \leq 20.$

**Default**

**deltaW** is determined from Eq. (8.15).

**pair**

The pairing correction in MeV. On the input line, we read **pair**, Z, A, value.

**Examples**

**pair 94 239 0.76**

**Range**

$0. \leq \text{pair} \leq 10.$

**Default**

**pair** is determined from Eq. (8.49).

**ctable**

Constant  $c$  of the adjustment function (8.94) for tabulated level densities, per nucleus. On the input line, we read **ctable**,  $Z$ ,  $A$ , value, and (optionally) fission barrier.

**Examples**

**ctable 29 65 2.8**

**ctable 92 234 0.1 1**

**Range**

**-10. ≤ ctable ≤ 10.**

**Default**

**ctable Z A 0.**



---

**ctableadjust**

Adjustable shift for *ctable* of the adjustment function (8.94) for tabulated level densities, per nucleus. On the input line, we read **ctableadjust**, *Z*, *A*, value, and (optionally) fission barrier.

**Examples**

```
ctableadjust 29 65 2.8
ctableadjust 92 234 0.1 1
```

**Range**

**-10. ≤ ctableadjust ≤ 10.**

**Default**

**ctableadjust *Z A 0.***

**ptable**

Constant  $\delta$  of the adjustment function (8.94) for tabulated level densities, per nucleus. On the input line, we read **ptable**,  $Z$ ,  $A$ , value, and (optionally) fission barrier.

**Examples**

**ptable 29 65 -0.6**

**ptable 92 237 0.2 1**

**Range**

**-10.  $\leq$  ptable  $\leq$  10.**

**Default**

**ptable  $Z A 0$ .**

---

**ptheadjust**

Adjustable shift for *pthead* of the adjustment function (8.94) for tabulated level densities, per nucleus. On the input line, we read **ptheadjust**, *Z*, *A*, value, and (optionally) fission barrier.

**Examples**

```
ptheadjust 29 65 2.8
```

```
ptheadjust 92 234 0.1 1
```

**Range**

$-10. \leq \text{ptheadjust} \leq 10.$

**Default**

**ptheadjust** *Z A 0.*

**Nlow**

Lower level to be used in the temperature matching problem of the Constant Temperature formula. On the input line, we read **Nlow**,  $Z$ ,  $A$ , value, the ground state or fission barrier to which it applies (optional). If the fission barrier is not given or equal to 0, it concerns the ground state of the nucleus.

**Examples**

**Nlow 41 93 4**

**Nlow 94 239 2 1**

**Range**

$0 \leq \text{Nlow} \leq 200$

**Default**

**Nlow Z A 2**

## **Ntop**

Upper level to be used in the temperature matching problem of the Constant Temperature formula. On the input line, we read **Ntop**, *Z*, *A*, value, the ground state or fission barrier to which it applies (optional). If the fission barrier is not given or equal to 0, it concerns the ground state of the nucleus.

## **Examples**

**Ntop 41 93 14**

**Ntop 94 239 20 1**

## **Range**

$0 \leq \mathbf{Ntop} \leq 200$ , and  $\mathbf{Ntop} \geq \mathbf{Nlow}$ .

## **Default**

**Ntop** is read from the various subdirectories of *talys/structure/density/ground/*. If not present there, **Ntop** is equal to the last discrete level used for the Hauser-Feshbach calculation.

### Exmatch

The matching energy between the constant temperature and Fermi gas region in MeV, see Eq. (8.35). On the input line, we read **Exmatch**,  $Z$ ,  $A$ , value, the ground state or fission barrier to which it applies (optional). If the fission barrier is not given or is equal to 0, TALYS assumes it concerns the ground state of the nucleus.

### Examples

**Exmatch 41 93 4.213**

**Exmatch 94 239 5.556 2**

### Range

$0.1 \leq \text{Exmatch} \leq 20$ .

### Default

**Exmatch** is determined from Eq. (8.42).

## Exmatchadjust

Normalisation factor for the matching energy of the four-component formula. This parameter can be used as a relative normalisation instead of the absolute value of **Exmatchadjust**. On the input line, we read **Exmatchadjust**, Z, A, value, the ground state or fission barrier to which it applies (optional). If the fission barrier is not given or is equal to 0, TALYS assumes it concerns the ground state

### Examples

```
Exmatchadjust 41 93 0.9
```

```
Exmatchadjust 94 239 -1.13 1
```

### Range

$0.1 \leq \text{Exmatchadjust} \leq 10$ .

### Default

**Exmatchadjust** Z A 1..

- As a general comment for 'adjust' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

**T**

The temperature of the Gilbert-Cameron formula in MeV, see Eq. (8.39). On the input line, we read **T**, *Z*, *A*, value, the ground state or fission barrier to which it applies (optional). If the fission barrier is not given or is equal to 0, TALYS assumes it concerns the ground state of the nucleus.

**Examples**

**T 41 93 0.332**

**T 94 239 0.673 1**

**Range**

$0.001 \leq T \leq 10$ .

**Default**

**T** is determined from Eq. (8.39).

- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.



## Tadjust

Normalisation factor for the temperature of the four-component formula. This parameter can be used as a relative normalisation instead of the absolute value of **T**. On the input line, we read **Tadjust**, Z, A, value, the ground state or fission barrier to which it applies (optional). If the fission barrier is not given or is equal to 0, TALYS assumes it concerns the ground state

## Examples

```
Tadjust 41 93 0.9
```

```
Tadjust 94 239 -1.13 1
```

## Range

$0.1 \leq \text{Tadjust} \leq 10$ .

## Default

**Tadjust** Z A 1..

- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

**E0**

The "back-shift" energy of the four-component formula in MeV, see Eq. (8.36). On the input line, we read **E0**, *Z*, *A*, value, the ground state or fission barrier to which it applies (optional). If the fission barrier is not given or is equal to 0, TALYS assumes it concerns the ground state of the nucleus.

**Examples**

**E0 41 93 0.101**

**E0 94 239 -0.451 1**

**Range**

$-10. \leq E0 \leq 10.$

**Default**

**E0** is determined from Eq. (8.36).

- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

## **E0adjust**

Normalisation factor for the "back-shift" energy of the four-component formula. This parameter can be used as a relative normalisation instead of the absolute value of **E0**. On the input line, we read **E0adjust**, Z, A, value, the ground state or fission barrier to which it applies (optional). If the fission barrier is not given or is equal to 0, it concerns the ground state

## **Examples**

**E0adjust 41 93 0.9**

**E0adjust 94 239 -1.13 1**

## **Range**

**$0.1 \leq E0adjust \leq 10$ .**

## **Default**

**E0adjust Z A 1..**

- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

**Pshift**

An extra pairing shift for adjustment of the Fermi Gas level density, in MeV. On the input line, we read **Pshift**, *Z*, *A*, value.

**Examples**

**Pshift 60 142 0.26**

**Range**

$-10. \leq \text{Pshift} \leq 10.$

**Default**

**Pshift** is determined from Eqs. (8.31), (8.49), or (8.63), depending on the level density model.

---

## **Pshiftadjust**

Adjustable pairing shift for adjustment of the Fermi Gas level density, in MeV. The difference with **Pshift** is that the default is 0. On the input line, we read **Pshiftadjust**, Z, A, value.

### **Examples**

**Pshiftadjust 60 142 -0.13**

### **Range**

$-10. \leq \text{Pshift} \leq 10.$

### **Default**

**Pshift 0.**

- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply shifts the nominal value with the value given on this keyword line without having to know the actual value.

**Pshiftconstant**

Global constant for the adjustable pairing shift in MeV.

**Examples**

**Pshiftconstant 1.03**

**Range**

$-5. \leq \text{Pshiftconstant} \leq 5.$

**Default**

**Pshiftconstant=1.09** for **ldmodel 3** and **Pshiftconstant=0.** otherwise.

**Rspincut**

Global adjustable constant for spin cut-off parameter. Eq. (8.25) is multiplied by **Rspincut** for all nuclides in the calculation.

**Examples**

**Rspincut 0.8**

**Range**

**$0. \leq \text{Rspincut} \leq 10.$**

**Default**

**Rspincut 1.**

**Rspincutff**

Global adjustable constant for spin cut-off parameter for fission fragments

**Examples**

**Rspincutff 8.**

**Range**

**$0. \leq \text{Rspincutff} \leq 20.$**

**Default**

**Rspincutff 9.**



## **s2adjust**

Adjustable constant for spin cut-off parameter per nuclide. Eq. (8.25) is multiplied by **s2adjust**. On the input line, we read **s2adjust**, Z, A, value, fission barrier. If the number of the fission barrier is not given or is equal to 0, TALYS assumes it concerns the ground state.

### **Examples**

```
s2adjust 41 93 0.8  
s2adjust 94 239 1.1 2
```

### **Range**

$0.01 \leq \text{s2adjust} \leq 10$ .

### **Default**

**s2adjust Z A 1.**

- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

**filedensity**

Flag to write the level density and associated parameters on a separate file *ldZZZAAA.tot*, where *ZZZ* is the charge number and *AAA* is the mass number in (i3.3) format. **filedensity** is only active if **outdensity y**.

**Examples**

**filedensity y**

**filedensity n**

**Range**

**y** or **n**

**Default**

**filedensity n**

**outdensity**

Flag for the output of level density parameters and level densities for each residual nucleus.

**Examples**

**outdensity y**  
**outdensity n**

**Range**

**y** or **n**

**Default**

The same value as **outbasic**: **outdensity n**

## colenhance

Flag to enable or disable explicit collective enhancement of the level density, using the  $K_{rot}$  and  $K_{vib}$  factors. This keyword can be used in combination with the **ldmodel** keyword. For fission, if **colenhance n**, collective effects for the ground state are included implicitly in the intrinsic level density, and collective effects on the barrier are determined relative to the ground state. It is possible to enable or disable the collective enhancement per nuclide considered in the reaction.

### Examples

**colenhance y**

**colenhance y 41 93**

### Range

y or n

### Default

**colenhance y** if **fission y**, **colenhance n** otherwise.

---

## **colldamp**

Flag for damping of collective effects in effective level density, i.e. in a formulation without explicit collective enhancement. For fission, collective effects for the ground state are included implicitly in the intrinsic level density, and collective effects on the barrier are determined relative to the ground state, see Section 15.1.2.

### **Examples**

**colldamp y**

### **Range**

**y** or **n**

### **Default**

**colldamp n**

- In practice, this option has been used for fission model parameterization as employed in some of the older Bruyeres-le-Chatel actinide evaluations by Pascal Romain. Using it affects also the spin cutoff parameterization.

**spincutmodel**

Model for spin cut-off parameter for the ground state level densities, see Section 8.1.3. There are 2 expressions.

**Examples**

**spincutmodel 1:**  $\sigma^2 = c \frac{a}{\bar{a}} \sqrt{\frac{U}{a}}$

**spincutmodel 2:**  $\sigma^2 = c \sqrt{\frac{U}{a}}$

**Range**

$$1 \leq \text{spincutmodel} \leq 2$$

**Default**

**spincutmodel 1**

**shellmodel**

Model for liquid drop expression for nuclear mass, to be used to calculate the shell correction. There are 2 expressions.

**Examples**

**shellmodel 1:** Myers-Swiatecki

**shellmodel 2:** Goriely

**Range**

**$1 \leq \text{shellmodel} \leq 2$**

**Default**

**shellmodel 1**

**kvibmodel**

Model for the vibrational enhancement of the level density. There are 2 expressions.

**Examples**

**kvibmodel 1:** Eq. (8.84)

**kvibmodel 2:** Eq. (8.78)

**Range**

$1 \leq \text{kvibmodel} \leq 2$

**Default**

**kvibmodel 2**



---

## **ctmglobal**

Flag to enforce global formulae for the Constant Temperature Model (CTM), see Section 8.1.4. By default, if enough discrete levels are available the CTM will always make use of them for the estimation of the level density at low energies. For an honest comparison with other level density models, and also to test the predictive power for nuclides for which no discrete levels are known, we have included the possibility to perform a level density calculation with a truly global CTM. In practice it means that the matching energy  $E_M$  is always determined from the empirical formula (8.45), while  $T$  and  $E_0$  are determined from  $E_M$  through Eqs.(8.39) and (8.36), respectively. This flag is only relevant if **ldmodel 1**.

### **Examples**

**ctmglobal y**

### **Range**

y or n

### **Default**

**ctmglobal n**

**parity**

Flag to enable or disable non-equiparity level densities. At present, this option only serves to average tabulated parity-dependent level densities (**ldmodel 5, 6**) over the two parities (using **parity n**), for comparison purposes.

**Examples**

**parity y**

**Range**

y or n

**Default**

**parity n** for **ldmodel 1-4**, **parity y** for **ldmodel 5, 6**.

**alphald**

Constant for the global expression for the asymptotic level density parameter  $\tilde{a}$ , see Eq. (8.13).

**Examples**

**alphald 0.054**

**Range**

**$0.01 \leq \text{alphald} \leq 0.2$**

**Default**

**alphald** is determined from the systematics given by Table 8.1, depending on the used level density model.

**betald**

Constant for the global expression for the asymptotic level density parameter  $\tilde{a}$ , see Eq. (8.13).

**Examples**

**betald 0.15**

**Range**

$-0.5 \leq \text{betald} \leq 0.5$  with the extra condition that if **betald** < **0**, then **abs(betald)** < **alphald** (to avoid negative **a** values).

**Default**

**betald** is determined from the systematics given by Table 8.1, depending on the used level density model.

---

**gammald**

The damping parameter for shell effects in the level density parameter, for a particular nucleus, in  $\text{MeV}^{-1}$ , see Eq. (8.12). On the input line, we read **gammald**,  $Z$ ,  $A$ , value.

**Examples**

**gammald 41 93 0.051**

**Range**

$0. \leq \text{gammald} \leq 1.$

**Default**

**gammald** is determined from either Eq. (8.14) or (8.20).

**gammashell1**

Constant for the global expression for the damping parameter for shell effects in the level density parameter  $\gamma$ , see Eq. (8.14).

**Examples**

**gammashell1 0.5**

**gammashell1 0.**

**Range**

**$0. \leq \text{gammashell1} \leq 1.$**

**Default**

**gammashell1** is determined from the systematics given by Table 8.1.

---

**gammashell2**

Constant for the global expression for the damping parameter for shell effects in the level density parameter  $\gamma$ , see Eq. (8.14).

**Examples**

**gammashell2 0.054**

**gammashell1 0.**

**Range**

**$0. \leq \text{gammashell2} \leq 0.2$**

**Default**

**gammashell2=0.**

**pairconstant**

Constant for the pairing energy expression in MeV, see Eq. (8.31).

**Examples**

`pairconstant 11.3`

**Range**

$0. \leq \text{pairconstant} \leq 30.$

**Default**

`pairconstant=12.`



---

## **Krotconstant**

Normalization constant for rotational enhancement for the ground state or fission barrier, to be multiplied with the r.h.s. of Eq. (8.87). On the input line, we read **Krotconstant**,  $Z$ ,  $A$ , value, fission barrier. If the number of the fission barrier is not given or is equal to 0, TALYS assumes it concerns the ground state.

## **Examples**

**Krotconstant 90 232 0.4**

**Krotconstant 94 239 1.1 2**

## **Range**

**$0.01 \leq \text{Krotconstant} \leq 100$ .**

## **Default**

**Krotconstant  $Z A 1$ .**

**cfermi**

Width  $C_f$  of the phenomenological Fermi distribution (8.90) for damping of collective effects, in MeV.

**Examples**

**cfermi 16.**

**Range**

**$0. \leq \text{cfermi} \leq 1000.$**

**Default**

**cfermi 10.**

**cfermibf**

Width  $C_f$  of the phenomenological Fermi distribution (8.90) for damping of collective effects on the fission barrier, in MeV.

**Examples**

**cfermibf 16.**

**Range**

**$0. \leq \text{cfermibf} \leq 1000.$**

**Default**

**cfermibf 10.**

**Ufermi**

Constant  $U_f$  of the phenomenological function (8.90) for damping of collective effects, in MeV.

**Examples**

Ufermi 45.

**Range**

$0. \leq Ufermi \leq 1000.$

**Default**

Ufermi 30.

**Ufermibf**

Constant  $U_f$  of the phenomenological function (8.90) for damping of collective effects on the fission barrier, in MeV.

**Examples**

**Ufermibf 90.**

**Range**

**$0. \leq \text{Ufermibf} \leq 1000.$**

**Default**

**Ufermibf 45.**

**Rtransmom**

Normalization constant for moment of inertia for transition states, see Eq. (8.86). On the input line, we read **Rtransmom**,  $Z$ ,  $A$ , value, fission barrier. If the number of the fission barrier is not given or is equal to 0, TALYS assumes it concerns the first barrier.

**Examples**

**Rtransmom 90 232 1.15**

**Rtransmom 94 239 1.1 2**

**Range**

$0.1 \leq \text{Rtransmom} \leq 10$ .

**Default**

**Rtransmom 0.6** for the first barrier, **Rtransmom 1.0** for the other barriers.

---

## **Rclass2mom**

Normalization constant for moment of inertia for class II/III states, see Eq. (8.86). On the input line, we read **Rclass2mom**,  $Z$ ,  $A$ , value, fission barrier. If the number of the fission barrier is not given or is equal to 0, TALYS assumes it concerns the first barrier well.

### **Examples**

**Rclass2mom 90 232 1.15**

**Rclass2mom 94 239 1.1 2**

### **Range**

$0.1 \leq \text{Rclass2mom} \leq 10.$

### **Default**

**Rclass2mom Z A 1.**

**cglobal**

Constant  $c$  of the adjustment function (8.94) for tabulated level densities, applied for all nuclides at the same time. Individual cases can be overruled by the **ctable** keyword. On the input line, we read **cglobal**, value.

**Examples**

**cglobal 1.5**

**Range**

**-10. ≤ cglobal ≤ 10.**

**Default**

**cglobal 0.**



---

**pglobal**

Constant  $\delta$  of the adjustment function (8.94) for tabulated level densities, applied for all nuclides at the same time. Individual cases can be overruled by the **ptable** keyword. On the input line, we read **pglobal**, value.

**Examples**

**pglobal 0.5**

**Range**

**-10.**

*eq* **pglobal  $\leq$  10**

**Default**

**pglobal 0.**



## **26. Keywords for gamma emission**

**gammax**

Maximum number of  $l$ -values for gamma multipolarity, whereby  $l = 1$  stands for M1 and E1 transitions,  $l = 2$  for M2 and E2 transitions, etc.

**Examples**

**gammax 1**

**Range**

**$1 \leq \text{gammax} \leq 6$**

**Default**

**gammax 2**

**gnorm**

Normalisation factor for gamma-ray transmission coefficient. This adjustable parameter can be used to scale e.g. the  $(n, \gamma)$  cross section.

**Examples**

**gnorm 1.6**

**gnorm -1.** (enforce automatic normalization)

**Range**

**0. ≤ gnorm ≤ 100.** or **gnorm -1.**

**Default**

**gnorm** is given by the normalization factor of Eq. (9.17) for **strength 1** or **2**.

**strength**

Model for  $E1$  gamma-ray strength function, see Section 9. There are 9 possibilities.

**Examples**

- strength 1** : Kopecky-Uhl generalized Lorentzian
- strength 2** : Brink-Axel Lorentzian
- strength 3** : Hartree-Fock BCS tables
- strength 4** : Hartree-Fock-Bogoliubov tables
- strength 5** : Goriely's hybrid model [48]
- strength 6** : Goriely T-dependent HFB
- strength 7** : T-dependent RMF
- strength 8** : Gogny D1M HFB+QRPA
- strength 9** : SMLO

**Range**

1 - 9

**Default**

**strength 1** for incident neutrons, **strength 2** for other incident particles.

**strengthM1**

Model for  $M1$  gamma-ray strength function. There are two possibilities.

**Examples**

**strengthM1 1** : Use Eq. (9.9)

**strengthM1 2** : Normalize the  $M1$  gamma-ray strength function with that of  $E1$  as  $f_{E1}/(0.0588A^{0.878})$ .

**Range**

1 - 2

**Default**

**strengthM1 2**

**egr**

Energy of the giant dipole resonance in MeV. On the input line, we read **egr**,  $Z$ ,  $A$ , value, type of radiation (the full symbol, i.e. M1, E1, E2, etc.), number of resonance (optional). If the number of the resonance is not given, it is assumed the keyword concerns the first Lorentzian.

**Examples**

```
egr 41 93 16.2 E1
```

```
egr 94 239 13.7 E1 2
```

**Range**

$1. \leq \text{egr} \leq 100$ . The optional number of the resonance must be either 1 or 2.

**Default**

**egr** is read from the *talys/structure/gamma/* directory. If the value for the first resonance is not present in the directory, it is calculated from systematics, see Section 9. If no parameter for the second resonance is given, this term is omitted altogether.



**sgr**

Strength of the giant dipole resonance in millibarns. On the input line, we read **sgr**, *Z*, *A*, value, type of radiation (the full symbol, i.e. M1, E1, E2, etc.), number of resonance (optional). If the number of the resonance is not given, it is assumed the keyword concerns the first Lorentzian.

**Examples**

**sgr 41 93 221. E1**

**sgr 94 239 384. E1 2**

**Range**

$0. \leq \text{sgr} \leq 10000$ . The optional number of the resonance must be either 1 or 2.

**Default**

**sgr** is read from the *talys/structure/gamma/* directory. If the value for the first resonance is not present in the directory, it is calculated from systematics, see Section 9. If no parameter for the second resonance is given, this term is omitted altogether.

**ggr**

Width of the giant dipole resonance in MeV. On the input line, we read **ggr**,  $Z$ ,  $A$ , value, type of radiation (the full symbol, i.e. M1, E1, E2, etc.), number of resonance (optional). If the number of the resonance is not given, it is assumed the keyword concerns the first Lorentzian.

**Examples**

**ggr 41 93 5.03 E1**

**ggr 94 239 4.25 E1 2**

**Range**

$1. \leq \mathbf{ggr} \leq 100$ . The optional number of the resonance must be either 1 or 2.

**Default**

**ggr** is read from the *talys/structure/gamma/* directory. If the value for the first resonance is not present in the directory, it is calculated from systematics, see Section 9. If no parameter for the second resonance is given, this term is omitted altogether.

## **egradjust**

Normalisation factor for the energy of the giant dipole resonance. This parameter can be used as a relative normalisation instead of the absolute value of **egr**.

### **Examples**

```
egradjust 45 104 0.9
```

### **Range**

$0.1 \leq \text{egradjust} \leq 10$ .

### **Default**

egradjust Z A 1..

### **Comments**

- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

### **sgradjust**

Normalisation factor for the strength of the giant dipole resonance. This parameter can be used as a relative normalisation instead of the absolute value of **sgr**.

#### **Examples**

```
sgradjust 45 104 0.9
```

#### **Range**

$0.1 \leq \text{sgradjust} \leq 10$ .

#### **Default**

sgradjustZ A 1..

#### **Comments**

- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

---

## **ggradjust**

Normalisation factor for the width of the giant dipole resonance. This parameter can be used as a relative normalisation instead of the absolute value of **ggr**.

### **Examples**

```
ggradjust 45 104 0.9
```

### **Range**

$0.1 \leq \text{ggradjust} \leq 10$ .

### **Default**

**ggradjustZ A 1.**

### **Comments**

- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

**gamgam**

The total radiative width,  $\Gamma_\gamma$  in eV. On the input line, we read **gamgam**,  $Z$ ,  $A$ , value.

**Examples**

**gamgam 26 55 1.8**

**Range**

**$0. \leq \text{gamgam} \leq 10.$**

**Default**

**gamgam** is read from the *talys/structure/resonances/* directory, or, if not present there, is taken from interpolation, see Section 9.

## gamgamadjust

Normalisation factor for the average radiative width  $\Gamma_\gamma$ . This parameter can be used to scale e.g. the  $(n, \gamma)$  cross section.

### Examples

```
gamgamadjust 45 104 0.9
```

### Range

$0.1 \leq \text{gamgamadjust} \leq 10$ .

### Default

gamgamadjustZ A 1.

### Comments

- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

**outgamma**

Flag for the output of gamma-ray parameters, strength functions, transmission coefficients and reaction cross sections.

**Examples**

`outgamma y`

`outgamma n`

**Range**

y or n

**Default**

the same value as **outbasic**: `outgamma n`



**D0**

The s-wave resonance spacing  $D_0$  in keV. On the input line, we read **D0**,  $Z$ ,  $A$ , value.

**Examples**

**D0 26 55 13.**

**Range**

**$1.e - 6 \leq D0 \leq 10000.$**

**Default**

**D0** is read from the *talys/structure/resonances/* directory.

**S0**

The s-wave strength function  $S_0$  in units of  $10^{-4}$ . On the input line, we read **S0**, Z, A, value.

**Examples**

**S0 26 55 6.90**

**Range**

**$0. \leq S0 \leq 10.$**

**Default**

**S0** is read from the *talys/structure/resonances/* directory.

**Rprime**

Potential scattering radius.

**Examples**

**Rprime 6.**

**Rprime 0.3**

**Range**

**$0. \leq \text{Rprime} \leq 10.$**

**Default**

**Rprime** is derived from the OMP.

**etable**

Constant  $E_{\text{shift}}$  of the adjustment function (9.15) for tabulated gamma strength functions densities, per nucleus. On the input line, we read **etable**,  $Z$ ,  $A$ , value.

**Examples**

**etable 29 65 -0.6**

**Range**

**-10. ≤ etable ≤ 10.**

**Default**

**etableZ A 0.**

---

**etableadjust**

Adjustable shift for  $E_{\text{shift}}$  of the adjustment function (9.15) for tabulated gamma strength functions densities, per nucleus. On the input line, we read **etableadjust**,  $Z$ ,  $A$ , value.

**Examples**

**etableadjust 29 65 -0.6**

**Range**

**-10. ≤ etableadjust ≤ 10.**

**Default**

**etableadjustZ A 0.**

**ftable**

Constant  $f^{nor}$  of the adjustment function (9.15) for tabulated gamma strength functions densities, per nucleus. On the input line, we read **ftable**,  $Z$ ,  $A$ , value.

**Examples**

**ftable 29 65 1.2**

**Range**

$0.1 \leq \text{ftable} \leq 10$ .

**Default**

**ftable** $Z A 1$ .

---

**ftableadjust**

Adjustable shift for  $f^{nor}$  of the adjustment function (9.15) for tabulated gamma strength functions densities, per nucleus. On the input line, we read **ftableadjust**,  $Z$ ,  $A$ , value.

**Examples**

**ftableadjust 29 65 1.2**

**Range**

**$0.1 \leq \text{ftableadjust} \leq 10$ .**

**Default**

**ftableadjust  $Z A 1$ .**

**wtable**

Adjustable width of the adjustment function (9.15) for tabulated gamma strength functions densities, per nucleus. On the input line, we read **wtable**,  $Z$ ,  $A$ , value.

**Examples**

**wtable 29 65 1.2**

**Range**

**$0. \leq \text{wtable} \leq 10.$**

**Default**

**wtable $Z A 1.$**



---

**wtableadjust**

Adjustable factor for the width of the adjustment function (9.15) for tabulated gamma strength functions densities, per nucleus. On the input line, we read **wtableadjust**,  $Z$ ,  $A$ , value.

**Examples**

**wtableadjust 29 65 1.2**

**Range**

**$0. \leq \text{wtableadjust} \leq 10.$**

**Default**

**wtableadjust  $Z A 1.$**

**E1file**

File with tabulated gamma-ray strength function. The format of the file is the same as that of the nuclear structure database *talys/structure/gamma/hfb/*. On the input line, we read **E1file**, *Z*, *A*, filename.

**Examples**

**E1file 26 56 gam.loc**

**Range**

**E1file** can be equal to any filename, provided it starts with a character.

**Default**

If **E1file** is not given in the input file, the default applies.

**E1file**

File with tabulated gamma-ray strength function. The format of the file is the same as that of the nuclear structure database *talys/structure/gamma/hfb/*. On the input line, we read **M1file**, *Z*, *A*, filename.

**Examples**

**M1file 26 56 gam.loc**

**Range**

**M1file** can be equal to any filename, provided it starts with a character.

**Default**

If **M1file** is not given in the input file, the default applies.

**filepsf**

Flag to write photon strength functions to a separate file.

**Examples**

**filepsf y**

**filepsf n**

**Range**

**y or n**

**Default**

**filepsf n**

**psfglobal**

Flag to use only global parameterization for PSF, instead of tuned parameters.

**Examples**

```
psfglobal y  
psfglobal n
```

**Range**

y or n

**Default**

psfglobal n

**fiso**

Correction factor for isospin-forbidden transitions to self-conjugate nuclei [141]. Since isospin-dependent level densities are currently not implemented in TALYS, this is a way to handle transitions to  $Z = N$  or  $Z = N \pm 1$  nuclei. On the input line, we read **fiso**, particle type value.

**Examples**

**fiso n 2.2**

**Range**

**$0.01 \leq \text{fiso} \leq 100$ .**

**Default**

If  $Z = N$ , **fiso 2.** for incident neutrons and protons, **fiso 5.** for incident alpha's. If  $Z = N \pm 1$ , **fiso 1.5** for incident neutrons, protons and alpha's. In all other cases **fiso 1.**

---

**fisom**

Correction factor for multiple emission for isospin-forbidden transitions to self-conjugate nuclei [141]. Since isospin-dependent level densities are currently not implemented in TALYS, this is a way to handle transitions to  $Z = N$  or  $Z = N \pm 1$  nuclei. On the input line, we read **fisom**, particle type, value.

**Examples**

**fisom n 2.2**

**Range**

**$0.01 \leq \text{fisom} \leq 100$ .**

**Default**

If  $Z = N$ , **fisom 2.** for incident neutrons and protons, **fisom 5.** for incident alpha's. If  $Z = N \pm 1$ , **fisom 1.5** for incident neutrons, protons and alpha's. In all other cases **fisom 1.**

**racap**

Flag for the inclusion of the direct capture model, for addition to the gamma cross section.

**Examples**

**racap y**

**racap n**

**Range**

**y or n**

**Default**

**racap n**



---

## **ldmodelracap**

Level density model for direct capture model. There are 3 options.

### **Examples**

**ldmodelracap 1:** Spin-parity dependent ph-state densities

**ldmodelracap 2:** Total ph-state densities

**ldmodelracap 3:** Spin-parity dependent total level densities (**ldmodel 5**)

### **Range**

$1 \leq \text{ldmodelracap} \leq 3$

### **Default**

**ldmodelracap 1**

**sfexp**

Experimental spectroscopic factor for direct capture reactions. This keyword is only active if **racap** y. On the input line, we read **sfexp**, Z, A, discrete level, value. The input for Z, A, discrete level, is optional.

**Examples**

```
sfexp 20 40 2 0.1
sfexp 2 0.3
sfexp 20 40 0.1
sfexp 0.2
```

**Range**

$0. \leq \text{sfexp} \leq 10.$

**Default**

**sfexp** is taken from *talys/structure/levels/spectn*, and if not available there, **sfexp Z A level 0.347**.

---

**sfth**

Theoretical spectroscopic factor for direct capture reactions, for levels in the continuum. This keyword is only active if **racap y**. On the input line, we read **sfth**, *Z*, *A*, value. The input for *Z*, *A*, is optional.

**Examples**

**sfth 20 40 0.1**

**sfth 0.3**

**Range**

**0. ≤ sfth ≤ 10.**

**Default**

**sfth *Z A* 0.5.**

**epr**

Energy of the Pygmy resonance in MeV. On the input line, we read **epr**,  $Z$ ,  $A$ , value, type of radiation (the full symbol, i.e. M1, E1, E2, etc.), number of resonance (optional). If the number of the resonance is not given, it is assumed the keyword concerns the first Lorentzian.

**Examples**

**epr 41 93 10.2 E1**

**epr 94 239 8.7 E1 2**

**Range**

$1. \leq \mathbf{epr} \leq 100$ . The optional number of the resonance must be either 1 or 2.

**Default**

no default.

---

**spr**

Strength of the Pygmy resonance in millibarns. On the input line, we read **spr**,  $Z$ ,  $A$ , value, type of radiation (the full symbol, i.e. M1, E1, E2, etc.), number of resonance (optional). If the number of the resonance is not given, it is assumed the keyword concerns the first Lorentzian.

**Examples**

**spr 41 93 21.6 E1**

**spr 94 239 3.4 E1 2**

**Range**

$0. \leq \text{spr} \leq 10000$ . The optional number of the resonance must be either 1 or 2.

**Default**

no default.

**gpr**

Width of the Pygmy resonance in MeV. On the input line, we read **gpr**,  $Z$ ,  $A$ , value, type of radiation (the full symbol, i.e. M1, E1, E2, etc.), number of resonance (optional). If the number of the resonance is not given, it is assumed the keyword concerns the first Lorentzian.

**Examples**

**gpr 41 93 5.03 E1**

**gpr 94 239 4.25 E1 2**

**Range**

$0.1 \leq \text{gpr} \leq 100$ . The optional number of the resonance must be either 1 or 2.

**Default**

no default.

---

## **epradjust**

Normalisation factor for the energy of the Pygmy resonance. This parameter can be used as a relative normalisation instead of the absolute value of **epr**.

### **Examples**

```
epradjust 45 104 0.9
```

### **Range**

$0.1 \leq \text{epradjust} \leq 10$ .

### **Default**

epradjust Z A 1..

### **Comments**

- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

**spradjust**

Normalisation factor for the strength of the Pygmy resonance. This parameter can be used as a relative normalisation instead of the absolute value of **spr**.

**Examples**

```
spradjust 45 104 0.9
```

**Range**

$0.1 \leq \text{spradjust} \leq 10$ .

**Default**

spradjust Z A 1.

**Comments**

- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.



## **gpradjust**

Normalisation factor for the width of the Pygmy resonance. This parameter can be used as a relative normalisation instead of the absolute value of **gpr**.

### **Examples**

```
gpradjust 45 104 0.9
```

### **Range**

$0.1 \leq \text{gpradjust} \leq 10$ .

### **Default**

**gpradjust Z A 1.**

### **Comments**

- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

**upbend**

Flag to use the low-energy upbend of given multipolarity.

**Examples**

**upbend y**  
**upbend n**

**Range**

**y or n**

**Default**

**upbend n**

**upbendc**

Normalisation factor for the low-energy upbend of the gamma ray strength function.

**Examples**

`upbendc 57 138 1.e-7 M1`

**Range**

$0. \leq \text{upbendc} \leq 1.e-5$

**Default**

`upbendc Z A 0.`

**upbende**

Energy-dependent factor for the low-energy upbend of the gamma ray strength function.

**Examples**

`upbende 57 138 1.5 M1`

**Range**

$0. \leq \text{upbende} \leq 10.$

**Default**

`upbende Z A 0..`

**upbendf**

Normalization factor for the low-energy upbend of the gamma ray strength function.

**Examples**

**upbendf 57 138 1.5 M1**

**Range**

**$-10. \leq \text{upbendf} \leq 10.$**

**Default**

**upbendf Z A 0..**



## **27. Keywords for optical model**

### **rvadjust**

Multiplier to adjust the OMP parameter  $rv$  of Eq. (10.7). On the input line, we read **rvadjust**, particle symbol, and value.

#### **Examples**

**rvadjust t 1.04**

#### **Range**

$0.1 \leq \text{rvadjust} \leq 10$ .

#### **Default**

**rvadjust** *particle* 1.

#### **Comments**

- This keyword can also be used for complex particle potentials.
- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.



---

## avadjust

Multiplier to adjust the OMP parameter  $av$  of Eq. (10.7). On the input line, we read **avadjust**, particle symbol, and value.

### Examples

**avadjust d 0.97**

### Range

$0.1 \leq \text{avadjust} \leq 10$ .

### Default

**avadjust** *particle* 1.

### Comments

- This keyword can also be used for complex particle potentials.
- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

### **v1adjust**

Multiplier to adjust the OMP parameter  $v_1$  of Eq. (10.7). On the input line, we read **v1adjust**, particle symbol, and value.

#### **Examples**

**v1adjust a 1.12**

#### **Range**

$0.1 \leq \text{v1adjust} \leq 10$ .

#### **Default**

**v1adjust** *particle* **1**.

#### **Comments**

- This keyword can also be used for double-folding and complex particle potentials.
- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

**v2adjust**

Multiplier to adjust the OMP parameter  $v_2$  of Eq. (10.7). On the input line, we read **v2adjust**, particle symbol, and value.

**Examples**

**v2adjust n 0.96**

**Range**

$0.1 \leq \text{v2adjust} \leq 10$ .

**Default**

**v2adjust** *particle* 1.

**Comments**

- This keyword can only be used for a potential of the KD03 form.
- This keyword does not apply to deuterons up to alpha's.
- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

**v3adjust**

Multiplier to adjust the OMP parameter  $v_3$  of Eq. (10.7). On the input line, we read **v3adjust**, particle symbol, and value.

**Examples**

**v3adjust p 1.10**

**Range**

$0.1 \leq \text{v3adjust} \leq 10$ .

**Default**

**v3adjust** *particle 1*.

**Comments**

- This keyword can only be used for a potential of the KD03 form.
- This keyword does not apply to deuterons up to alpha's.
- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

## **v4adjust**

Multiplier to adjust the OMP parameter  $v_4$  of Eq. (10.7). On the input line, we read **v4adjust**, particle symbol, and value.

### **Examples**

**v4adjust n 0.98**

### **Range**

$0.1 \leq \text{v4adjust} \leq 10$ .

### **Default**

**v4adjust** *particle* **1**.

### **Comments**

- This keyword can only be used for a potential of the Koning-Delaroche form.
- This keyword does not apply to deuterons up to alpha's.
- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

**rwadjust**

Multiplier to adjust the OMP parameter  $rw$  of Eq. (10.7). On the input line, we read **rwadjust**, particle symbol, and value.

**Examples**

**rwadjust t 1.04**

**Range**

$0.1 \leq \text{rwadjust} \leq 10$ .

**Default**

**rwadjust** *particle* 1.

**Comments**

- This keyword can also be used for complex particle potentials.
- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

## **awadjust**

Multiplier to adjust the OMP parameter  $aw$  of Eq. (10.7). On the input line, we read **awadjust**, particle symbol, and value.

### **Examples**

**awadjust d 0.97**

### **Range**

$0.1 \leq \text{awadjust} \leq 10$ .

### **Default**

**awadjust** *particle* **1**.

### **Comments**

- This keyword can also be used for complex particle potentials.
- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

**w1adjust**

Multiplier to adjust the OMP parameter  $w_1$  of Eq. (10.7). On the input line, we read **w1adjust**, particle symbol, and value.

**Examples**

**w1adjust p 1.10**

**Range**

$0.1 \leq \text{w1adjust} \leq 10$ .

**Default**

**w1adjust** *particle* **1**.

**Comments**

- This keyword can also be used for double-folding and complex particle potentials.
- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.



---

## w2adjust

Multiplier to adjust the OMP parameter  $w_2$  of Eq. (10.7). On the input line, we read **w2adjust**, particle symbol, and value.

### Examples

**w2adjust n 0.80**

### Range

$0.1 \leq \text{w2adjust} \leq 10$ .

### Default

**w2adjust** *particle* 1.

### Comments

- This keyword can only be used for a potential of the KD03 form.
- This keyword does not apply to deuterons up to alpha's.
- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

### w3adjust

Multiplier to adjust the OMP parameter  $w_3$  of Eq. (10.18). On the input line, we read **w3adjust**, particle symbol, and value.

### Examples

**w3adjust n 0.80**

### Range

$0.1 \leq \text{w3adjust} \leq 10$ .

### Default

**w3adjust** *particle* 1.

### Comments

- This keyword can only be used for the high-energy extension of the KD03 form.
- This keyword does not apply to deuterons up to alpha's.
- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

## w4adjust

Multiplier to adjust the OMP parameter  $w_4$  of Eq. (10.18). On the input line, we read **w4adjust**, particle symbol, and value.

### Examples

**w4adjust n 0.80**

### Range

$0.1 \leq \text{w4adjust} \leq 10$ .

### Default

**w4adjust** *particle* 1.

### Comments

- This keyword can only be used for the high-energy extension of the KD03 form.
- This keyword does not apply to deuterons up to alpha's.
- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

**rvdadjust**

Multiplier to adjust the OMP parameter *rvd* of Eq. (10.7). On the input line, we read **rvdadjust**, particle symbol, and value.

**Examples**

**rvdadjust d 0.97**

**Range**

$0.1 \leq \text{rvdadjust} \leq 10$ .

**Default**

**rvdadjust** *particle* 1.

**Comments**

- Usually **rvd** and **rvdadjust** are not used.
- This keyword can also be used for complex particle potentials.
- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

## **avdadjust**

Multiplier to adjust the OMP parameter *avd* of Eq. (10.7). On the input line, we read **avdadjust**, particle symbol, and value.

### **Examples**

```
avdadjust d 0.97
```

### **Range**

$0.1 \leq \text{avdadjust} \leq 10$ .

### **Default**

**avdadjust** *particle* 1.

### **Comments**

- Usually **avd** and **avdadjust** are not used.
- This keyword can also be used for complex particle potentials.
- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

### **rwdadjust**

Multiplier to adjust the OMP parameter *rwd* of Eq. (10.7). On the input line, we read **rwdadjust**, particle symbol, and value.

#### **Examples**

```
rwdadjust d 0.97
```

#### **Range**

$0.1 \leq \text{rwdadjust} \leq 10$ .

#### **Default**

**rwdadjust** *particle* 1.

#### **Comments**

- This keyword can also be used for complex particle potentials.
- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

---

## **awdadjust**

Multiplier to adjust the OMP parameter awd of Eq. (10.7). On the input line, we read **awdadjust**, particle symbol, and value.

### **Examples**

**awdadjust d 0.97**

### **Range**

$0.1 \leq \text{awdadjust} \leq 10$ .

### **Default**

**awdadjust** *particle* **1**.

### **Comments**

- This keyword can also be used for complex particle potentials.
- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

### **d1adjust**

Multiplier to adjust the OMP parameter  $d1$  of Eq. (10.7). On the input line, we read **d1adjust**, particle symbol, and value.

#### **Examples**

**d1adjust d 0.97**

#### **Range**

$0.1 \leq \text{d1adjust} \leq 10$ .

#### **Default**

**d1adjust** *particle* **1**.

#### **Comments**

- This keyword can also be used for complex particle potentials.
- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.



## **d2adjust**

Multiplier to adjust the OMP parameter  $d_2$  of Eq. (10.7). On the input line, we read **d2adjust**, particle symbol, and value.

### **Examples**

**d2adjust n 1.06**

### **Range**

$0.1 \leq \mathbf{d2adjust} \leq 10$ .

### **Default**

**d2adjust** *particle* **1**.

### **Comments**

- This keyword can only be used for a potential of the KD03 form.
- This keyword does not apply to deuterons up to alpha's.
- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

### **d3adjust**

Multiplier to adjust the OMP parameter  $d_3$  of Eq. (10.7). On the input line, we read **d3adjust**, particle symbol, and value.

#### **Examples**

**d3adjust n 1.06**

#### **Range**

$0.1 \leq \mathbf{d3adjust} \leq 10$ . This keyword does not apply to deuterons up to alpha's.

#### **Default**

**d3adjust** *particle* **1**.

#### **Comments**

- This keyword can only be used for a potential of the KD03 form.
- This keyword does not apply to deuterons up to alpha's.
- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

## **vso1adjust**

Multiplier to adjust the OMP parameter `vso1` of Eq. (10.7). On the input line, we read **vso1adjust**, particle symbol, and value.

### **Examples**

**vso1adjust d 1.15**

### **Range**

$0.1 \leq \text{vso1adjust} \leq 10$ .

### **Default**

**vso1adjust** *particle* **1**.

### **Comments**

- This keyword can also be used for complex particle potentials.
- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

### **vso2adjust**

Multiplier to adjust the OMP parameter  $vso2$  of Eq. (10.7). On the input line, we read **vso2adjust**, particle symbol, and value.

#### **Examples**

**vso2adjust n 1.06**

#### **Range**

$0.1 \leq \text{vso2adjust} \leq 10$ . This keyword does not apply to deuterons up to alpha's.

#### **Default**

**vso2adjust** *particle* 1.

#### **Comments**

- This keyword can only be used for a potential of the Koning-Delaroche form.
- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

## wso1adjust

Multiplier to adjust the OMP parameter wso1 of Eq. (10.7). On the input line, we read **wso1adjust**, particle symbol, and value.

### Examples

**wso1adjust d 1.15**

### Range

$0.1 \leq \text{wso1adjust} \leq 10$ .

### Default

**wso1adjust** *particle* **1**.

### Comments

- This keyword can also be used for complex particle potentials.
- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

### wso2adjust

Multiplier to adjust the OMP parameter wso2 of Eq. (10.7). On the input line, we read **wso2adjust**, particle symbol, and value.

### Examples

**wso2adjust n 1.06**

### Range

$0.1 \leq \text{wso2adjust} \leq 10$ . This keyword does not apply to deuterons up to alpha's.

### Default

**wso2adjust** *particle* **1**.

### Comments

- This keyword can only be used for a potential of the Koning-Delaroche form.
- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

## **rvsoadjust**

Multiplier to adjust the OMP parameter  $rvso$  of Eq. (10.7). On the input line, we read **rvsoadjust**, particle symbol, and value.

### **Examples**

**rvsoadjust d 1.15**

### **Range**

$0.1 \leq \text{rvsoadjust} \leq 10$ .

### **Default**

**rvsoadjust** *particle* 1.

### **Comments**

- This keyword can also be used for complex particle potentials.
- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

### **avsoadjust**

Multiplier to adjust the OMP parameter avso of Eq. (10.7). On the input line, we read **avsoadjust**, particle symbol, and value.

#### **Examples**

**avsoadjust d 1.15**

#### **Range**

$0.1 \leq \text{avsoadjust} \leq 10$ .

#### **Default**

**avsoadjust** *particle* **1**.

#### **Comments**

- This keyword can also be used for complex particle potentials.
- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.



## **rwsadjust**

Multiplier to adjust the OMP parameter *rwso* of Eq. (10.7). On the input line, we read **rwsadjust**, particle symbol, and value.

### **Examples**

**rwsadjust d 1.15**

### **Range**

$0.1 \leq \text{rwsadjust} \leq 10$ .

### **Default**

**rwsadjust** *particle* **1**.

### **Comments**

- This keyword can also be used for complex particle potentials.
- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

### **awsoadjust**

Multiplier to adjust the OMP parameter *awso* of Eq. (10.7). On the input line, we read **awsoadjust**, particle symbol, and value.

#### **Examples**

**awsoadjust d 1.15**

#### **Range**

$0.1 \leq \text{awsoadjust} \leq 10$ .

#### **Default**

**awsoadjust** *particle* **1**.

#### **Comments**

- This keyword can also be used for complex particle potentials.
- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

---

## **rcadjust**

Multiplier to adjust the OMP parameter  $rc$  of Eq. (10.7). On the input line, we read **rcadjust**, particle symbol, and value.

### **Examples**

**rcadjust d 1.15**

### **Range**

$0.1 \leq \text{rcadjust} \leq 10$ .

### **Default**

**rcadjust** *particle* **1**.

### **Comments**

- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

**dispersion**

Flag to invoke the dispersive optical model, see Section 10.2. These potentials are only available as tabulated neutron local potentials. If not available, TALYS will automatically resort to normal OMP's.

**Examples**

**dispersion n**

**Range**

y or n

**Default**

**dispersion n.**

**localomp**

Flag to overrule the local, nucleus-specific optical model by the global optical model of Eqs. (10.8) or (10.9). This may be helpful to study global mass-dependent trends.

**Examples**

**localomp n**

**Range**

**y or n**

**Default**

**localomp y**, i.e. a nucleus-specific optical model, when available.

**deuteronomp**

Specific deuteron OMP. The default deuteron optical model coming from standard Watanabe folding does not always perform well. Therefore, we have included various other options:

**Examples**

**deuteronomp 1:** Normal deuteron potential [60]

**deuteronomp 2:** Deuteron potential of Daehnick et al. [62]

**deuteronomp 3:** Deuteron potential of Bojowald et al.[63]

**deuteronomp 4:** Deuteron potential of Han et al.[64]

**deuteronomp 5:** Deuteron potential of Haixia An et al.[65]

**Range**

$1 \leq \text{deuteronomp} \leq 5$

**Default**

**deuteronomp 1**

---

## **alphaomp**

Specific alpha OMP.

### **Examples**

**alphaomp 1:** Normal alpha potential[66]

**alphaomp 2:** Alpha potential of McFadden and Satchler

**alphaomp 3:** Alpha potential of Demetriou and Goriely [67], table 1.

**alphaomp 4:** Alpha potential of Demetriou and Goriely [67], table 2.

**alphaomp 5:** Alpha potential of Demetriou and Goriely [67], dispersive model.

**alphaomp 6:** Alpha potential of Avrigeanu et al. [69].

**alphaomp 7:** Alpha potential of Nolte et al. [146].

**alphaomp 8:** Alpha potential of Avrigeanu et al. [147].

### **Range**

$1 \leq \text{alphaomp} \leq 8$

### **Default**

**alphaomp 6**

**aradialcor**

Normalization factor for the shape of the double-folding alpha potential. This keyword is only relevant for **alphaomp**  $\geq 3$ . On the input line, we read **aradialcor** and value.

**Examples**

**aradialcor 1.07**

**Range**

**$0.5 \leq \text{aradialcor} \leq 1.5$**

**Default**

**aradialcor 1.**



**adepthcor**

Normalization factor for the depth of the double-folding alpha potential. This keyword is only relevant for **alphaomp**  $\geq 3$ . On the input line, we read **adepthcor** and value.

**Examples**

**adepthcor 1.07**

**Range**

**$0.5 \leq \text{adepthcor} \leq 1.5$**

**Default**

**adepthcor 1.**

**jlmomp**

Flag to use the JLM microscopic optical model potential instead of the phenomenological optical model potential, see Section 10.4.

**Examples**

**jlmomp n**

**Range**

**y or n**

**Default**

**jlmomp n**, i.e. to use the phenomenological OMP.

## **jlmmode**

Keyword to enable different normalizations for the imaginary potential of the JLM optical model, as explained in Ref. [148] These different normalizations apply to the constant 0.44 in Eq. (10.47). This keyword is only active if 'jlmomp y'.

### **Examples**

**jlmmode 0:** standard JLM imaginary potential of Eq. (10.47)

**jlmmode 1:** 0.44 replaced by  $1.1 \exp(-0.4E^{1/2})$

**jlmmode 2:** 0.44 replaced by  $1.25 \exp(-0.2E^{1/2})$

**jlmmode 3:** as jlmmode 2 but with  $\lambda_W(E)$  twice as large, recommended for energies below 1 MeV.

### **Range**

$0 \leq \text{jlmmode} \leq 3$

### **Default**

**jlmmode 0**, i.e. the standard JLM OMP.

**Ivadjust**

Normalization factor for the real central potential for JLM calculations, see Eq. (10.44). On the input line, we read **Ivadjust** and value.

**Examples**

**Ivadjust 1.15**

**Range**

$0.5 \leq \text{Ivadjust} \leq 1.5$

**Default**

**Ivadjust 1.**

**Comments**

- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

**lwadjust**

Normalization factor for the imaginary central potential for JLM calculations, see Eq. (10.45). On the input line, we read **lwadjust** and value.

**Examples**

**lwadjust 1.15**

**Range**

$0.5 \leq \text{lwadjust} \leq 1.5$

**Default**

**lwadjust 1.**

**Comments**

- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

**lv1adjust**

Normalization factor for the real isovector potential for JLM calculations, see Eq. (10.46). On the input line, we read **lv1adjust** and value.

**Examples**

**lv1adjust 1.15**

**Range**

$0.5 \leq \text{lv1adjust} \leq 1.5$

**Default**

**lv1adjust 1.**

**Comments**

- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

**lw1adjust**

Normalization factor for the imaginary isovector potential for JLM calculations, see Eq. (10.47). On the input line, we read **lw1adjust** and value.

**Examples**

**lw1adjust 1.15**

**Range**

$0.5 \leq \text{lw1adjust} \leq 1.5$

**Default**

**lw1adjust 1.**

**Comments**

- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

**lvsoadjust**

Normalization factor for the real spin-orbit potential for JLM calculations, see Eq. (10.50). On the input line, we read **lvsoadjust** and value.

**Examples**

**lvsoadjust 1.15**

**Range**

$0.5 \leq \text{lvsoadjust} \leq 1.5$

**Default**

**lvsoadjust 1.**

**Comments**

- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.



---

## **lwsoadjust**

Normalization factor for the imaginary spin-orbit potential for JLM calculations, see Eq. (10.51). On the input line, we read **lwsoadjust** and value.

### **Examples**

**lwsoadjust 1.15**

### **Range**

$0.5 \leq \text{lwsoadjust} \leq 1.5$

### **Default**

**lwsoadjust 1.**

### **Comments**

- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

**radialmodel**

Model for radial matter densities in the JLM optical model. There are two options. This keyword is only active if 'jlmomp y'.

**Examples**

**radialmodel 1:** HFB-Skyrme based matter densities

**radialmodel 2:** HFB-Gogny based matter densities

**Range**

$1 \leq \text{radialmodel} \leq 2$

**Default**

**radialmodel 2.**

**radialfile**

File with radial matter densities. The format of the file is exactly the same as that of the nuclear structure database *talys/structure/optical/jlm/*. In practice, the user can copy a file from this database to the working directory and change it. In this way, changes in the “official” database are avoided. Note that even if only changes for one isotope are required, the entire file needs to be copied if for the other isotopes the originally tabulated values are to be used. On the input line, we read **radialfile**, *Z*, filename.

**Examples**

**radialfile 26 Fe.loc**

**Range**

**radialfile** can be equal to any filename, provided it starts with a character.

**Default**

If **radialfile** is not given in the input, radial matter densities are taken from *talys/structure/optical/jlm/*.

**Ejoin**

Joining energy  $E_j$ , in MeV, between the original form of the KD03 OMP and its high-energy extension, see Eq. (10.18). On the input line, we read **Ejoin**, particle symbol, and value.

**Examples**

**Ejoin n 250.**

**Range**

**0. ≤ Ejoin ≤ 1000..**

**Default**

**Ejoin 200.**

## Vinfadjust

Multiplier to adjust the OMP parameter  $V_\infty$  for the real volume potential of Eq. (10.18). On the input line, we read **Vinfadjust**, particle symbol, and value.

### Examples

**Vinfadjust n 1.10**

### Range

$0.01 \leq \text{Vinfadjust} \leq 10$ .

### Default

**Vinfadjust** *particle 1*.

### Comments

- This keyword can only be used for the high-energy extension of the KD03 form.
- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

**incadjust**

Flag to use adjusted optical model parameters for the incident channel *and* exit channels. Disabling this flag allows to use a different OMP for the exit channels than for the incident channel, e.g. using **rvadjust n 1.10** together with **incadjust n** means that the OMP for the incident channel is unaltered, while that for the outgoing channel has an adjust  $r_V$  parameter. This can be used to simulate e.g. temperature-dependent OMP effects.

**Examples**

**incadjust n**

**Range**

y or n

**Default**

**incadjust y**

**omponly**

Flag to let TALYS perform *only* an optical model calculation. In this way, TALYS acts simply as a driver for ECIS. All non-elastic calculations for the various reaction channels are skipped. This is helpful for systematic, and quick, testing of optical model potentials.

**Examples**

- **omponly n**
- **omponly y**

**Range**

y or n

**Default**

**omponly n**

## optmod

File with tabulated phenomenological optical model parameters as a function of energy, see Section 10. This can be helpful if one wishes to use an optical model parameterisation which is not hardwired in TALYS. One could write a driver to automatically generate a table with parameters. On the input line, we read **optmod**,  $Z$ ,  $A$ , filename, and (optionally) particle type. From the file, TALYS first reads the number of lines to be read with format (8x,i4). Next, from each line of the file, TALYS reads  $E$ ,  $v$ ,  $rv$ ,  $av$ ,  $w$ ,  $rw$ ,  $aw$ ,  $vd$ ,  $rvd$ ,  $avd$ ,  $wd$ ,  $rwd$ ,  $awd$ ,  $vso$ ,  $rvso$ ,  $avso$ ,  $wso$ ,  $rwso$ ,  $awso$ , and  $rc$  in free format.

## Examples

```
optmod 40 90 ompzr90 d
optmod 94 239 omppu239
```

## Range

**optmod** can be equal to any filename, provided it starts with a character. The particle type must be equal to either **n**, **p**, **d**, **t**, **h** or **a**. A table of up to **numomp** incident energies (this is set in a module or the input) and associated parameters can be specified.

## Default

If the particle type is not given, as in the second example above, neutrons are assumed. If **optmod** is not given in the input file, the optical model parameters are taken from the *talys/structure/optical* database per nucleus or, if not present there, from the global optical model.



## **optmodfileN**

File with the neutron optical model parameters of Eq. (10.7). The format of the file is exactly the same as that of the nuclear structure database *talys/structure/optical/neutron/*. In practice, the user can copy a file from this database, e.g. *n-Fe.omp*, to the working directory and change it. In this way, changes in the “official” database are avoided. Note that even if only changes for one isotope are required, the file for the whole element needs to be copied if for the other isotopes the originally tabulated values are to be used. On the input line, we read **optmodfileN**, *Z*, filename.

### **Examples**

**optmodfileN 26 Fe.loc**

### **Range**

**optmodfileN** can be equal to any filename, provided it starts with a character, which should be present in the working directory.

### **Default**

If **optmodfileN** is not given in the input file, the optical model parameters are taken from the *talys/structure/optical/neutron/* database per nucleus or, if not present there, from the global optical model.

### **optmodfileP**

File with the proton optical model parameters of Eq. (10.7). The format of the file is exactly the same as that of the nuclear structure database *talys/structure/optical/proton/*. In practice, the user can copy a file from this database, e.g. *p-Fe.omp*, to the working directory and change it. In this way, changes in the “official” database are avoided. Note that even if only changes for one isotope are required, the file for the whole element needs to be copied if for the other isotopes the originally tabulated values are to be used. On the input line, we read **optmodfileP**, *Z*, filename.

### **Examples**

**optmodfileP 26 Fe.loc**

### **Range**

**optmodfileP** can be equal to any filename, provided it starts with a character, which should be present in the working directory.

### **Default**

If **optmodfileP** is not given in the input file, the optical model parameters are taken from the *talys/structure/optical/proton/* database per nucleus or, if not present there, from the global optical model.

---

**optmodall**

Flag for a new optical model calculation for each compound nucleus in the decay chain. In usual multiple Hauser-Feshbach decay, the transmission coefficients for the first compound nucleus are used for the whole decay chain. When a residual nucleus is far away from the initial compound nucleus, this approximation may become dubious. With **optmodall y**, new optical model calculations are performed for every compound nucleus that is depleted, for all types of emitted particles.

**Examples**

**optmodall y**

**Range**

y or n

**Default**

**optmodall n**

**outomp**

Flag for the output of optical model parameters for each particle and energy.

**Examples**

**outomp y**

**outomp n**

**Range**

**y** or **n**

**Default**

the same value as **outbasic**: **outomp n**

**outkd**

Flag for the output of KD03 OMP parameters for each particle and energy.

**Examples**

**outkd y**  
**outkd n**

**Range**

**y or n**

**Default**

**outkd n**

**soukho**

Flag for the use of Soukhovitskii OMP for actinides instead of KD03.

**Examples**

**soukho y**  
**soukho n**

**Range**

**y** or **n**

**Default**

**soukho y**

**Comments**

This is currently overruled by RIPL OMP.

**riplomp**

RIPL OMP number for particle. On the input line, we read **riplomp**, particle type, RIPL OMP number.

**Examples**

**riplomp n 2408**

**riplomp n 2413**

**Range**

**riplomp** should exist as a RIPL OMP number.

**Default**

**riplomp** is only used as default for incident neutrons on actinides, in which case **riplomp n 2408**.

**riprisk**

Flag for using the RIPL OMP outside the specified mass range.

**Examples**

**riprisk y**  
**riprisk n**

**Range**

**y or n**

**Default**

**riprisk n**



## rvadjustF

Energy-dependent function to adjust the OMP parameter  $r_V$ . If physically adequate OMPs fail, such energy-dependent adjustment can be invoked as a last resort. As long as the deviation from the original model is not too large, unpleasant surprises in the various reaction channels are avoided. On the input line, we read **rvadjustF**, particle symbol, begin energy  $E_b$  and end energy  $E_e$  in MeV, maximal deviation  $D$  in %, and variance  $\sigma$  in MeV, of the function. If this keyword is specified,  $r_V$  will keep its original constant value for  $E < E_b$  and  $E > E_e$  while in between these two values it will be multiplied by the function

$$f(E) = 1 + D \exp(-(E - E_m)^2 / 2\sigma^2) + R \quad (27.1)$$

where  $E_m = (E_e + E_b)/2$  and the offset value

$$R = -D \exp(-(E_e - E_m)^2 / 2\sigma^2) \quad (27.2)$$

ensures continuity at  $E_b$  and  $E_e$ . Fig. 27.1 shows an example for  $r_V$  with an original value of 1.20, which is locally multiplied by the function  $f$  with parameters  $E_b=2$ ,  $E_e=10$ ,  $D = 5$ ,  $\sigma = 2$ . Up to 10 energy ranges, i.e. **rvadjustF** keywords, per particle can be used. We suggest however that the keyword **rvadjustF** is now replaced by the method explained in Section 3.2.1 for all parameters. We keep **rvadjustF** and related OMP keywords for backward compat

## Examples

**rvadjustF n 2. 10. 5. 2.**

## Range

The particle symbol should be equal to n, p, d, t, h or a,  $0 \leq E_b \leq 250$ ,  $0 \leq E_e \leq 250$ ,  $E_b < E_e$ ,  $-100 \leq D \leq 100$ ,  $0 \leq \sigma \leq 100$ . If  $\sigma = 0$ , then the value  $\sigma = (E_e - E_m)/2$  will be adopted.

## Default

**rvadjustF** is not applied.

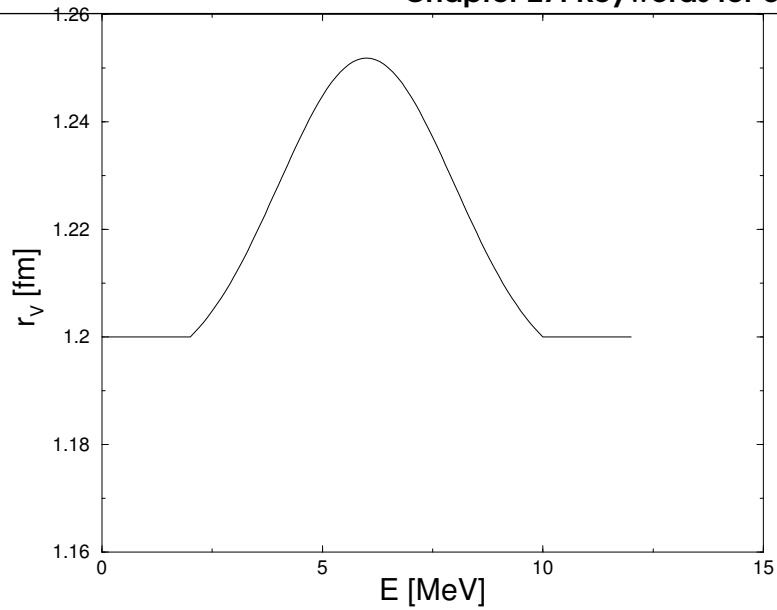


Figure 27.1: Energy-dependent radius  $r_V$  obtained with the **rvadjustF** keyword

### avadjustF

Energy-dependent function to adjust the OMP parameter  $a_V$ . On the input line, we read **avadjustF**, particle symbol, begin energy  $E_b$  and end energy  $E_e$  in MeV, maximal deviation  $D$  in % , and variance  $\sigma$  in MeV, of the function. The same formalism as explained for the **rvadjustF** keyword applies.

### Examples

**avadjustF n 2. 10. 5. 2.**

### Range

See **rvadjustF** keyword.

### Default

**avadjustF** is not applied.

**rwadjustF**

Energy-dependent function to adjust the OMP parameter  $a_V$ . On the input line, we read **rwadjustF**, particle symbol, begin energy  $E_b$  and end energy  $E_e$  in MeV, maximal deviation D in % , and variance  $\sigma$  in MeV, of the function. The same formalism as explained for the **rvadjustF** keyword applies.

**Examples**

```
rwadjustF n 2. 10. 5. 2.
```

**Range**

See **rvadjustF** keyword.

**Default**

**rwadjustF** is not applied.

**awadjustF**

Energy-dependent function to adjust the OMP parameter  $a_V$ . On the input line, we read **awadjustF**, particle symbol, begin energy  $E_b$  and end energy  $E_e$  in MeV, maximal deviation D in % , and variance  $\sigma$  in MeV, of the function. The same formalism as explained for the **rvadjustF** keyword applies.

**Examples**

**awadjustF n 2. 10. 5. 2.**

**Range**

See **rvadjustF** keyword.

**Default**

**awadjustF** is not applied.

---

## **rvdadjustF**

Energy-dependent function to adjust the OMP parameter `rvd`. On the input line, we read **rvdadjustF**, particle symbol, begin energy  $E_b$  and end energy  $E_e$  in MeV, maximal deviation  $D$  in % , and variance  $\sigma$  in MeV, of the function. The same formalism as explained for the **rvadjustF** keyword applies.

### **Examples**

```
rvdadjustF n 2. 10. 5. 2.
```

### **Range**

See **rvadjustF** keyword.

### **Default**

**rvdadjustF** is not applied.

**avdadjustF**

Energy-dependent function to adjust the OMP parameter *avd*. On the input line, we read **avdadjustF**, particle symbol, begin energy  $E_b$  and end energy  $E_e$  in MeV, maximal deviation  $D$  in % , and variance  $\sigma$  in MeV, of the function. The same formalism as explained for the **rvadjustF** keyword applies.

**Examples**

**avdadjustF n 2. 10. 5. 2.**

**Range**

See **rvadjustF** keyword.

**Default**

**avdadjustF** is not applied.

**rwdadjustF**

Energy-dependent function to adjust the OMP parameter `rwd`. On the input line, we read **rwdadjustF**, particle symbol, begin energy  $E_b$  and end energy  $E_e$  in MeV, maximal deviation  $D$  in % , and variance  $\sigma$  in MeV, of the function. The same formalism as explained for the **rvadjustF** keyword applies.

**Examples**

```
rwdadjustF n 2. 10. 5. 2.
```

**Range**

See **rvadjustF** keyword.

**Default**

**rwdadjustF** is not applied.

**awdadjustF**

Energy-dependent function to adjust the OMP parameter awd. On the input line, we read **awdadjustF**, particle symbol, begin energy  $E_b$  and end energy  $E_e$  in MeV, maximal deviation  $D$  in % , and variance  $\sigma$  in MeV, of the function. The same formalism as explained for the **rvadjustF** keyword applies.

**Examples**

**awdadjustF n 2. 10. 5. 2.**

**Range**

See **rvadjustF** keyword.

**Default**

**awdadjustF** is not applied.



---

**rvsoadjustF**

Energy-dependent function to adjust the OMP parameter  $rvso$ . On the input line, we read **rvsoadjustF**, particle symbol, begin energy  $E_b$  and end energy  $E_e$  in MeV, maximal deviation  $D$  in % , and variance  $\sigma$  in MeV, of the function. The same formalism as explained for the **rvadjustF** keyword applies.

**Examples**

**rvsoadjustF n 2. 10. 5. 2.**

**Range**

See **rvadjustF** keyword.

**Default**

**rvsoadjustF** is not applied.

**avsoadjustF**

Energy-dependent function to adjust the OMP parameter avso. On the input line, we read **avsoadjustF**, particle symbol, begin energy  $E_b$  and end energy  $E_e$  in MeV, maximal deviation D in % , and variance  $\sigma$  in MeV, of the function. The same formalism as explained for the **rvadjustF** keyword applies.

**Examples**

**avsoadjustF n 2. 10. 5. 2.**

**Range**

See **rvadjustF** keyword.

**Default**

**avsoadjustF** is not applied.

---

## **rwsoadjustF**

Energy-dependent function to adjust the OMP parameter `rwso`. On the input line, we read **rwsoadjustF**, particle symbol, begin energy  $E_b$  and end energy  $E_e$  in MeV, maximal deviation  $D$  in % , and variance  $\sigma$  in MeV, of the function. The same formalism as explained for the **rvadjustF** keyword applies.

### **Examples**

```
rwsoadjustF n 2. 10. 5. 2.
```

### **Range**

See **rvadjustF** keyword.

### **Default**

**rwsoadjustF** is not applied.

**awsoadjustF**

Energy-dependent function to adjust the OMP parameter awso. On the input line, we read **awsoadjustF**, particle symbol, begin energy  $E_b$  and end energy  $E_e$  in MeV, maximal deviation  $D$  in % , and variance  $\sigma$  in MeV, of the function. The same formalism as explained for the **rvadjustF** keyword applies.

**Examples**

**awsoadjustF n 2. 10. 5. 2.**

**Range**

See **rvadjustF** keyword.

**Default**

**awsoadjustF** is not applied.

## **28. Keywords for direct reactions and ECIS**

**spherical**

Flag to enforce a spherical OMP calculation, regardless of the availability of a deformed OMP and a coupling scheme. Direct inelastic scattering will then be treated by DWBA.

**Examples**

**spherical n**

**Range**

**y or n**

**Default**

**spherical n**

## **rotational**

Flag to enable or disable the rotational optical model for the various particles appearing in the calculation. This flag is to enable or disable coupled-channels calculations for the inverse channels provided a coupling scheme is given in the deformation database. Using no rotational model at all can be set with another keyword: **spherical y**.

### **Examples**

**rotational n**

**rotational n p a**

### **Range**

**rotational** can be any combination of **n, p, d, t, h** and **a**

### **Default**

**rotational n p**. Warning: setting e.g. **rotational a** will thus automatically disable the default setting. If this needs to be retained as well, set **rotational n p a**

**Tljadjust**

Multiplier to adjust the OMP transmission coefficient per  $l$ -value. On the input line, we read **Tljadjust**, particle symbol, value and  $l$ -value.

**Examples**

**Tljadjust n 0.80 0**

**Range**

$0.001 \leq \text{Tljadjust} \leq 1000$ .

**Default**

**Tljadjust** *particle 1. l-value*

**Comments**

- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.



**sysreaction**

The types of particles for which the optical model reaction cross section is overruled by values obtained from systematics, see Section 10.5. The optical model transmission coefficients will be accordingly normalized.

**Examples**

**sysreaction p**

**sysreaction d a**

**Range**

**sysreaction** can be any combination of **n, p, d, t, h** and **a**

**Default**

**sysreaction** is disabled for any particle.

**ecissave**

Flag for saving ECIS input and output files. This has two purposes: (a) if the next calculation will be performed with already existing reaction cross sections and transmission coefficients. This is helpful for time-consuming coupled-channels calculations, (b) to study the ECIS input and output files in detail. **ecissave** *must* be set to **y**, if in the next run **inccalc n** or **eciscalc n** will be used. If not, an appropriate error message will be given and TALYS stops.

**Examples**

```
ecissave y
ecissave n
```

**Range**

y or n

**Default**

**ecissave n**

**Comments**

- Suppose you want to study the effect of changing variables other than for the OMP in a calculation. Then it can be time consuming to repeat complicated coupled-channels calculations while you are only interested in changing e.g. a level density parameter. Then, in the first run you do

```
ecissave y
inccalc y      (the default)
eciscalc y     (the default)
```

while the next runs can all be done with

```
ecissave y
inccalc n
eciscalc n
```

which reads in the previously generated transmission coefficients, cross sections etc. from files.

## **eciscalc**

Flag for the ECIS calculation of transmission coefficients and reaction cross sections for the inverse channels. If this calculation has already been performed in a previous run, and in that previous run **ecissave y** has been set, it may be helpful to put **eciscalc n**, which avoids a new calculation. This saves time, especially in the case of coupled-channels calculations. We stress that it is the responsibility of the user to ensure that the first run of a particular problem is done with **ecissave y**. You also have to make sure that the same energy grid for inverse channels is used.

### **Examples**

```
eciscalc y
```

```
eciscalc n
```

### **Range**

y or n

### **Default**

```
eciscalc y
```

### **Comments**

- Suppose you want to study the effect of changing variables other than for the OMP in a calculation. Then it can be time consuming to repeat complicated coupled-channels calculations while you are only interested in changing e.g. a level density parameter. Then, in the first run you do

```
ecissave y
inccalc y      (the default)
eciscalc y      (the default)
```

while the next runs can all be done with

```
ecissave y
inccalc n
eciscalc n
```

which reads in the previously generated transmission coefficients, cross sections etc. from files.

**inccalc**

Flag for the ECIS calculation of transmission coefficients and reaction cross sections for the incident channel. If this calculation has already been performed in a previous run, and in that previous run **ecissave y** has been set, it may be helpful to put **inccalc n**, which avoids a new ECIS calculation. This saves time, especially in the case of coupled-channels calculations. We stress that it is the responsibility of the user to ensure that the first run of a particular problem is done with **eciscalc y**. If not, an appropriate error message will be given and TALYS stops. You also have to make sure that the same grid of incident energies is used.

**Examples**

```
inccalc y
inccalc n
```

**Range**

y or n

**Default**

inccalc y

**Comments**

- Suppose you want to study the effect of changing variables other than for the OMP in a calculation. Then it can be time consuming to repeat complicated coupled-channels calculations while you are only interested in changing e.g. a level density parameter. Then, in the first run you do

```
ecissave y
inccalc y      (the default)
eciscalc y     (the default)
```

while the next runs can all be done with

```
ecissave y
inccalc n
eciscalc n
```

which reads in the previously generated transmission coefficients, cross sections etc. from files.

**maxrot**

Number of excited levels to be included in a rotational band of a deformed nucleus for coupled-channels calculations. For example, use **maxrot 4** if the  $0^+ - 2^+ - 4^+ - 6^+ - 8^+$  states need to be included.

**Examples**

**maxrot 4**

**Range**

**$0 \leq \text{maxrot} \leq 10$**

**Default**

**maxrot 2**

**maxband**

Maximum number of vibrational bands added to the rotational coupling scheme, regardless of the number of bands specified in the deformation database.

**Examples**

**maxband 4**

**Range**

**$0 \leq \text{maxband} \leq 10$**

**Default**

**maxband 0**

**autorot**

Flag for automatic rotational coupled-channels calculations, see Section 11.1, for  $A > 150$ . The discrete level database is scanned and an attempt is made to automatically identify the lowest rotational band. Deformation parameters are also read from the database so automated coupled-channels calculations can be performed. This option is possible for the rare earth and actinide region. Note that for all natural isotopes, the coupling scheme is already given in the *talys/structure/deformation* database.

**Examples**

**autorot y**

**Range**

y or n

**Default**

**autorot n**

**outdirect**

Flag for the output of the results from the direct reaction calculation of ECIS (DWBA, giant resonances and coupled-channels).

**Examples**

**outdirect y**

**outdirect n**

**Range**

y or n

**Default**

the same value as **outbasic**: **outdirect n**



**outinverse**

Flag for the output of particle transmission coefficients and inverse reaction cross sections.

**Examples**

**outinverse y**  
**outinverse n**

**Range**

**y** or **n**

**Default**

the same value as **outbasic**: **outinverse n**

**outecis**

Flag for keeping the various ECIS output files produced during a TALYS run. This is mainly for diagnostic purposes.

**Examples**

**outecis y**

**outecis n**

**Range**

**y or n**

**Default**

**outecis n**

**endfecis**

Flag for the ECIS calculation of transmission coefficients and reaction cross sections for the ENDF-6 energy grid. If this calculation has already been performed in a previous run, it may be helpful to put **endfecis n**, which avoids a new calculation. This saves time, especially in the case of coupled-channels calculations. We stress that it is the responsibility of the user to ensure that the first run of a particular problem is done with **ecissave y** and **endfecis y**. If not, an appropriate error message will be given and TALYS stops.

**Examples**

**endfecis y**

**endfecis n**

**Range**

**y** or **n**

**Default**

**endfecis y**

**giantresonance**

Flag for the calculation of giant resonance contributions to the continuum part of the spectrum. The GMR, GQR, LEOR and HEOR are included.

**Examples**

**giantresonance y**

**giantresonance n**

**Range**

y or n

**Default**

**giantresonance y** for incident neutrons and protons, **giantresonance n** otherwise.

**elwidth**

Width of elastic peak in MeV. For comparison with experimental angle-integrated and double-differential spectra, it may be helpful to include the energy-broadened cross sections for discrete states in the high-energy tail of the spectra. **elwidth** is the width of the Gaussian spreading that takes care of this. **elwidth** is only active if **outspectra y** or if **ddxmode 1, 2** or **3**.

**Examples**

**elwidth 0.2**

**Range**

$1.e-6 \leq \text{elwidth} \leq 100$

**Default**

**elwidth 0.5**

**outangle**

Flag for the output of angular distributions for scattering to discrete states.

**Examples**

**outangle y**  
**outangle n**

**Range**

**y or n**

**Default**

**outangle n**

## **outlegendre**

Flag for the output of Legendre coefficients for the angular distributions for scattering to discrete states. **outlegendre** is only active if **outangle y**.

### **Examples**

**outlegendre y**

**outlegendre n**

### **Range**

**y** or **n**

### **Default**

**outlegendre n**

**outdiscrete**

Flag for the output of cross sections to each individual discrete state. This is given for both the direct and the compound component.

**Examples**

**outdiscrete y**

**outdiscrete n**

**Range**

y or n

**Default**

the same value as **outbasic**: **outdiscrete n**



**adddiscrete**

Flag for the addition of energy-broadened non-elastic cross sections for discrete states to the continuum spectra. **adddiscrete** is only active if **outspectra y**.

**Examples**

**adddiscrete y**

**adddiscrete n**

**Range**

y or n

**Default**

**adddiscrete y**

**addelastic**

Flag for the addition of energy-broadened elastic cross sections to the continuum spectra. This case is treated separately from **adddiscrete**, since sometimes the elastic contribution is already subtracted from the experimental spectrum. **addelastic** is only active if **outspectra y**.

**Examples**

**addelastic y**

**addelastic n**

**Range**

**y** or **n**

**Default**

the same value as **adddiscrete**: **addelastic y**

**coulomb**

Flag for Coulomb excitation calculation with ECIS, to be used for incident charged particles.

**Examples**

**coulomb y**

**Range**

**y or n**

**Default**

**coulomb y**

**cpang**

Flag for compound angular distribution calculation for incident charged particles.

**Examples**

**cpang y**

**Range**

**y or n**

**Default**

**cpang n**

**soswitch**

Energy switch to on-set deformed spin-orbit calculation and sequential iterations in ECIS. For coupled-channels calculations on rotational nuclei, such a switch needs to be made. On the input line, we read **soswitch**, value.

**Examples**

**soswitch 1.2**

**Range**

**$0.1 \leq \text{soswitch} \leq 10$ .**

**Default**

**soswitch 3.** MeV.

**statepot**

Flag for a different optical model parameterisation for each excited state in a DWBA or coupled-channels calculation. This may be appropriate if the emission energy of the ejectile, corresponding to a large excitation energy, differs considerably from the incident energy.

**Examples**

statepot y

**Range**

y or n

**Default**

statepot n

**outtransenergy**

Flag for the output of transmission coefficients sorted per energy (**y**) or per angular momentum (**n**). **outtransenergy** is only active if **outinverse y**.

**Examples**

**outtransenergy y**

**outtransenergy n**

**Range**

**y** or **n**

**Default**

**outtransenergy y**

**deformfile**

File with deformation parameters and coupling schemes. The format of the file is exactly the same as that of the nuclear structure database *talys/structure/deformation/*. In practice, the user can copy a file from this database, e.g. *Fe.def*, to the working directory and change it. In this way, changes in the “official” database are avoided. Note that even if only changes for one isotope are required, the entire file needs to be copied if for the other isotopes the originally tabulated values are to be used. On the input line, we read **deformfile**, *Z*, filename.

**Examples**

**deformfile 26 Fe.loc**

**Range**

**deformfile** can be equal to any filename, provided it starts with a character.

**Default**

If **deformfile** is not given in the input file, discrete levels are taken from *talys/structure/deformation*.



**core**

Integer to denote the even-even core for the weak-coupling model for direct scattering of odd-A nuclei. A value of -1 means the even-even core is determined by subtracting a nucleon from the target nucleus, while a value of +1 means a nucleon is added.

**Examples**

**core 1**

**Range**

**-1 or 1**

**Default**

**core -1**



## **29. Keywords for compound nucleus**

**compound**

Flag for compound nucleus calculation. This keyword can be used to disable compound nucleus evaporation if one is for example only interested in high-energy pre-equilibrium spectra.

**Examples**

**compound y**

**compound n**

**Range**

y or n

**Default**

compound y

## **widthfluc**

Enabling or disabling width fluctuation corrections (WFC) in compound nucleus calculations, see Section 13.2. For **widthfluc**, the user has 3 possibilities: **y**, **n** or a value for the energy above which WFC's are disabled. The latter option is helpful in the case of a calculation with several incident energies. Then, the user may want to set the width fluctuation off as soon as the incident energy is high enough, in order to save computing time. We have taken care of this by the default, **widthfluc=S**, where S is the projectile separation energy ( $\sim 8$  MeV), of the target nucleus. This default is rather safe, since in practice width fluctuation corrections are already negligible for incident energies above a few MeV, because the presence of many open channels reduces the correction to practically zero, i.e. the WFC factors to 1. Note that the disabling of width fluctuations for *any* incident energy can be accomplished by **widthfluc n**, which is equivalent to **widthfluc 0**. or any other energy lower than the (lowest) incident energy. Similarly, **widthfluc y**, equivalent to **widthfluc 20.**, will activate width fluctuations for any incident energy. To avoid numerical problems, width fluctuations are never calculated for incident energies beyond 20 MeV.

### **Examples**

**widthfluc y**  
**widthfluc n**  
**widthfluc 4.5**

### **Range**

**y** or **n** or **0.**  $\leq$  **widthfluc**  $<$  **20.**

### **Default**

**widthfluc** is equal to the projectile separation energy S, i.e. width fluctuation corrections are only used for incident energies below this value.

**widthmode**

Model for width fluctuation corrections in compound nucleus calculations, see Section 13.2.

**Examples**

**widthmode 0:** no width fluctuation, i.e. pure Hauser-Feshbach model

**widthmode 1:** Moldauer model

**widthmode 2:** Hofmann-Richert-Tepel-Weidenmüller model

**widthmode 3:** GOE triple integral model

**Range**

$0 \leq \text{widthmode} \leq 3$

**Default**

**widthmode 1**

**resonance**

Flag for variation of resonance parameters for an ENDF-6 general purpose file, by making a system call to the resonance code TARES.

(Only relevant if **#tmc y**.)

**Examples**

**#resonance n**

**#resonance y**

**Range**

**y** or **n**

**Default**

**#resonance y**

**urr**

Flag for the output of unresolved resonance parameters (URR). Since a full compound nucleus model and all its parameters are included in TALYS, it is only a small step to produce the URR parameters in the output. All calculated parameters, per incident energy, are stored in the file *urr.dat*. Energy-dependent tables can be found in the files *urrspacinglj.LLL*, with LL the orbital angular momentum in (i2.2) format, for the l,j-dependent neutron spacing, *urrneustrengthl.100* for the l-dependent neutron spacing, *urrneuwidth.LLL* for the neutron width, *urrgamwidth.LLL* for the gamma width, *urrfiswidth.LLL* for the fission width, and *urrcomwidth.LLL* for the competitive width, These can be used for evaluated nuclear data files.

**Examples**

**urr y**

**urr n**

**Range**

y or n

**Default**

**urr y**



**lurr**

Maximum orbital angular momentum taken into account for URR calculations. This keyword is only active if **urr y**.

**Examples**

**lurr 1**

**lurr 4**

**Range**

$0 \leq \text{lurr} \leq \text{numl}$ , where *numl* is specified in *talys.cmb* (currently  $\text{lurr}=60$ ).

**Default**

**lurr 2**

**urnjoy**

Flag for normalization of URR parameters with NJOY method. This keyword is only active if **urr y**. In addition to the files mentioned under the **urr** keyword, a few other files are produced: *urrtalys.tot*, the cross sections as calculated with the TALYS URR parameters, *urnjoy.tot*, the cross sections as calculated with the NJOY URR method, and *urrratio.tot*, containing the ratio between the aforementioned two.

**Examples**

**urnjoy y**  
**urnjoy n**

**Range**

y or n

**Default**

**urnjoy y**

---

**fullhf**

Flag for Hauser-Feshbach calculation using the full  $j,l$  coupling. This keyword can be used to enable/disable the loop over total angular momentum of the ejectile  $j'$  in Eq. (13.2). If **fullhf n**, the transmission coefficients are averaged over  $j$ , reducing the calculation time of the full Hauser-Feshbach model. In practice, the difference with the results from the full calculation is negligible.

**Examples****fullhf y****fullhf n****Range**

y or n

**Default****fullhf n**

**eciscompound**

Flag for compound nucleus calculation by ECIS-06, done in parallel with TALYS. This keyword is used for checking purposes only and does not influence the TALYS results. An ECIS input file is created that contains the same discrete levels, level density parameters etc., as the TALYS calculation. The compound nucleus results given by ECIS can be compared with the results from TALYS, but are not used in TALYS. The results are written on a separate ECIS output file.

**Examples**

**eciscompound y**  
**eciscompound n**

**Range**

y or n

**Default**

**eciscompound n**

**group**

Flag for processing pointwise resonance cross sections into energy groups. This keyword is only relevant if **resonance y**.

**Examples**

**group y**

**group n**

**Range**

**y or n**

**Default**

**group n**

**reslib**

Choice for the source of the resonance parameters. This keyword is only relevant if **resonance y**.

**Examples**

```
reslib default
reslib endfb7.1
reslib jendl4.0
reslib jeff3.2
```

**Range**

One of the 4 choices listed above.

**Default**

```
reslib default
```

**Tres**

Temperature, in Kelvin, for the broadening of resonances. This keyword is only relevant if resonance y.

**Examples**

**Tres 600.**

**Tres 0.**

**Range**

**0. < Tres < 1.e12**

**Default**

**Tres 293.16**

**skipCN**

Flag to skip the decay of a compound nucleus altogether. In this way, for example (n, $\gamma$ n) reactions can be skipped. Furthermore, the flag can be used for debugging purposes or other te

**Examples**

**skipCN 40 91**

**skipCN 92 239**

**Range**

**skipCN Z A.**

**Default**

**skipCN** not used.



**xscaptherm**

The thermal capture cross section in millibarn. By default, these are read from the nuclear structure database or taken from systematics. The **xscaptherm** keyword gives the possibility to overwrite this by using the input file.

**Examples**

**xscaptherm 320.2**

**Range**

$10^{-20} \leq \text{xscaptherm} \leq 10^{10}$

**Default**

**xscaptherm** is read from the nuclear structure database or taken from systematics.

**xalphatherm**

The thermal ( $n, \alpha$ ) cross section in millibarn. By default, these are read from the nuclear structure database or taken from systematics. The **xalphatherm** keyword gives the possibility to overwrite this by using the input file.

**Examples**

**xalphatherm 0.2**

**Range**

$10^{-20} \leq \text{xalphatherm} \leq 10^{10}$

**Default**

**xalphatherm** is read from the nuclear structure database or taken from systematics.

**xsptherm**

The thermal (n,p) cross section in millibarn. By default, these are read from the nuclear structure database or taken from systematics. The **xsptherm** keyword gives the possibility to overwrite this by using the input file.

**Examples**

**xsptherm 0.2**

**Range**

$10^{-20} \leq \text{xsptherm} \leq 10^{10}$

**Default**

**xsptherm** is read from the nuclear structure database or taken from systematics.



## **30. Keywords for pre-equilibrium reactions**

### preequilibrium

Enabling or disabling the pre-equilibrium reaction mechanism. For **preequilibrium**, the user has 3 possibilities: **y**, **n** or a value for the starting energy. The latter option is helpful in the case of a calculation with several incident energies. Then, the user may want to set pre-equilibrium contributions on as soon as the incident energy is high enough. We have taken care of this by the default, **preequilibrium=Ex(Nm)**, where  $Ex(Nm)$  is the excitation energy of the last discrete level  $Nm$  of the target nucleus. This default is very safe, since in practice the pre-equilibrium contribution becomes only sizable for incident energies several MeV higher than  $Ex(Nm)$ . Note that the disabling of pre-equilibrium for *any* incident energy can be accomplished by **preequilibrium n**. Similarly, **preequilibrium y**, equivalent to **preequilibrium 0.**, will enable pre-equilibrium for any incident energy.

### Examples

**preequilibrium y**  
**preequilibrium n**  
**preequilibrium 4.5**

### Range

**y** or **n** or **0.**  $\leq$  **preequilibrium**  $<$  **250.**

### Default

**preequilibrium** is equal to  $Ex(NL)$ , i.e. pre-equilibrium calculations are included for incident energies above the energy of the last discrete level of the target nucleus.

---

## **preeqmode**

Model for pre-equilibrium reactions. There are four possibilities, see Section 12.

### **Examples**

**preeqmode 1:** Exciton model: Analytical transition rates with energy-dependent matrix element.

**preeqmode 2:** Exciton model: Numerical transition rates with energy-dependent matrix element.

**preeqmode 3:** Exciton model: Numerical transition rates with optical model for collision probability.

**preeqmode 4:** Multi-step direct/compound model

### **Range**

$1 \leq \text{preeqmode} \leq 4$

### **Default**

**preeqmode 2**

## **multipreeq**

Enabling or disabling multiple pre-equilibrium reaction mechanism. For **multipreeq**, the user has 3 possibilities: **y**, **n** or a value for the starting energy. The latter option is helpful in the case of a calculation with several incident energies. Then, the user may want to set multiple pre-equilibrium contributions on as soon as the incident energy is high enough. We have taken care of this by the default, **multipreeq 20.** This default is very safe, since in practice the multiple pre-equilibrium contribution becomes only sizable for incident energies a few tens of MeV higher than the default. Note that the disabling of multiple pre-equilibrium for *any* incident energy can be accomplished by **multipreeq n**. Similarly, **multipreeq y**, equivalent to **multipreeq 0.**, will activate multiple pre-equilibrium for any incident energy.

### **Examples**

**multipreeq y**  
**multipreeq n**  
**multipreeq 40.**

### **Range**

**y** or **n** or **0.**  $\leq$  **multipreeq**  $<$  **250.**

### **Default**

**multipreeq 20.**, i.e. multiple pre-equilibrium calculations are included for incident energies above this value. TALYS always sets **multipreeq n** if **preequilibrium n**.



**mpreeqmode**

Model for multiple pre-equilibrium reactions. There are two possibilities, see Section 14.2 for an explanation.

**Examples**

**mpreeqmode 1:** Multiple exciton model

**mpreeqmode 2:** Transmission coefficient method

**Range**

$1 \leq \text{mpreeqmode} \leq 2$

**Default**

**mpreeqmode 2**

**preeqspin**

Flag to use the pre-equilibrium or compound nucleus spin distribution for the pre-equilibrium population of the residual nuclides. For backward-compatibility with earlier versions of TALYS, the following options are now possible:

**Examples**

**preeqspin n** or **preeqspin 1**: the pre-equilibrium spin distribution is made equal to the relative spin-dependent population after compound nucleus emission

**preeqspin 2**: the spin distribution from total level densities is adopted

**preeqspin y** or **preeqspin 3**: the pre-equilibrium spin distribution is based on particle-hole state densities

**Range**

y or n, or  $1 \leq \text{preeqspin} \leq 3$ .

**Default**

**preeqspin n**

**preeqsurface**

Flag to use surface corrections in the exciton model.

**Examples**

**preeqsurface y**  
**preeqsurface n**

**Range**

**y or n**

**Default**

**preeqsurface y**

**outpreequilibrium**

Flag for the output of pre-equilibrium parameters and cross sections. **outpreequilibrium** is only active if **preequilibrium y**.

**Examples**

**outpreequilibrium y**

**outpreequilibrium n**

**Range**

y or n

**Default**

**outpreequilibrium n**

**Esurf**

Effective well depth for surface effects in MeV in the exciton model, see Eq. (12.11).

**Examples**

**Esurf** 25.

**Range**

$0. \leq \mathbf{Esurf} \leq \mathbf{Efermi}$ , where **Efermi** = 38 MeV is the Fermi well depth.

**Default**

**Esurf** is given by Eq. (12.11).

**preeqcomplex**

Flag to use the Kalbach model for pickup, stripping and knockout reactions, in addition to the exciton model, in the pre-equilibrium region.

**Examples**

**preeqcomplex y**

**preeqcomplex n**

**Range**

y or n

**Default**

**preeqcomplex y**

**twocomponent**

Flag to use the two-component (y) or one-component (n) exciton model.

**Examples**

**twocomponent y**

**twocomponent n**

**Range**

**y or n**

**Default**

**twocomponent y**

**pairmodel**

Model for pairing correction for pre-equilibrium model.

**Examples**

**pairmodel 1:** Fu's pairing energy correction, see Eq. (12.5).

**pairmodel 2:** Compound nucleus pairing correction

**Range**

$1 \leq \text{pairmodel} \leq 2$

**Default**

**pairmodel 1**



**phmodel**

Model for particle-hole state densities. There are two possibilities.

**Examples**

**phmodel 1:** Phenomenological particle-hole state densities

**phmodel 2:** Microscopic particle-hole state densities

**Range**

$1 \leq \text{phmodel} \leq 2$

**Default**

phmodel 1

**Kph**

Value for the constant of the single-particle level density parameter, i.e.  $g = A/K_{ph}$ , or  $g_{\pi} = Z/K_{ph}$  and  $g_{\nu} = N/K_{ph}$

**Examples**

**Kph 12.5**

**Range**

**$1. \leq K_{ph} \leq 100.$**

**Default**

**Kph 15.**

**g**

The single-particle level density parameter  $g$  in  $\text{MeV}^{-1}$ . On the input line, we read  $g$ ,  $Z$ ,  $A$ , value.

**Examples**

**g 41 93 7.15**

**g 94 239 17.5**

**Range**

**$0.1 \leq g \leq 100$ .**

**Default**

$g = A/Kph$

**gp**

The single-particle proton level density parameter  $g_{\pi}$  in  $\text{MeV}^{-1}$ . On the input line, we read **gp**,  $Z$ ,  $A$ , value.

**Examples**

**gp 41 93 3.15**

**gp 94 239 7.2**

**Range**

**$0.1 \leq \text{gp} \leq 100$ .**

**Default**

$g_p = Z/Kph$

**gn**

The single-particle neutron level density parameter  $g_v$  in  $\text{MeV}^{-1}$ . On the input line, we read **gn**,  $Z$ ,  $A$ , value.

**Examples**

**gn 41 93 4.1**

**gn 94 239 11.021**

**Range**

**$0.1 \leq \text{gn} \leq 100$ .**

**Default**

$g_n = N/Kph$

**gpadjust**

Multiplier to adjust the partial level density parameter  $g_\pi$ . On the input line, we read **gpadjust**, Z, A and value.

**Examples**

```
gpadjust 41 93 1.04
```

**Range**

$0.5 \leq \text{gpadjust} \leq 2.$

**Default**

**gpadjust** Z A 1.

**Comments**

- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

## **gnadjust**

Multiplier to adjust the partial level density parameter  $g_v$ . On the input line, we read **gnadjust**, Z, A and value.

### **Examples**

```
gnadjust 41 93 1.04
```

### **Range**

$0.5 \leq \text{gnadjust} \leq 2.$

### **Default**

**gnadjust Z A 1.**

### **Comments**

- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

**gadjust**

Multiplier to adjust the partial level density parameter  $g$ . On the input line, we read **gadjust**,  $Z$ ,  $A$  and value.

**Examples**

**gadjust 41 93 1.04**

**Range**

$0.5 \leq \text{gadjust} \leq 2$ .

**Default**

**gadjust Z A 1.**

**Comments**

- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.



**gshell**

Flag to include the damping of shell effects with excitation energy in single-particle level densities. The Ignatyuk parameterisation for total level densities is also applied to the single-particle level density parameters.

**Examples**

**gshell y**

**gshell n**

**Range**

**y or n**

**Default**

**gshell n**

**M2constant**

Overall constant for the matrix element, or the optical model strength, in the exciton model. The parameterisation of the matrix element is given by Eq. (12.46) for the one-component model, and by Eq. (12.26) for the two-component model. **M2constant** is also used to scale the MSD cross section (preeqmode 4).

**Examples**

**M2constant 1.22**

**Range**

$0. \leq \text{M2constant} \leq 100.$

**Default**

**M2constant 1.**

**M2limit**

Constant to scale the asymptotic value of the matrix element in the exciton model. The parameterisation of the matrix element is given by Eq. (12.46) for the one-component model, and by Eq. (12.26) for the two-component model.

**Examples**

**M2limit 1.22**

**Range**

**$0. \leq \text{M2limit} \leq 100.$**

**Default**

**M2limit 1.**

**M2shift**

Constant to scale the energy shift of the matrix element in the exciton model. The parameterisation of the matrix element is given by Eq. (12.46) for the one-component model, and by Eq. (12.26) for the two-component model.

**Examples**

**M2shift 1.22**

**Range**

**$0. \leq \text{M2shift} \leq 100.$**

**Default**

**M2shift 1.**

**Rnnu**

Neutron-neutron ratio for the matrix element in the two-component exciton model, see Eq. (12.24).

**Examples**

**Rnnu 1.6**

**Range**

**$0. \leq \text{Rnnu} \leq 100.$**

**Default**

**Rnnu 1.5**

**Rnupi**

Neutron-proton ratio for the matrix element in the two-component exciton model, see Eq. (12.24).

**Examples**

**Rnupi 1.6**

**Range**

**$0. \leq \text{Rnupi} \leq 100.$**

**Default**

**Rnupi 1.**

**Rpipi**

Proton-proton ratio for the matrix element in the two-component exciton model, see Eq. (12.24).

**Examples**

**Rpipi 1.6**

**Range**

**$0. \leq \text{Rpipi} \leq 100.$**

**Default**

**Rpipi 1.**

**Rpinu**

Proton-neutron ratio for the matrix element in the two-component exciton model, see Eq. (12.24).

**Examples**

**Rpinu 1.6**

**Range**

**$0. \leq \text{Rpinu} \leq 100.$**

**Default**

**Rpinu 1.**



**Rgamma**

Adjustable parameter for pre-equilibrium gamma decay.

**Examples**

**Rgamma 1.22**

**Range**

**$0. \leq \text{Rgamma} \leq 100.$**

**Default**

**Rgamma 2.**

**Cstrip**

Adjustable parameter for the stripping or pick-up process, to scale the complex-particle pre-equilibrium cross section per outgoing particle, see Section 12.4. On the input line, we read **Cstrip**, particle symbol, and value.

**Examples**

**Cstrip d 1.3**

**Cstrip a 0.4**

**Range**

$0. \leq \text{Cstrip} \leq 10.$

**Default**

**Cstrip 1.**

## **Cknock**

Adjustable parameter for the knock-out process, to scale the complex-particle pre-equilibrium cross section per outgoing particle, see Section 12.4. In practice, for nucleon-induced reactions this parameter affects only alpha-particles. This parameter is however also used as scaling factor for break-up reactions (such as (d,p) and (d,n)). On the input line, we read **Cknock**, particle symbol, and value.

### **Examples**

**Cknock a 0.4**

### **Range**

**$0. \leq \text{Cknock} \leq 10.$**

### **Default**

**Cknock 1.**

**Cbreak**

Adjustable parameter for the break-up process, to scale the complex-particle pre-equilibrium cross section per outgoing particle, see Section 12.4. On the input line, we read **Cbreak**, particle symbol, and value.

**Examples**

**Cbreak d 1.3**

**Cbreak a 0.4**

**Range**

$0. \leq \text{Cbreak} \leq 10.$

**Default**

**Cbreak 1.**

**breakupmodel**

Model for break-up reactions.

**Examples**

**breakupmodel 1:** Kalbach model

**breakupmodel 2:** Avrigeanu model

**Range**

$1 \leq \text{breakupmodel} \leq 2$

**Default**

**breakupmodel 1**

**ecisdwba**

Flag for DWBA calculations for multi-step direct calculations. If this calculation has already been performed in a previous run, it may be helpful to put **ecisdwba n**, which avoids a new calculation and thus saves time. We stress that it is the responsibility of the user to ensure that the first run of a particular problem is done with **ecisdwba y**. If not, an appropriate error message will be given and TALYS stops.

**Examples**

**ecisdwba y**  
**ecisdwba n**

**Range**

y or n

**Default**

**ecisdwba y**

**outdwba**

Flag for the output of DWBA cross sections for the multi-step direct model.

**Examples**

outdwba y  
outdwba n

**Range**

y or n

**Default**

outdwba n

**onestep**

Flag for inclusion of *only* the one-step direct contribution in the continuum multi-step direct model. This is generally enough for incident energies up to about 14 MeV, and thus saves computing time.

**Examples**

`onestep y`

`onestep n`

**Range**

y or n

**Default**

`onestep n`



**msdbins**

The number of emission energy points for the DWBA calculation for the multi-step direct model.

**Examples**

**msdbins 8**

**Range**

$2 \leq \text{msdbins} \leq \text{numenmsd}/2-1$ , where **numenmsd** is specified in the file *talys.cmb*. Currently, **numenmsd=18**

**Default**

**msdbins 6**

**Emsdmin**

The minimal emission energy in MeV for the multi-step direct calculation.

**Examples**

**Emsdmin 8.**

**Range**

**$0. \leq \text{Emsdmin}$**

**Default**

**Emsdmin** is equal to **eninc/5**, where **eninc** is the incident energy.

## **31. Keywords for fission**

**fission**

Flag for enabling or disabling fission. By default **fission** is enabled if the target mass is above 209. Hence, for lower masses, it is necessary to set **fission y** manually at high incident energies (subactinide fission).

**Examples**

**fission y**

**fission n**

**Range**

**y** or **n**. Fission is not allowed for  $A \leq 56$

**Default**

**fission y** for  $A > 209$ , **fission n** for  $A \leq 209$ . The default enabling or disabling of fission is thus mass dependent.

---

**fismodel**

Model for fission barriers. **fismodel** is only active if **fission y**. There are 5 possibilities:

**Examples**

**fismodel 1**: “experimental” fission barriers

**fismodel 2**: theoretical fission barriers, Mamdouh table

**fismodel 3**: theoretical fission barriers, Sierk model

**fismodel 4**: theoretical fission barriers, rotating liquid drop

**fismodel 5**: WKB approximation for fission path model

**Range**

$1 \leq \text{fismodel} \leq 5$

**Default**

**fismodel 1**

**fismodelalt**

"Back-up" model for fission barriers, for the case that the parameters of the tables used in **fismodel 1-2** are not available. There are two possibilities:

**Examples**

**fismodelalt 3** : theoretical fission barriers, Sierk model

**fismodelalt 4** : theoretical fission barriers, rotating liquid drop model

**Range**

$3 \leq \text{fismodelalt} \leq 4$

**Default**

**fismodelalt 4**

## **axtype**

Type of axiality of the fission barrier. There are five options:

- 1: axial symmetry
- 2: left-right asymmetry
- 3: triaxial and left-right asymmetry
- 4: triaxial no left-right asymmetry
- 5: no symmetry

On the input line, we read **axtype**,  $Z$ ,  $A$ , value, fission barrier. If the number of the fission barrier is not given or is equal to 0, it concerns the first barrier.

## **Examples**

**axtype 90 232 3**

**axtype 94 239 1 2**

## **Range**

$1 \leq \text{axtype} \leq 5$

## **Default**

**axtype 2** for the second barrier and  $N > 144$ , **axtype 3** for the first barrier and  $N > 144$ , **axtype 1** for the rest.

**fisbar**

Fission barrier in MeV. On the input line, we read **fisbar**, Z, A, value, fission barrier. This keyword overrules the value given in the nuclear structure database. If the number of the fission barrier is not given or is equal to 0, it concerns the first barrier.

**Examples**

**fisbar 90 232 5.6**

**fisbar 94 239 6.1 2**

**Range**

$0. \leq \text{fisbar} \leq 100.$

**Default**

**fisbar** is read from the *talys/structure/fission/* directory, or determined by systematics according to the choice of **fismodel**.



---

**fishw**

Fission barrier width in MeV. On the input line, we read **fishw**, *Z*, *A*, value, fission barrier. This keyword overrules the value given in the nuclear structure database. If the number of the fission barrier is not given or is equal to 0, it concerns the first barrier.

**Examples**

**fishw 90 232 0.8**

**fishw 94 239 1.1 2**

**Range**

$0.01 \leq \text{fishw} \leq 10$ .

**Default**

**fishw** is read from the *talys/structure/fission/* directory or determined by systematics, according to the choice of **fismodel**.

**fisbaradjust**

Normalisation factor for the fission barrier. This parameter can be used as a relative normalisation instead of the absolute value of **fisbar**. On the input line, we read **fisbaradjust**, *Z*, *A*, value, fission barrier. If the number of the fission barrier is not given or is equal to 0, it concerns the first barrier.

**Examples**

```
fisbaradjust 90 232 0.9
```

```
fisbaradjust 94 239 1.13 2
```

**Range**

$0.1 \leq \text{fisbaradjust} \leq 10$ .

**Default**

**fisbaradjust** *Z A* 1.

**Comments**

- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

## **fishwadjust**

Normalisation factor for the fission barrier width. This parameter can be used as a relative normalisation instead of the absolute value of **fishw**. On the input line, we read **fishwadjust**, *Z*, *A*, value, fission barrier. If the number of the fission barrier is not given or is equal to 0, it concerns the first barrier.

### **Examples**

```
fishwadjust 90 232 0.9
fishwadjust 94 239 1.13 2
```

### **Range**

$0.1 \leq \text{fishwadjust} \leq 10.$

### **Default**

**fishwadjust** *Z A* 1..

### **Comments**

- As a general comment for '**adjust**' keywords: this avoids having to look up the actual parameter value in the TALYS input file, hardwired default value in the source code or this tutorial, or one of the data tables that TALYS uses. One simply multiplies the nominal value with the value given on this keyword line without having to know the actual value.

**bdamp**

The fission damping parameter. On the input line, we read **bdamp**,  $Z$ ,  $A$ , value, fission barrier. If the number of the fission barrier is not given or is equal to 0, it concerns the first barrier.

**Examples**

**bdamp 90 232 0.9**

**bdamp 94 239 1.13 2**

**Range**

$0. \leq \text{bdamp} \leq 50.$

**Default**

**bdamp  $Z A 1.$**

## **bdampadjust**

Normalisation factor for the fission damping parameter. This parameter can be used as a relative normalisation instead of the absolute value of **bdamp**. On the input line, we read **bdampadjust**, *Z*, *A*, value, fission barrier. If the number of the fission barrier is not given or is equal to 0, it concerns the first barrier.

### **Examples**

```
bdampadjust 90 232 0.9
```

```
bdampadjust 94 239 1.13 2
```

### **Range**

**$0.01 \leq \text{bdampadjust} \leq 100$ .**

### **Default**

**bdampadjust *Z A* 1.**

**fispartdamp**

Flag for partial damping in the fission model.

**Examples**

```
fispartdamp y  
fispartdamp n
```

**Range**

**y** or **n**. **fispartdamp** can only be **y** for **fismodel 5**.

**Default**

**fispartdamp n**.

**hbstate**

Flag to use head band states in fission.

**Examples**

**hbstate y**  
**hbstate n**

**Range**

**y** or **n**.

**Default**

**hbstate y**.

### **hbtransfile**

File with head band transition states. The format of the file is exactly the same as that of the nuclear structure database *talys/structure/fission/barrier/*. In practice, the user can copy a file from this database, e.g. *hbstates.eo*, to the working directory and change it. In this way, changes in the “official” database are avoided. Note that one file in the working directory can only be used for one isotope. On the input line, we read **hbtransfile**, *Z, A, filename*.

### **Examples**

```
hbtransfile 92 238 u238.hb
```

### **Range**

**hbtransfile** can be equal to any filename, provided it starts with a character.

### **Default**

If **hbtransfile** is not given in the input file, the head band transition states are taken from the *talys/structure/fission/states* database.



**class2**

Flag for the enabling or disabling of class II/III states in fission. **class2** is only active if **fission y**.

**Examples**

**class2 y**  
**class2 n**

**Range**

**y** or **n**

**Default**

**class2 y**

**class2width**

Width of class II/III states. On the input line, we read **class2width**, *Z*, *A*, value, fission barrier. If the number of the fission barrier is not given or is equal to 0, it concerns the first barrier well.

**Examples**

**class2width 90 232 0.35**

**class2width 94 239 0.15 2**

**Range**

**$0.01 \leq \text{class2width} \leq 10$ .**

**Default**

**class2width *Z A* 0.2**

---

## **class2file**

File with class II/III transition states. The format of the file is exactly the same as that of the nuclear structure database *talys/structure/fission/states/*. In practice, the user can copy a file from this database, e.g. *class2states.eo*, to the working directory and change it. In this way, changes in the “official” database are avoided. Note that one file in the working directory can only be used for one isotope. On the input line, we read **class2file**, *Z*, *A*, filename.

## **Examples**

```
class2file 92 238 u238.c2
```

## **Range**

**class2file** can be equal to any filename, provided it starts with a character.

## **Default**

If **class2file** is not given in the input file, the head band transition states are taken from the *talys/structure/fission/states* database.

**betafiscor**

Factor to adjust the width of the WKB fission path. (only applies for **fismodel 5**). On the input line, we read **betafiscor**, *Z*, *A*, value.

**Examples**

**betafiscor 92 239 1.2**

**Range**

**$0.1 \leq \text{betafiscor} \leq 10$ .**

**Default**

**betafiscor *Z A* 1.**

**vfiscor**

Factor to adjust the height of the WKB fission path. (only applies for **fismodel 5**). On the input line, we read **vfiscor**, *Z*, *A*, value.

**Examples**

**vfiscor 92 239 0.9**

**Range**

$0.1 \leq \text{vfiscor} \leq 10$ .

**Default**

**vfiscor Z A 1.**

**betafiscoradjust**

Adjustable factor for betafiscor, to adjust the width of the WKB fission path. (only applies for **fismodel 5**). On the input line, we read **betafiscoradjust**, *Z*, *A*, value.

**Examples**

**betafiscoradjust 92 239 1.2**

**Range**

$0.1 \leq \text{betafiscoradjust} \leq 10$ .

**Default**

**betafiscoradjust Z A 1.**

**vfiscoradjust**

Adjustable factor for vfiscor, to adjust the height of the WKB fission path. (only applies for fismodel 5). On the input line, we read **vfiscoradjust**, Z, A, value.

**Examples**

**vfiscoradjust 92 239 0.9**

**Range**

**$0.1 \leq \text{vfiscoradjust} \leq 10$ .**

**Default**

**vfiscoradjust Z A 1.**

**Rfiseps**

Ratio for limit for fission cross section per nucleus. This parameter determines whether the mass distribution for a residual fissioning nucleus will be calculated. Cross sections smaller than **Rfiseps** times the fission cross section are not used in the calculations, in order to reduce the computation time.

**Examples**

**Rfiseps 1.e-5**

**Range**

$0. \leq \text{Rfiseps} \leq 1.$

**Default**

**Rfiseps 1.e-3**



---

## **fymodel**

Model for the calculation of fission yields.

### **Examples**

**fymodel 1:** Brosa model (the only option until TALYS-1.4)

**fymodel 2:** GEF model from Schmidt-Jurado [130]

**fymodel 3:** GEF model from Schmidt-Jurado [130], but evaporation of fission fragments done by Hauser-Feshbach model

**fymodel 4:** Okumura model - read in yields and excitation energies [149]

**fymodel 5:** General model - read in full population per fission fragment [149]

### **Range**

$1 \leq \text{fymodel} \leq 5$

### **Default**

**fymodel 2**

**ffmodel**

Model for the distribution of fission fragments.

**Examples**

**ffmodel 1:** GEF model , fission fragments generated by Ali Al-Adili and Fredrik Nordstroem.

**ffmodel 2:** Okumura model [149]

**ffmodel 3:** SPY model [[Lemaitre2019](#)]

**Range**

$1 \leq \text{ffmodel} \leq 3$

**Default**

**ffmodel 1**

**pfnsmodel**

Model for prompt fission neutron spectra.

**Examples**

**pfnsmodel 1:** Iwamoto model (2007)

**pfnsmodel 2:** Explicit decay from fission fragments

**Range**

$1 \leq \text{pfnsmodel} \leq 2$

**Default**

**pfnsmodel 2** if **massdis y**, **pfnsmodel 1** otherwise.

**Fsadjust**

Adjustable factor for the number of scission neutrons, for correction of prompt fission neutron spectrum. This works for the Iwamoto model for PFNS, **pfnsmodel 1**. On the input line, we read **Fsadjust**, value.

**Examples**

**Fsadjust 1.2**

**Range**

**$0.1 \leq \text{Fsadjust} \leq 10$ .**

**Default**

**Fsadjust 1.**

**Tmadjust**

Adjustable factor for the temperature of prompt fission neutron spectrum. This works for the Iwamoto model for PFNS, **pfnsmodel 1**. On the input line, we read **Tmadjust**, value.

**Examples**

**Tmadjust 1.2**

**Range**

**$0.1 \leq \text{Tmadjust} \leq 10$ .**

**Default**

**Tmadjust 1.**

**outfy**

Detailed output of fission yields from each excited bin.

**Examples**

**outfy y**

**outfy n**

**Range**

**y or n**

**Default**

**outfy n**

**ffspin**

Flag to enable a choice between spin dependent excitation functions for excited fission fragments as emerging from GEF, or assignment of spin distributions by TALYS. This only applies to **fymodel 3**.

**Examples**

**ffspin y**

**ffspin n**

**Range**

**y** or **n**

**Default**

**ffspin n**

**gefran**

Number of random samples for GEF calculation.

**Examples**

**gefran 18000**

**Range**

**$1000 \leq \text{gefran} \leq 1000000$**

**Default**

**gefran 50000**



**Cnubar1**

Adjustable parameter for the energy-dependent constant of nubar from Wahl's systematics. On the input line, we read **Cnubar1** and value.

**Examples**

**Cnubar1 1.3**

**Cnubar1 0.4**

**Range**

**$0.1 \leq \text{Cnubar1} \leq 10$ .**

**Default**

**Cnubar1 1.**

**Cnubar2**

Adjustable parameter for the energy-dependent constant of nubar from Wahl's systematics. On the input line, we read **Cnubar2** and value.

**Examples**

**Cnubar2 1.3**

**Cnubar2 0.4**

**Range**

**$0.1 \leq \text{Cnubar2} \leq 10.$**

**Default**

**Cnubar2 1.**

**massdis**

Flag for the calculation of the fission-fragment mass distribution.

**Examples**

`massdis y`

`massdis n`

**Range**

`y` or `n`

**Default**

`massdis n`

**outfission**

Flag for the output of fission parameters, transmission coefficients and partial cross sections. **outfission** is only active if **fission y**.

**Examples**

**outfission y**

**outfission n**

**Range**

y or n

**Default**

the same value as **outbasic**: **outfission n**

---

**filefission**

Flag to write all the fission cross sections as a function of incident energy on a separate file *fission.tot*. The file contains the incident energy and the total fission cross section. If in addition *filechannels* **y**, the exclusive fission cross sections will be written to files *fisNPDTHA.tot*, where N is the neutron number of the exclusive channel, P the proton number, etc., in (a1) format. For example, *fis200000.tot* contains the excitation function for  $\sigma(n, 2nf)$ , also known as the third chance fission cross section. **filefission** is only active if **fission** **y**.

**Examples****filefission** **y****filefission** **n****Range****y** or **n****Default****filefission** **n**

**ffevaporation**

Flag to enable phenomenological correction for evaporated neutrons from fission fragments with the Brosa model.

**Examples**

**ffevaporation y**

**ffevaporation n**

**Range**

y or n

**Default**

**ffevaporation n**

**yieldfile**

File with fission fragment yields per pair, and their respective TXE and TKE values, to be used for evaporation using TALYS. The format of the file is as used by Okumura et al. [149] On the input line, we read **yieldfile**, filename.

**Examples**

**yieldfile** fy.dat

**Range**

**yieldfile** can be equal to any filename, provided it starts with a character.

**Default**

If **yieldfile** is not given in the input.

**fisfeed**

Flag for the output of the fission contribution per excitation energy bin. This allows to couple TALYS with e.g. fission yield software, by using the feeding per bin as input for normalization. The associated files have the name *fisZZZAAA.nex* where *ZZZ* is the charge number and *AAA* is the mass number in (i3.3) format.

**Examples**

**fisfeed y**  
**fisfeed n**

**Range**

y or n

**Default**

**fisfeed n**



## **32. Keywords for astrophysics**

**astro**

Flag for the calculation of thermonuclear reaction rates for astrophysics, see Section 18.

**Examples**

**astro y**

**astro n**

**Range**

**y or n**

**Default**

**astro n**

**astrog**

Flag for treating the target in the ground state only, for astrophysical reaction calculations. In the default case, **astrog n**, an average between excited target states will be made. This keyword is only active if **astro y**.

**Examples**

**astrog y**

**astrog n**

**Range**

**y** or **n**

**Default**

**astrog n**

**astroE**

Energy, in MeV, for thermonuclear reactions around which Maxwellian averaging should take place. This is only active if **astro y**. Actually, specifying this value will automatically activate **astro y** and **astrogs y**. Hence, if you require the Maxwellian averaged 30 keV capture cross section, all you need to give is the input line below. **astroE** may not be specified simultaneously with **astroT9**. The **energy** keyword must still be given in the TALYS input file, but becomes irrelevant since it concerns an average over a large number of incident energies.

**Examples**

**astroE 0.03**

**Range**

$0.00001 \leq \text{astroE} \leq 1.$

**Default**

**astroE 0.**

---

**astroT**

Temperature, in units of  $10^9$  K, for thermonuclear reactions and Maxwellian averaging. This is only active if **astro y**. Actually, specifying this value will automatically activate **astro y**. **astroT** may not be specified simultaneously with **astroE**. The **energy** keyword must still be given in the TALYS input file, but becomes irrelevant since it concerns an average over a large number of incident energies.

**Examples**

**astroT 0.53**

**Range**

**$0.0001 \leq \text{astroT} \leq 10.$**

**Default**

**astroT 0.**

**astroex**

Flag for calculation of astrophysics reaction rate to final long-lived excited states. This extends the calculation of the rate for the ground state only. This keyword is only active if **astro y**.

**Examples**

**astroex y**

**astroex n**

**Range**

**y or n**

**Default**

**astroex n**

**nonthermlev**

Non-thermalized level in the calculation of astrophysics rate.

**Examples**

**nonthermlev 2**

**Range**

$0 \leq \text{nonthermlev} \leq \text{numlev}$ , where **numlev** is specified in the file *talys.cmb*.

**Default**

**nonthermlev** is not used.





## **33. Keywords for medical isotope production**

**production**

Flag to run TASMANT for covariances or optimization of isotope production.

**Examples**

```
#production n  
#production y
```

**Range**

y or n

**Default**

```
#production n
```

**Ebeam**

The incident energy of the particle beam in MeV. This keyword is only active if **production y**.

**Examples**

**Ebeam 140.**

**Ebeam 16.**

**Range**

$10^{-11}$  MeV  $\leq$  **Ebeam** < **E<sub>max</sub>** MeV, where **E<sub>max</sub>** is the highest incident energy.

**Default**

None, **Ebeam** must be given if **production y**.

**Eback**

The lower end of the energy range at the back end of the target in MeV. This energy degradation is directly related to the effective thickness of the target. This keyword is only active if **production y**.

**Examples**

**Eback 130.**

**Eback 12.**

**Range**

$E_{\min} \leq \mathbf{Eback} < \mathbf{Ebeam}$ , where  $E_{\min}$  is the lowest incident energy.

**Default**

**Eback = Ebeam - 5 MeV.**

**Ibeam**

Particle beam current in mA. This keyword is only active if **production y**.

**Examples**

**Ibeam 0.1**

**Ibeam 5.**

**Range**

**0 <= Ibeam <= 10000.**

**Default**

**Ibeam 1. mA.**

**Area**

Area of the target in cm<sup>2</sup>. This keyword is only active if **production y**.

**Examples**

Area 1.

**Range**

0 <= Area <= 10000.

**Default**

Area 10. cm<sup>2</sup>.

**rho**

Material density of the target in g/cm<sup>3</sup>. This keyword is only active if **production y**.

**Examples**

**rho 10.**

**Range**

**0 <= rho <= 100.**

**Default**

**rho 1.** is read from a hardwired material density table.

**radiounit**

Unit for radioactivity, to be used in the output files for isotope production. This keyword is only active if **production y**.

*The following units are possible:*

**radiounit Bq:** Becquerel

**radiounit kBq:** kiloBecquerel

**radiounit MBq:** MegaBecquerel

**radiounit GBq:** GigaBecquerel

**radiounit Ci:** Curie

**radiounit mCi:** milliCurie

**radiounit kCi:** kiloCurie

**Range**

**radiounit** should be equal to one of the units above.

**Default**

**radiounit Mbq.**



**yieldunit**

Unit for isotope yield, in weight or number of isotopes. This keyword is only active if **production y**.

*The following units are possible:*

**yieldunit mug**: microgram

**yieldunit mg**: milligram

**yieldunit g**: gram

**yieldunit kg**: kilogram

**yieldunit num**: number of isotopes

**Range**

**yieldunit** should be equal to one of the units above.

**Default**

**yieldunit num**.

**Tirrad**

Irradiation time. A general input for the irradiation time has been enabled. On the input line we read integer values and time units, which can be y (years), d (days), h (hours), m (minutes) or s (seconds). These all need to be separated by blanks. This keyword is only active if **production y**.

**Examples**

```
Tirrad    2 d 5 h
Tirrad    32 h 30 m
Tirrad    1 d 6 h 24 m 12 s
```

**Range**

$0 \leq \text{Tirrad} \leq 1e6$  for every time unit.

**Default**

**Tirrad 1 d.**

## Tcool

Target cooling time. A general input for the cooling time has been enabled. On the input line we read integer values and time units, which can be y (years), d (days), h (hours), m (minutes) or s (seconds). These all need to be separated by blanks. This keyword is only active if **production y**.

### Examples

```
Tcool    2 d 5 h
Tcool    32 h 30 m
Tcool    1 d 6 m 24 m 12 s
```

### Range

$0 \leq \text{Tcool} \leq 1\text{e}6$  for every time unit.

### Default

Tcool 1 d.



## **34. Keywords for output**

**outmain**

Flag for the main output. The header of TALYS is printed, together with the input variables and the automatically adopted default values. Also the most important computed cross sections are printed.

**Examples**

```
outmain y
outmain n
```

**Range**

y or n

**Default**

```
outmain y
```

---

## outbasic

Flag for the output of *all* basic information needed for the nuclear reaction calculation, such as level/bin populations, numerical checks, optical model parameters, transmission coefficients, inverse reaction cross sections, gamma and fission information, discrete levels and level densities. If **outbasic** is set to **y** or **n**, the keywords **outpopulation**, **outcheck**, **outlevels**, **outdensity**, **outomp**, **outdirect**, **outdiscrete**, **outinverse**, **outgamma** and **outfission** (see below for their explanation) will all be set to the same value automatically. Setting **outbasic y** is generally not recommended since it produces a rather large output file. Less extensive output files can be obtained by enabling some of the aforementioned keywords separately.

### Examples

**outbasic y**

**outbasic n**

### Range

**y** or **n**

### Default

**outbasic n**

**block**

Flag to put output data in large data blocks. If the output is activated, exclusive spectra, particle spectra, discrete gamma-ray cross sections, angular distributions and Legendre coefficients are all stored in individual files if **block n**. If **block y**, they are stored in large data blocks.

**Examples**

**block y**

**block n**

**Range**

y or n

**Default**

**block n**, though **block y** will be set automatically if **endf y**.

**Comments**

For example, if **block n**, there could be output files *sp100000E0001.000.tot*, *sp100000E0002.000.tot* etc. while with **block y**, the data for all incident energies is stored in *sp100000.tot*.



**outpopulation**

Flag for the output of the population, as a function of excitation energy, spin and parity, of each compound nucleus in the reaction chain before it decays.

**Examples**

**outpopulation y**

**outpopulation n**

**Range**

y or n

**Default**

the same value as **outbasic**: **outpopulation n**

**outcheck**

Flag for the output of various numerical checks. This is to check interpolation schemes for the transformation from the emission grid to the excitation energy grid and vice versa, and to test the WFC method by means of flux conservation in the binary compound nucleus calculation. Also, the emission spectra integrated over energy are compared with the partial cross sections, and summed exclusive channel cross sections are checked against total particle production cross sections and residual production cross sections.

**Examples**

**outcheck y**

**outcheck n**

**Range**

**y or n**

**Default**

the same value as **outbasic**: **outcheck n**

**outdecay**

Flag for the output of detailed decay information per excitation bin in the decaying nuclide. Thus flag may give a **HUGE** output and is to be used with care. For every bin in the continuum, there is a detailed table for the decay via all particles to all other bins.

**Examples**

**outdecay y**

**outdecay n**

**Range**

y or n

**Default**

**outdecay n**

**Comments**

Use with care, this can give a huge output file.

**outexcitation**

Flag for the output of excitation functions, i.e. cross sections as a function of incident energy, such as residual production cross sections, inelastic cross sections, etc.

**Examples**

**outexcitation y**

**outexcitation n**

**Range**

y or n

**Default**

**outexcitation n** if only one incident energy is given in the input file, and **outexcitation y** for more than one incident energy.

**outspectra**

Flag for the output of angle-integrated emission spectra.

**Examples**

**outspectra y**

**outspectra n**

**Range**

**y** or **n**

**Default**

**outspectra y** if only one incident energy is given in the input file, and **outspectra n** for more than one incident energy.

**outbinspectra**

Flag for the output of emission spectra per compound nucleus bin. **outbinspectra** is only active if **outspectra y** and **population y**.

**Examples**

**outbinspectra y**

**outbinspectra n**

**Range**

y or n

**Default**

**outbinspectra n.**

## **ddxmode**

Option for the output of double-differential cross sections. There are 4 possibilities.

### **Examples**

**ddxmode 0:** No output.

**ddxmode 1:** Output per emission energy as a function of angle (angular distributions).

**ddxmode 2:** Output per emission angle as a function of energy (spectra).

**ddxmode 3:** Output per emission energy and per emission angle.

### **Range**

$$0 \leq \text{ddxmode} \leq 3$$

### **Default**

**ddxmode 0.** If there is a **fileddxe** keyword, see p. 680, in the input file, **ddxmode 1** will be set automatically. If there is a **fileddxa** keyword, see p. 681, in the input file, **ddxmode 2** will be set automatically. If both **fileddxe** and **fileddxa** are present, **ddxmode 3** will be set automatically.

**outgamdis**

Flag for the output of discrete gamma-ray intensities. All possible discrete gamma transitions for all nuclei are followed. In the output they are given in tables per nuclide and for each decay from state to state.

**Examples**

**outgamdis y**

**outgamdis n**

**Range**

**y or n**

**Default**

**outgamdis n**



**sacs**

Flag for the output of the statistical analysis of cross sections. With this flag the maximum cross section and the corresponding energy per excitation function are printed on a file.

**Examples**

**sacs y**  
**sacs n**

**Range**

**y or n**

**Default**

**sacs n**

**integral**

Flag to run TASMANT for covariances or optimization of integral activation data.

**Examples**

```
#integral n  
#integral y
```

**Range**

y or n

**Default**

```
#integral n
```

---

**fileelastic**

Flag to write the elastic angular distribution on a separate file *XXYYY.YYYang.L00*, where *XX* is the particle symbol in (2a1) format (e.g. *nn* for elastic scattering), and *YYY.YYY* the incident energy in (f7.3) format. The file contains the angle, and 3 columns containing the total, shape elastic, and compound elastic angular distribution, respectively. If in addition **outlegendre y**, the elastic scattering Legendre coefficients will be written on a file *XXYYY.YYYleg.L00*. This file contains the *L*-value, and 4 columns containing the total, direct, compound and normalized Legendre coefficient. **fileelastic** is only active if **outangle y**.

**Examples**

**fileelastic y**, giving files *nn014.000ang.L00*, and (if **outlegendre y**) *nn014.000leg.L00*, for an incident energy of 14 MeV.

**fileelastic n**

**Range**

y or n

**Default**

**fileelastic n**

## fileangle

Designator for the output of the non-elastic angular distribution of one specific level on a separate file *PXYYY.YYYang.LMM*, where P and X are the particle symbols in (a1) format for the projectile and ejectile, respectively, YYY.YYY is the incident energy in (f7.3) format, and MM is the level number in (i2.2) format. The file contains the angle and 3 columns with the total inelastic, direct inelastic and compound inelastic angular distribution to the specified level. On the input line we read the level number. The **fileangle** keyword can appear more than once in an input file, one for each level that one is interested in. It will automatically produce files for all ejectiles. If in addition **outlegendre y**, the non-elastic scattering Legendre coefficients will be written on a file *PXYYY.YYYleg.LMM*. This file contains the *L*-value, and 4 columns containing the total, direct, compound and normalized Legendre coefficient. **fileangle** is only active if **outdiscrete y** and **outangle y**.

## Examples

**fileangle 2**, giving files *np014.000ang.L02* and (**outlegendre y**) *np014.000leg.L02*, for the (*n, p*) reaction to the second discrete level and an incident energy of 14 MeV, and similarly for the other ejectiles.

## Range

$0 < \text{fileangle} < \text{numlev}$ . Currently, **numlev=30**

## Default

**fileangle** not active.

## **filediscrete**

Designator for the output of the excitation function of one specific non-elastic level on a separate file *PX.LMM*, where P and X are the particle symbols in (a1) format for the projectile and ejectile, respectively, and MM is the level number in (i2.2) format. The file contains the incident energy and 3 columns with the total inelastic, direct inelastic and compound inelastic cross section to the specified level. On the input line we read the level number. The **filediscrete** keyword can appear more than once in an input file, one for each level that one is interested in. It automatically produces a file for each ejectile. **filediscrete** is only active if **outdiscrete y**.

### **Examples**

**filediscrete 2** , giving files *nn.L02*, *np.L02*, etc. for the excitation functions of (inelastic and other) neutron scattering to the second discrete level.

### **Range**

$0 < \text{filediscrete} < \text{numlev}$ . Currently, **numlev=30**

### **Default**

**filediscrete** not active.

## filechannels

Flag to write the exclusive channel cross sections as a function of incident energy on separate files. The files will be called *xsNPDTHA.tot*, where N is the neutron number of the exclusive channel, P the proton number, etc., in (a1) format. For example *xs210000.tot* contains the excitation function for  $\sigma(n, 2np)$ , if the incident particle was a neutron. The files contain the incident energy and 3 columns with the exclusive cross section, the associated gamma-ray production cross section, and the fraction of this cross section relative to the total residual production cross section. If in addition isomers can be produced, files called *xsNPDTHA.LMM* will be created with MM the isomeric level number in (i2.2) format. If **filechannels y**, the exclusive binary continuum cross sections, such as continuum inelastic scattering, will also be written to files *PX.con*, where P and X are the particle symbols in (a1) format for the projectile and ejectile, respectively. If *outspectra y*, the exclusive channel spectra will be written on files *spNPDTHAEYYY.YYY.tot* where YYY.YYY is the incident energy in (f7.3) format. The files contain the incident energy and 6 columns, with the spectra per outgoing particle type. **filechannels** is only active if **channels y**.

### Examples

**filechannels y**

**filechannels n**

### Range

y or n

### Default

**filechannels n**

---

## filespectrum

Designator for the output of the composite particle spectrum for a specific particle type on a separate file. On the input line we read the particle symbols. This will result in files *XspecYYY.YYY.tot*, where X is the outgoing particle symbol in (a1) format, and YYY.YYY the incident energy in (f7.3) format. The file contains the emission energy and 5 columns with the total, direct, pre-equilibrium, multiple pre-equilibrium and compound spectrum, respectively. **filespectrum** is only active if **outspectra y**.

### Examples

**filespectrum n p a**, giving files *nspec014.000.tot*, *pspec014.000.tot* and *aspec014.000.tot*, for an incident energy of 14 MeV.

### Range

**n p d t h a**

### Default

**filespectrum** not active.

**fileddxe**

Designator for the output of double-differential cross sections per emission energy for a specific particle type. On the input line we read the particle type and the emission energy. This will result in files *XddxYYY.Y.mev*, where X is the particle symbol in (a1) format and YYY.Y the emission energy in (f5.1) format. The file contains the emission angle and 5 columns with the total, direct, pre-equilibrium, multiple pre-equilibrium and compound spectrum, respectively. The **fileddxe** keyword can appear more than once in an input file, one for each outgoing energy that one is interested in. If there is at least one **fileddxe** keyword in the input **ddxmode**, see p. 671, will automatically be enabled.

**Examples**

**fileddxe n 60.** (giving a file *nddx060.0.mev*).

**Range**

**n p d t h a** for the particles and **0.** -  $E_{inc}$  for outgoing energies.

**Default**

**fileddxe** not active.



## **fileddxa**

Designator for the output of double-differential cross sections per emission angle for a specific particle type. On the input line we read the particle type and the emission angle. This will result in files *XddxYYY.Y.deg*, where X is the particle symbol in (a1) format and YYY.Y the emission angle in (f5.1) format. The file contains the emission energy and 5 columns with the total, direct, pre-equilibrium, multiple pre-equilibrium and compound spectrum, respectively. The **fileddxa** keyword can appear more than once in an input file, one for each outgoing angle that one is interested in. If there is at least one **fileddxa** keyword in the input **ddxmode**, see p. 671, will automatically be enabled.

## **Examples**

**fileddxa n 30.** (giving a file *nddx030.0.deg*).

## **Range**

**n p d t h a** for the particles and **0. - 180.** for outgoing angles.

## **Default**

**fileddxa** not active.

**filegamdis**

Flag to write the discrete gamma-ray intensities as a function of incident energy to separate files. This will result in files *gamZZZAAALYYLMM.tot*, where *ZZZ* is the charge number and *AAA* is the mass number in (i3.3) format, *YY* is the number of the initial discrete state and *MM* the number of the final discrete state. **filegamdis** is only active if **outgamdis y**.

**Examples**

```
filegamdis y  
filegamdis n
```

**Range**

y or n

**Default**

filegamdis n

---

**filetotal**

Flag to write all the total cross sections as a function of incident energy on a separate file *total.tot*. The file contains the incident energy, and 9 columns containing the non-elastic, total elastic, total, compound elastic, shape elastic, reaction, compound non-elastic, direct and pre-equilibrium cross section. In addition, the total particle production cross sections will be written on files *Xprod.tot*, with X the particle symbol in (a1) format. Finally, separate x-y tables will be made for the total cross section, on *totalxs.tot*, the total elastic cross section, on *elastic.tot*, and the total nonelastic cross section, on *nonelastic.tot*. **filetotal** is only active if **outexcitation y** or **endf y**.

**Examples****filetotal y****filetotal n****Range**

y or n

**Default****filetotal n**

**fileresidual**

Flag to write all the residual production cross sections as a function of incident energy on separate files. The files for the total (i.e. the sum over ground state + isomers) residual production cross sections have the name *rpZZZAAA.tot* where *ZZZ* is the charge number and *AAA* is the mass number in (i3.3) format. If a residual nuclide contains one or more isomeric states, there are additional files *rpZZZAAA.LMM*, where *MM* is the number of the isomer (ground state=0) in (i2.2) format. The files contain the incident energy and the residual production cross section. **fileresidual** is only active if **outexcitation y** or **endf y**.

**Examples****fileresidual y****fileresidual n****Range**

y or n

**Default****fileresidual n**

---

## **filerecoil**

Flag to write the recoil spectra of the residual nuclides as a function of incident energy on separate files. The files for the recoil spectra have the name *recZZZAAAspecYYY.YYY.tot* where *ZZZ* is the charge number and *AAA* is the mass number in (i3.3) format and *YYY.YYY* the incident energy in (f7.3) format. If in addition **flagchannels y**, there are additional files *spNPDTHAEYYY.YYY.rec*, where *N* is the neutron number of the exclusive channel, *P* the proton number, etc. The files contain the incident energy and the recoil.

### **Examples**

**filerecoil y**

**filerecoil n**

### **Range**

**y or n**

### **Default**

**filerecoil n**

**components**

Flag to write the direct, pre-equilibrium and compound components of the cross sections in the various output files.

**Examples**

**components y**

**components n**

**Range**

**y or n**

**Default**

**components n**

## **35. Keywords of TALYS**

Table 35.1: The keywords of TALYS.

Keyword	Range	Default	Page
a	1. - 100.	table or systematics	354
aadjust	0.5 - 2.	1.	355
abundance	filename	no default	320
addiscrete	y,n	y	537
addelastic	y,n	y	538
adepthcor	0.5 - 1.5	1.	481
alimit	1. - 100.	systematics	356
alphald	0.01 - 0.2	systematics	387
alphaomp	1 - 8	1	479
angles	1-numang	90	304
anglescont	1-numangcont (36)	36	305
anglesrec	1-numangrec (9)	9	311
aradialcor	0.5 - 1.5	1.	480
area	0. - 10000.	1.	654
astro	y,n	n	642
astroE	0.00001-1.	0.	644
astroex	y,n	n	646
astrog	y,n	n	643
astroT	0.0001-10.	0.	645
asys	y,n	n	357
autorot	y,n	n	527
avadjust	0.5 - 2.	1.	449
avadjustF	-100 - 100	no default	506
avdadjust	0.5 - 2.	1.	461
avdadjustF	-100 - 100	no default	510
avsoadjust	0.5 - 2.	1.	472
avsoadjustF	-100 - 100	no default	514
awadjust	0.5 - 2.	1.	455
awadjustF	-100 - 100	no default	508
awdadjust	0.5 - 2.	1.	463
awdadjustF	-100 - 100	no default	512
awsoadjust	0.5 - 2.	1.	474
awsoadjustF	-100 - 100	no default	516
axtype	1-5	1 and 2,3 for $N > 144$	607
bdamp	0. - 50.	0.01	612
bdampadjust	0.01 - 100.	1.	613
best	y,n	n	294
bestbranch	y,n	y	346
bestend	y,n	n	323
bestpath	pathname	no default	324



Continuation of Table 35.1.

Keyword	Range	Default	Page
beta2	0. - 1.5	0.-1. (barrier dep.)	332
betafiscor	0.1 - 10.	1.	620
betafiscoradjust	0.1 - 10.	1.	622
betald	0. - 0.5	systematics	388
bins	0, 2 - numbins (100)	40	286
block	y,n	n	664
branch	0 - numlev	no default	344
breakupmodel	1 - 2	1	597
Cbreak	0. - 10.	1.	596
cfermi	0. - 1000.	10.	394
cfermibf	0. - 1000.	10.	395
cglobal	-10. - 10.	1.	400
channelenergy	y,n	n	314
channels	y,n	n	292
Cknock	0. - 10.	1.	595
class2	y,n	n	617
class2file	filename	no default	619
class2width	0.01 - 10.	0.2	618
Cnubar1	0.1 - 10.	1.	633
Cnubar2	0.1 - 10.	1.	634
colenhance	y,n [Z,A]	n	380
colldamp	y,n	n	381
components	y,n	n	686
compound	y,n	y	548
core	-1,1	-1	545
coulomb	y,n	y	539
cpang	y,n	n	540
Cstrip	0. - 10.	1.	594
ctable	-10. - 10.	1.	360
ctableadjust	-10. - 10.	1.	361
ctmglobal	y,n	n	385
D0	1.e-6 - 10000.	table	417
d1adjust	0.2 - 5.	1.	464
d2adjust	0.2 - 5.	1.	465
d3adjust	0.2 - 5.	1.	466
ddxmode	0 - 3	0	671
deformfile	filename	no default	544
deltaW	-20. - 20.	calculated	358
deuteronomp	1 - 5	1	478
disctable	1 - 3	1	342
dispersion	y,n	n	476

Continuation of Table 35.1.

Keyword	Range	Default	Page
E0	-10. - 10.	calculated	370
E0adjust	0.1 - 10.	1.	371
E1file	filename	no default	426
Eback	0. - Ebeam	Ebeam - 5.	652
Ebeam	0. - Ein	no default	651
eciscalc	y,n	y	523
eciscompound	y,n	n	556
ecisdwba	y,n	y	598
ecissave	y,n	n	522
egr	1. - 100.	table or systematics	408
egradjust	0.1 - 10.	1.	411
ejectiles	g n p d t h a	g n p d t h a	295
Ejoin	0. - 1000.	200.	492
electronconv	y,n	n	345
element	3 - 124 or Li - C4	no default	279
Elow	1.e-6 - 1.	$D_0$	317
elwidth	1.e-6 - 100.	0.5	533
Emsdmin	>0.	Eninc/5.	602
endf	y,n	n	302
endfdetail	y,n	y for n,g; n otherwise	303
endfecis	y,n	y	531
energy	1.e-11 - 250.	no default	281
epr	1. - 100.	no default	436
epradjust	0.1 - 10.	1.	439
equidistant	y,n	y	315
Estop	1.e-5 - 1000.	1000.	301
Esurf	0. - 38.	systematics	573
etable	-10. - 10.	0.	420
etableadjust	-10. - 10.	0.	421
Exmatch	0.1 - 20.	calculated	366
Exmatchadjust	0.1 - 10.	1.	367
expmass	y,n	y	331
ffevaporation	y,n	n	638
ffmodel	1 - 2	1	626
ffspin	y,n	n	631
fileangle	0 - numlev	no default	676
filechannels	y,n	n	678
fileddxa	n,...,a 0. - 180.	no default	681
fileddxe	n,...,a 0. - energy	no default	680
filedensity	y,n	n	378

Continuation of Table 35.1.

Keyword	Range	Default	Page
filediscrete	0 - numlev	no default	677
fileelastic	y,n	n	675
filefission	y,n	n	637
filegamdis	y,n	n	682
filepsf	y,n	n	428
filerecoil	y,n	n	685
fileresidual	y,n	n	684
filespectrum	g n p d t h a	no default	679
filetotal	y,n	n	683
fisbar	0. - 100.	table or systematics	608
fisbaradjust	0.1 - 10.	1.	610
fisfeed	y,n	n	640
fishw	0.01 - 10.	table or systematics	609
fishwadjust	0.1 - 10.	1.	611
fismodel	1 - 5	1	605
fismodelalt	3 - 4	4	606
fiso	0.01 - 100.	1.5, 2 or 5 (nuc-dep)	430
fisom	0.01 - 100.	1.5, 2 or 5 (nuc-dep)	431
fispartdamp	y,n	n	614
fission	y,n,A $\geq 56$	y for mass > 209, n for mass $\leq 209$	604
Fsadjust	0.1 - 10.	1.	628
ftable	0.1 - 10.	1.	422
fullhf	y,n	n	555
fymodel	1 - 5	2	625
g	0.1-100.	systematics	579
gadjust	0.5 - 2.	1.	584
gamgam	0. - 10.	table or systematics	414
gamgamadjust	0.1 - 10.	1.	415
gammald	0.01 - 1.	systematics	389
gammashell1	0. - 1.	systematics	390
gammashell2	0. - 0.2	0.	391
gammax	1 - 6	2	404
gefran	1000 - 1000000	50000	632
ggr	1. - 100.	table or systematics	410
ggradjust	0.1 - 10.	1.	413
giantresonance	y,n	y for n,p; n otherwise	532
gn	0.1 - 100.	systematics	581
gnadjust	0.5 - 2.	1.	583
gnorm	0. - 100. or -1.	calculated	405
gp	0.1 - 100.	systematics	580
gpadjust	0.5 - 2.	1.	582

Continuation of Table 35.1.

Keyword	Range	Default	Page
gpr	1. - 100.	no default	438
gpradjust	0.1 - 10.	1.	441
group	y,n	n	557
gshell	y,n	n	585
hbstate			615
hbtransfile	filename	no default	616
Ibeam	0. - 10000.	1.	653
incadjust	y,n	y	494
inccalc	y,n	y	524
integral	filenames	no default	674
isomer	0. - 1.e38	1.	291
jlmode	0 - 3	1	483
jlmomp	y,n	n	482
Kph	1. - 100.	15.	578
Krotconstant	0.01 - 100.	1.	393
kvibmodel	1 - 2	2	384
labddx	y,n	n	310
ldmodel	1 - 6 [Z,A]	1	352
ldmodelCN	1 - 6	ldmodel	353
ldmodelracap	1 - 3	1	433
levelfile	filename	no default	343
Liso	0 - numlev	no default	290
localomp	y,n	y	477
Ltarget	0 - numlev	0	287
lurr	1 - numl	2	553
lv1adjust	0.5 - 1.5	1.	486
lvadjust	0.5 - 1.5	1.	484
lvsoadjust	0.5 - 1.5	1.	488
lw1adjust	0.5 - 1.5	1.	487
lwadjust	0.5 - 1.5	1.	485
lwsoadjust	0.5 - 1.5	1.	489
M1file	filename	no default	427
M2constant	0. - 100.	1.	586
M2limit	0. - 100.	1.	587
M2shift	0. - 100.	1.	588
mass	0, 5 - 339	no default	280
massdir	filename	no default	333
massdis	y,n	n	635
massexcess	-500. - 500.	mass table	335
massmodel	1 - 3	3	330

Continuation of Table 35.1.

Keyword	Range	Default	Page
massnucleus	A-0.5 - A+0.5	mass table	334
maxband	0 - 100	0	526
maxchannel	0 - 8	4	306
maxenrec	0 - numenrec (25)	10	312
maxlevelsbin	0 - numlev	10 (g,n,p,a) and 5 (d,t,h)	339
maxlevelsres	0 - numlev	10	340
maxlevelstar	0 - numlev	20	338
maxN	0 - numN-2	22	289
maxNrp	0 - numN-2	22	319
maxrot	0 - 10	2	525
maxZ	0 - numZ-2	10	288
maxZrp	0 - numZ-2	10	318
micro	y,n	n	321
mpreeqmode	1 - 2	1	569
msdbins	2 - numenmsd/2-1 (8)	6	601
multiptreeq	y,n or 0.001 - 250.	20.	568
nafit	y,n	n	350
ngfit	y,n	n	348
nnfit	y,n	n	350
Nlevels	0 - numlev	equal to maxlevelsres	341
Nlow	0 - 200	2	364
Ntop	0 - 200	table or equal to Nlevels	365
nonthermlev	0 - numlev	not used	647
nulldev	filename	no default	326
ompenenergyfile	filename	no default	322
omponly	y,n	n	495
onestep	y,n	n	600
optmod	filename	no default	496
optmodall	y,n	n	499
optmodfileN	filename	no default	497
optmodfileP	filename	no default	498
outangle	y,n	y for one energy, n for many	534
outbasic	y,n	n	663
outbinspectra	y,n	n	670
outcheck	y,n	equal to outbasic	666
outdecay	y,n	n	667
outdensity	y,n	equal to outbasic	379
outdirect	y,n	equal to outbasic	528
outdiscrete	y,n	equal to outbasic	536
outdwba	y,n	n	599

Continuation of Table 35.1.

Keyword	Range	Default	Page
outecis	y,n	n	530
outexcitation	y,n	n for one energy, y for many	668
outfission	y,n	equal to outbasic	636
outfy	y,n	n	630
outgamdis	y,n	n	672
outgamma	y,n	equal to outbasic	416
outinverse	y,n	equal to outbasic	529
outkd	y,n	n	501
outlegendre	y,n	n	535
outlevels	y,n	equal to outbasic	347
outmain	y,n	y	662
outomp	y,n	equal to outbasic	500
outpopulation	y,n	equal to outbasic	665
outpreequilibrium	y,n	n	572
outspectra	y,n	y for one energy, n for many	669
outransenergy	y,n	y	543
pair	0. - 10.	systematics	359
pairconstant	0. - 30.	12.	392
pairmodel	1 - 2	1	576
parity	y,n	n	386
partable	y,n	n	293
pfnsmodel	1 - 2	1	627
pglobal	-10. - 10.	0.	401
phmodel	1 - 2	1	577
popeps	0. - 1000.	1.e-3	299
popMeV	y,n	n	300
preeqcomplex	y,n	y	574
preeqmode	1 - 4	2	567
preeqspin	y,n or 1-3	n	570
preeqsurface	y,n	y	571
preequilibrium	y,n or 0.001 - 250.	Ex(NL)	566
production	y,n	n	650
projectile	n,p,d,t,h,a,g,0	no default	278
psfglobal	y,n	n	429
Pshift	-10. - 10.	systematics	372
Pshiftadjust	-10. - 10.	systematics	373
Pshiftconstant	-5. - 5.	0. or 1.09	374
ptable	-10. - 10.	0.	362
ptableadjust	-10. - 10.	1.	363
racap	y,n	n	432

Continuation of Table 35.1.

Keyword	Range	Default	Page
radialfile	filename	no default	491
radialmodel	1 - 2	2	490
radiounit	(k,M,G)Bq, (m,k)Ci	Gbq	656
rcadjust	0.5 - 2.	1.	475
Rclass2mom	0.1 - 10.	1.	399
reaction	y,n	y	308
recoil	y,n	n	309
recoilaverage	y,n	n	313
relativistic	y,n	y	307
rescuefile	filename	no default	325
reslib	library name, default	default	558
resonance	y,n	n	551
Rfiseps	0. - 1.	1.e-3	624
Rgamma	0. - 100.	2.	593
rho	0. - 100.	from table	655
riplomp	in RIPL table	2408 for actinides	503
riplrisk	y,n	n	504
Rnunu	0. - 100.	1.	589
Rnupi	0. - 100.	1.	590
rotational	n p d t h a	n p	519
Rpinu	0. - 100.	1.	592
Rpipi	0. - 100.	1.	591
Rprime	0. - 10.	not given	419
Rspincut	0. - 10.	1.	375
Rspincutff	0. - 20.	9.	376
Rtransmom	0.1 - 10.	1.	398
rvadjust	0.5 - 2.	1.	448
rvadjustF	-100 - 100	no default	505
rvdadjust	0.5 - 2.	1.	460
rvdadjustF	-100 - 100	no default	509
rvsoadjust	0.5 - 2.	1.	471
rvsoadjustF	-100 - 100	no default	513
rwadjust	0.5 - 2.	1.	454
rwadjustF	-100 - 100	no default	507
rwdadjust	0.5 - 2.	1.	462
rwdadjustF	-100 - 100	no default	511
rwsoadjust	0.5 - 2.	1.	473
rwsoadjustF	-100 - 100	no default	515
S0	0. - 10.	table	418
s2adjust	0.01 - 10.	1.	377
sacs	y,n	n	673

Continuation of Table 35.1.

Keyword	Range	Default	Page
segment	1 - 4	1	316
sfexp	0. - 10.	table or 0.347	434
sfth	0. - 10.	table or 0.5	435
sgr	0. - 10000.	table or systematics	409
sgradjust	0.1 - 10.	1.	412
shellmodel	1 - 2	1	383
skipCN	[Z,A]	no default	560
soswitch	0.1 - 10.	3.	541
soukho	y,n	y	502
spherical	y,n	n	518
spincutmodel	1 - 2	1	382
spr	0. - 10000.	table or systematics	437
spradjust	0.1 - 10.	1.	440
statepot	y,n	n	542
strength	1 - 8	1 for n, 2 for others	406
strengthM1	1 - 2	2	407
strucpath	filename	no default	327
sysreaction	n p d t h a	d t h a	521
T	0.001 - 10.	calculated	368
Tadjust	0.1 - 10.	1.	369
Tcool	0. - 1.e6	1 d	659
Tirrad	0. - 1.e6	1 d	658
Tladjust	0.001 - 1000.	1.	520
Tmadjust	0.1 - 10.	1.	629
transeps	0. - 1.	1.e-8	297
transpower	2 - 20	5	296
Tres	0. - 1.e12	293.16	559
twocomponent	y,n	n	575
Ufermi	0. - 1000.	30.	396
Ufermibf	0. - 1000.	45.	397
upbend	y,n	n	442
upbendc	0. - 1.e-5	0.	443
upbende	0. - 10.	0.	444
upbendf	-10. - 10.	0.	445
urr	y,n	n	552
urnjoy	y,n	y	554
v1adjust	0.2 - 5.	1.	450
v2adjust	0.2 - 5.	1.	451
v3adjust	0.2 - 5.	1.	452
v4adjust	0.2 - 5.	1.	453



Continuation of Table 35.1.

Keyword	Range	Default	Page
vfiscor	0.1 - 10.	1.	621
vfiscoradjust	0.1 - 10.	1.	623
Vinfadjust	-200. - 0.	-30.	493
vso1adjust	0.2 - 5.	1.	467
vso2adjust	0.2 - 5.	1.	468
w1adjust	0.2 - 5.	1.	456
w2adjust	0.2 - 5.	1.	457
w3adjust	0.2 - 5.	1.	458
w4adjust	0.2 - 5.	1.	459
wso1adjust	0.2 - 5.	1.	469
wso2adjust	0.2 - 5.	1.	470
widthfluc	y,n or 0.001 - 20.	S(n)	549
widthmode	0 - 3	1	550
wtable	0. - 10.	1.	424
wtableadjust	0. - 10.	1.	425
xalphatherm	$10^{-20} - 10^{10}$	table or systematics	562
xscaptherm	$10^{-20} - 10^{10}$	table or systematics	561
xseps	0. - 1000.	1.e-7	298
xsptherm	$10^{-20} - 10^{10}$	table or systematics	563
yieldfile	filename	no default	639
yieldunit	num, mug, mg, g or kg	num	657



# Part Four: Miscellaneous

<b>36</b>	<b>Development of TALYS .....</b>	<b>701</b>
<b>37</b>	<b>Outlook and conclusions .....</b>	<b>707</b>
	<b>Bibliography .....</b>	<b>708</b>
<b>A</b>	<b>Log file of changes since TALYS-1.0 ..</b>	<b>717</b>
<b>B</b>	<b>Log file of changes since TALYS-1.2 ..</b>	<b>729</b>
<b>C</b>	<b>Log file of changes since TALYS-1.4 ..</b>	<b>737</b>
<b>D</b>	<b>Log file of changes since TALYS-1.6 ..</b>	<b>745</b>
<b>E</b>	<b>Log file of changes since TALYS-1.8 ..</b>	<b>751</b>
<b>F</b>	<b>Log file of changes since TALYS-1.9 ..</b>	<b>755</b>
<b>G</b>	<b>Log file of changes since TALYS-1.95 ..</b>	<b>759</b>
<b>H</b>	<b>Derivation of isotope production equations .....</b>	<b>765</b>
<b>I</b>	<b>GNU GENERAL PUBLIC LICENSE .....</b>	<b>767</b>



## 36. Development of TALYS

(Beta) versions of TALYS have been available since 2005, and since then the possibilities of the code have grown. For several users, who may revisit the code every now and then, it may be useful to see what has been added to the code since they last used it. Therefore, from the previous TALYS-1.x manuals we have copied the temporary development in terms of features and keywords.

### 36.1 From TALYS-1.0 to TALYS-1.2

On December 21, 2007 the first official version of the code, TALYS-1.0, was released. Since then, the code has undergone changes that fall in the usual two categories: significant extensions and corrections that may affect a large part of the user community, and several small bug fixes. As appendix to this manual, we add the full log file of changes since the release of TALYS-1.0. Here we list the most important updates:

- Further unification of microscopic structure information from Hartree-Fock-Bogoliubov (HFB) calculations. Next to masses, HFB deformation parameters are now also provided. The latest HFB-based tabulated level densities for both the ground state and fission barriers were included in the structure database. Another new addition is a database of microscopic particle-hole densities for preequilibrium calculations.
- Introduction of more keywords for adjustment. Often, these are defined relative to the default values, and thus often have a default value of 1. This is convenient since one does not have to look up the current value of the parameter if it is to be changed by a certain amount. Instead, giving e.g. **adjust 41 93 1.04** means multiplying the built-in value for the level density parameter  $a$  by 1.04. Such keywords are now available for many parameters of the optical model, level density, etc., enabling easy sensitivity and covariance analyses with TALYS.
- An alternative fission model has been added, or more precisely, revived. Pascal Romain and colleagues of CEA Bruyeres-le-Chatel has been more successful to fit actinide data with

older TALYS options for fission (present in versions of the code before the first released beta version TALYS-0.64) using effective level densities, than with the newer options with explicit collective enhancement for the level densities[150, 151]. Therefore we decided to re-include that option again, through the **colldamp** keyword. Also his corrections for the class-II states were adopted. We generally use this option now for our actinide evaluations, until someone is able to do that with the newer fission models built in the code.

- Average resonance parameters for the unresolved resonance range are calculated.
- More flexibility has been added for cases where TALYS fails to fit experimental data. It is now possible to normalize TALYS directly to experimental or evaluated data for each reaction channel. This is not physical, but needed for several application projects.
- Perhaps the most important error was found by Arjan Plompen and Olivier Bersillon. In the discrete level database of TALYS-1.0, the internal conversion coefficients were wrongly applied to the gamma-ray branching ratios. A new discrete level database has been generated and internal conversion is now correctly taken into account. In some cases, this affects the production of discrete gamma lines and the cross section for isomer production.
- TALYS can be turned into a "pure" optical model program with the keyword **omonly y**. After the optical model calculation and output it then skips the calculation for all nonelastic reaction channels. In this mode, TALYS basically becomes a driver for ECIS.
- It is now possible to save and use your best input parameters for a particular isotope in the structure database (helpful for data evaluation). With the **best** keyword these parameter settings can automatically be invoked. We include part of our collection in the current version.
- Pygmy resonance parameters for gamma-ray strength functions can be included.

To accommodate all this, plus other options, the following new keywords were introduced: **aadjust**, see page 355, **rvadjustF**, see page 505, **avadjustF**, see page 506, **rvdadjustF**, see page 509, **avdadjustF**, see page 510, **rvsoadjustF**, see page 513, **avsoadjustF**, see page 514, **best**, see page 294, **colldamp**, see page 381, **coulomb**, see page 539, **epr**, see page 436, **fiso**, see page 430, **gamgamadjust**, see page 415, **gnadjust**, see page 583, **gpadjust**, see page 582, **gpr**, see page 438, **hbstate**, see page 615, **jlmmode**, see page 483, **micro**, see page 321, **ompenergyfile**, see page 322, **omonly**, see page 495, **phmodel**, see page 577, **radialmodel**, see page 490, **rescuefile**, see page 325, **s2adjust**, see page 377, **soswitch**, see page 541, **spr**, see page 437, **strengthM1**, see page 407, **urr**, see page 552, **xsalphatherm**, see page 562, **xscaptherm**, see page 561, **xsptherm**, see page 563.

## 36.2 From TALYS-1.2 to TALYS-1.4

On December 23, 2009 the second official version of the code, TALYS-1.2, was released. Since then, the code has undergone changes that fall in the usual two categories: significant extensions and corrections that may affect a large part of the user community, and several small bug fixes. As appendix to this manual, we add the full log file of changes since the release of TALYS-1.2. Here we list the most important updates:

- A new phenomenological model for break-up reactions by Connie Kalbach was included. This model is documented in an unpublished report to the FENDL-3 meeting at the IAEA, December 2010, and resolves some of the cross section prediction problems that were observed in earlier versions of TALYS. The contribution can be scaled by a normalization factor (**Cbreak** keyword).

- The alpha double-folding potential by Demetriou et al[67] was added as an option.
- Several deuteron OMP's were added to provide a better prediction of deuteron reaction cross sections and transmission coefficients.
- As part of the output of the binary reaction information, the compound nucleus formation cross section as a function of spin and parity is now also printed.
- TALYS is now able to produce unresolved resonance range (URR) parameters with a higher quality than that of TALYS-1.2, thanks to modelling and programming help by Gilles Noguere, CEA-Cadarache, and testing by Paul Koehler, ORNL.
- We increased the flexibility of using level densities. Up to TALYS-1.2, it was possible to choose between 5 level density models, which were then used for *all* nuclides in the calculation. It is now possible to choose one of these 5 level density models *per nuclide*, e.g. the Constant Temperature model for the target nucleus and the Backshifted Fermi Gas model for the compound nucleus. The **ldmodel** and **colenhance** keywords have been extended with (optional) Z,A identifiers.
- We have added the possibility to print the direct, pre-equilibrium and compound components of each cross section in the output files, in order to check which reaction mechanism is responsible for different parts of the excitation function. This can be enabled with the new **components** keyword.
- It is now possible to calculate the Maxwellian averaged cross section (MACS) at a user-defined energy. By using the **astroE** keyword, one can for example compare TALYS with experimental 30 keV averaged capture cross sections.
- We have included the possibility to calculate the so-called effective cross section for integral activation measurements, by folding the excitation functions by an experimental flux. In *talys/structure/flux*, we have stored more than 40 spectra which have been used in past activation benchmarks.

For TALYS-1.4 the following new keywords were introduced: **Cbreak**, see page 596, **deuteronomp**, see page 478, **rwadjustF**, see page 507, **awadjustF**, see page 508, **rwdadjustF**, see page 511, **awdadjustF**, see page 512, **rwsadjustF**, see page 515, **awsoadjustF**, see page 516, **integral**, see page 674, **urnjoy**, see page 554, **components**, see page 686, **astroE**, see page 644, **astroT**, see page 645, while the possibilities of **alphaomp**, see page 479, **ldmodel**, see page 352, **colenhance**, see page 380, were extended.

It is worthwhile to mention here that the structure database has almost not changed since the release of version 1.2. The only exceptions are the *structure/mass/hfb* directory, which contains the latest Hartree-Fock-Bogoliubov + Skyrme-based theoretical masses, and the *structure/flux* directory, which contains a collection of experimental neutron spectra for the calculation of effective integral cross sections.

### 36.3 From TALYS-1.4 to TALYS-1.6

On December 28, 2011 the third official version of the code, TALYS-1.4, was released. Since then, the code has undergone changes that fall in the usual two categories: significant extensions and corrections that may affect a large part of the user community, and several small bug fixes. As appendix to this manual, we add the full log file of changes since the release of TALYS-1.4. Here we list the most important updates:

- The activity of all residual products can be given in the output. This means that TALYS can now directly be used for medical isotope production with accelerators through the use of

the **production** keyword. For this the decay data library is added to the TALYS structure database. Excitation functions are thus automatically transferred into isotope production rates in MBq or Ci, as a function of time. Various extra keywords for this are included for flexibility.

- Part of the GEF code for fission yields, by Schmidt and Jurado, has been translated into FORTRAN and into a TALYS subroutine by Vasily Simutkin and Michail Onegin. This now allows the calculation of fission yields and fission neutron multiplicities as a function of Z, N, A and the number of emitted neutrons, while the calculation of fission neutron spectra will be done in a future version of TALYS. GEF is designed to produce this information starting from an excited state of a fissile system. Hence, the TALYS-GEF combination can now give estimates for these fission quantities in the case of multi-chance fission.
- Non-equidistant binning for the excitation energy grid was introduced. The excitation energies can now be tracked on a logarithmic grid, and has now become the default. It allows to use less excitation energy bins while not losing precision. The old situation of TALYS-1.4 and before can be invoked with the new **equidistant** keyword, but from now on the default will be **equidistant n**. Especially for cross sections above 100 MeV the improvement is significant in terms of smoothness and more reliable absolute values.
- Thanks to the previous improvement, TALYS can now technically be used up to 1 GeV. For this, the Koning-Delaroche OMP was extended as well.
- A radiative capture model for the direct capture cross section was added.
- An extra option for the pre-equilibrium spin distribution was included, see the **preeqspin** keyword.
- The **energy** keyword has been made more flexible, avoiding long files with energy values to be constructed by the user.
- About 70 keywords can now have energy-dependent values. After the keyword and its value are given in the input, an energy range can be specified over which the corresponding model parameter is altered. Hence, it is for example possible to have a (usual) constant value for the optical model radius  $r_V$ , but to increase its value between e.g. 6 and 10 MeV, through a simple addition on the input line.

For TALYS-1.6 the following new keywords were introduced: **equidistant**, see page 315, **disctable**, see page 342, **Tljadjust**, see page 520, **branch**, see page 344, **racap**, see page 432, **sfexp**, see page 434, **sftth**, see page 435, **ldmodelracap**, see page 433, **Vinfadjust**, see page 493, **Ejoin**, see page 492, **w3adjust**, see page 458, **w4adjust**, see page 459, **incadjust**, see page 494, **Liso**, see page 290, **fisfeed**, see page 640, **production**, see page 650, **Ebeam**, see page 651, **Eback**, see page 652, **Tcool**, see page 659, **Tirrad**, see page 658, **rho**, see page 655, **Ibeam**, see page 653, **Area**, see page 654, **radiounit**, see page 656, **yieldunit**, see page 657, **cpang**, see page 540, **egradjust**, see page 411, **ggradjust**, see page 413, **sgradjust**, see page 412, **epradjust**, see page 439, **gpradjust**, see page 441, **spradjust**, see page 440, **E0adjust**, see page 371, **Exmatchadjust**, see page 367, **Tadjust**, see page 369, **fisbaradjust**, see page 610, **fishwadjust**, see page 611, **fymodel**, see page 625, **outfy**, see page 630, **gefran**, see page 632, while the possibilities of **preeqspin**, see page 570, **ldmodel**, see page 352, were extended.

### 36.4 From TALYS-1.6 to TALYS-1.8

On December 23, 2013 the fourth official version of the code, TALYS-1.6, was released. Since then, the code has undergone changes that fall in the usual two categories: significant extensions



and corrections that may affect a large part of the user community, and several small bug fixes. As appendix to this manual, we add the full log file of changes since the release of TALYS-1.6. Here we list the most important updates:

- Low energy resonance cross sections were added to the output. For this, the resonance keyword was introduced. If **resonance y**, TALYS will read in resonance parameters, from various possible sources, and call the RECENT code of Red Cullen's PREPRO package[152]. Low energy pointwise resonance cross sections are added to channels like total, elastic, fission and capture. Choices for adoption of resonance parameters from various libraries can be made with the **reslib** keyword, which can be equal to default, endfb7.1, jeff3.1, jendl4.0. In addition, use can be made of Cullen's SIGMA1 code, for resonance broadening, and GROUPIE, for groupwise cross sections.
- The stand-alone Fortran version of the GEF code was added as a subroutine to enable, in principle, the evaporation of fission fragments by TALYS. For this, TALYS "loops over itself", i.e. it is restarted internally for every excited fission fragment. The results are not yet good however. The option for this is **fymodel 3**. In principle, quantities like the neutron spectrum for evaporated fission fragments, nubar etc. are available now.
- The range of elements that is allowed in the input is extended up to  $Z=124$ , with nuclear symbols Rg(111), Cn(112), Nh(113), Fl(114), Mc(115), Lv(116), Ts(117), Og(118), B9(119), C0-4(120-124).
- More models (**strength 6, 7, 8** for gamma-ray strength functions are added.
- More models (**alphaomp 6, 7, 8** for the alpha optical model potential are added. *alphaomp 6*, i.e. of Avrigeanu et al [69] is now the default alpha OMP potential.

For TALYS-1.8 the following new keywords were introduced: **pshiftadjust**, see page 373, **resonance**, see page 551, **Tres**, see page 559, **group**, see page 557, **reslib**, see page 558, **bestend**, see page 323, **E1file**, see page 426, **Estop**, see page 301, **popMeV**, see page 300, **bestbranch**, see page 346, **breakupmodel**, see page 597, **outhinspectra**, see page 670, **ffspin**, see page 631, **skipCN**, see page 560, while the possibilities of **strength**, see page 406, **element**, see page 279, **alphaomp**, see page 479, **fymodel**, see page 625, were extended.

### 36.5 From TALYS-1.8 to TALYS-1.9

On December 28, 2015 the fifth official version of the code, TALYS-1.8, was released. Since then, the code has undergone changes that fall in the usual two categories: significant extensions and corrections that may affect a large part of the user community, and several small bug fixes. As appendix to this manual, we add the full log file of changes since the release of TALYS-1.8. Here we list the most important updates:

- More flexibility for gamma-ray strength functions, such as the possibility to read in M1 strength function tables.
- Extension of astrophysical reaction rate calculations to excited states.

For TALYS-1.9 the following new keywords were introduced: **M1file**, see page 427, **massdir**, see page 333, **upbendc**, see page 443, **upbende**, see page 444, **astroex**, see page 646, **nonthermlev**, see page 647, while the possibilities of **etable**, see page 420, **fable**, see page 422, were extended.

### 36.6 From TALYS-1.9 to TALYS-1.95

On December 23, 2017 the sixth official version of the code, TALYS-1.9, was released. Since then, the code has undergone changes that fall in the usual two categories: significant extensions and

corrections that may affect a large part of the user community, and several small bug fixes. As appendix to this manual, we add the full log file of changes since the release of TALYS-1.9. Here we list the most important updates:

- More, and better, options for the photon strength functions (PSF). New D1M/QRPA based M1 tables are provided, including so called 'upbend' formula to scale low-energy behavior, as well as improved E1 data tables and the Simplified Modified Lorentzian (SMLO), allowing better default predictions of the PSF. This impacts both neutron capture cross sections and photonuclear data.
- For fission, a model for partial damping (so far only full damping was included) was added, including extra flexibility for damping parameters.

For TALYS-1.95 the following new keywords were introduced: **bdamp**, see page 612, **bdampadjust**, see page 613, **ctableadjust**, see page 361, **ptableadjust**, see page 363, **etableadjust**, see page 421, **ftableadjust**, see page 423, **wtable**, see page 424, **wtableadjust**, see page 425, **outkd**, see page 501, **filepsf**, see page 428, **psfglobal**, see page 429, **maxNrp**, see page 319, **maxZrp**, see page 318, **outdecay**, see page 667, **Rprime**, see page 419, **sacs**, see page 673, **upbend**, see page 442, **upbendf**, see page 445, while the possibilities of **strength**, see page 406, were extended.

## 37. Outlook and conclusions

This manual describes TALYS-1.96/2.0, a nuclear reaction code. After several years of development, and application in various areas, by many users, we decided that after the previous (TALYS-0.64, TALYS-0.72) beta releases, the first official Version 1.0 of the code could be released at the end of 2007. After the release of TALYS-1.2 in December 2009, TALYS-1.4 in December 2011, TALYS-1.6 in December 2013, TALYS-1.8 in December 2015, TALYS-1.9 in December 2017, TALYS-1.95 in December 2019, TALYS-1.96 is the fifth update of that release.

We note that various extensions are possible for the physics included in TALYS, and some will be mentioned below. Obviously, we can not guarantee that these will be included in a future release (if any). This depends on the required effort, future careers of the authors, your willingness to share your extensions with us, our willingness to implement them, and in the case of significant extensions, financial input from research programs that require nuclear data.

In general, the nuclear structure database can still be extended with more tables based on microscopic nuclear structure calculations. Through a trivial change in the TALYS code, the impact of these ingredients on reaction calculations can immediately be tested.

TALYS is already being used with microscopic OMPs, but only with the spherical JLM method. An extension to deformed JLM has been made available by Eric Bauge, but is not yet implemented. A few direct reaction items, such as the prediction of the Isobaric Analogue State, are also yet to be completed.

A general analysis of all actinides simultaneously should result in a stable, ready-to-use fission database. It is clear that the theoretical fission models themselves are also not yet mature, even though microscopic fission paths are now included.

Concerning continuum reactions, there exists a microscopic multi-step direct code, MINGUS, for quantum-mechanical pre-equilibrium calculations (MSD/MSD) [153], which still needs to be merged with TALYS.

Coupling with high-energy intranuclear cascade (INC) codes is possible, now that TALYS is

able to take a pre-defined population distribution as the starting point. The INC code would take care of energies above e.g. 200 MeV, while TALYS takes over below that cut-off energy. The well-validated pre-equilibrium and Hauser-Feshbach approach at lower energies may then lead to more precise simulated data (including isomer production), even for reactions in the GeV

As for computational possibilities, the current day computer power enables to use nuclear model codes in ways that were previously thought impossible. Activities that have already proven to be possible are the generation of nuclear-model based covariances with Monte Carlo methods, automatic multi-parameter fitting of all partial cross sections to the existing experimental data, and dripline-to-dripline generation of all cross sections over the entire energy and projectile range, see e.g. the TENDL link on our website. The applications range from basic science (e.g. astrophysics) to the production of nuclear data libraries for existing and future nuclear technologies.

We are considering a complete upgrade of TALYS to Fortran90/95, which will be called TALYS-2.0. A prototype version is available and is expected to be released soon. For the moment, however, we have restricted ourselves to adding as much modules to TALYS as possible, using Fortran77, without letting that project interfere with the use of other programming languages. Once we feel TALYS has reached a certain level, in terms of included physics and options, that calls for a global upgrade to a more modern language, we will certainly do so and we will apply such an update to the whole code *at once*. At the moment, all authors of TALYS master at least Fortran77, which is a strong argument in favor of the present approach. However, TALYS-1.96 may have been the last version for which we insist on full Fortran77 compatibility. A huge extra effort would be needed to make our methods fully quality assured, even though we know that we are already deviating, in a positive sense, from those currently practised in computational nuclear science.

There are two important satellite codes for TALYS, written at NRG, that have appeared in the literature: TASMAN [154], for determining nuclear model based covariances and automatic optimization to experimental data, and TEFAL [154], for translating the results of TALYS into ENDF-6 data libraries. However, this is proprietary software, and it is not yet foreseen that this software becomes generally available. The current plan is to include both in a new Fortran90/95/03/08 version of TALYS, but they will also be released as separate packages shortly.

The development of TALYS has always followed the “first completeness, then quality” principle. This merely means that, in our quest for completeness, we tried to divide our effort equally among all nuclear reaction types. We think that, with the exception of a few issues the code is indeed *complete* in terms of predicted quantities. We now hope that TALYS also qualifies for “completeness and quality”. Nevertheless, it is certain that future theoretical enhancements as suggested above are needed to bring our computed results even closer to measurements.

## Bibliography

- [1] P.G. Young, E.D. Arthur, and M.B. Chadwick. In: *Workshop on Computation and Analysis of Nuclear Data Relevant to Nuclear Energy and Safety*, edited by M.K. Mehta and J.J. Schmidt, Feb. 10 - March 13 1992, Trieste, Italy (1993), p. 622.
- [2] M. Blann. In: *Workshop on Computation and Analysis of Nuclear Data Relevant to Nuclear Energy and Safety*, edited by M.K. Mehta and J.J. Schmidt, Feb. 10 - March 13 1992, Trieste, Italy (1993), p. 622.
- [3] M. Uhl and B. Strohmaier. “Computer code for particle induced activation cross sections and related quantities”. In: *IRK Vienna report 76/01* (1976).
- [4] M. Herman. In: *Workshop on Nuclear Reaction Data and Nuclear Reactors: Physics, Design and Safety*, edited by N. Paver, M. Herman and A. Gandini, March 13 - April 14 2000, Trieste Italy (2001), p. 137.
- [5] J. Raynal. “Notes on ECIS94”. In: *CEA Saclay Report CEA-N-2772* (1994).
- [6] R. Capote et al. “RIPL - Reference Input Parameter Library for calculation of nuclear reactions and nuclear data evaluation”. In: *Nucl. Data Sheets 110* (2009), p. 3107.
- [7] R. Michel et al. “Cross sections for the production of residual nuclides by low- and medium-energy protons from the target elements C, N, O, Mg, Al, Si, Ca, Ti, V, Mn, Fe, Co, Ni, Cu, Sr, Y, Zr, Nb, Ba and Au”. In: *Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms 129* (1997), pp. 153–193. DOI: 10.1016/S0168-583X(97)00213-9.
- [8] H. Vonach et al. “ $^{207,208}\text{Pb}(n,xn\gamma)$  reactions for neutron energies from 3 to 200 MeV”. In: *Phys. Rev. C 50* (4 1994), pp. 1952–1963. DOI: 10.1103/PhysRevC.50.1952. URL: <https://link.aps.org/doi/10.1103/PhysRevC.50.1952>.
- [9] A. Marcinkowski et al. “Neutron emission cross sections on  $^{93}\text{Nb}$  and  $^{209}\text{Bi}$  at 20 MeV incident energy”. In: *Nuclear Physics A 530* (1991), pp. 75–93. DOI: 10.1016/0375-9474(91)90756-V.

- [10] A.M. Baldin. In: *Kinematics of Nuclear Reactions*, Oxford University Press (1961).
- [11] M.B. Chadwick et al. In: *Nucl. Sci. Eng.* 131 (1999), p. 293.
- [12] M.B. Chadwick et al. “High Energy Nuclear Data Libraries for Accelerator-Driven Technologies: Computational Method for Heavy Recoils”. In: *Second International Conference on Accelerator-Driven Transmutation Technologies and Applications, Kalmar, Sweden, June 3-7 1996* (1996), p. 483.
- [13] J.-C. Sublet et al. “EASY-II(12): modern modelling of n, d, p, g, a activation and transmutation processes”. In: *Proceedings of the 12th International conference on Radiation Shielding, Sep 2-7 2012, Nara, Japan, Progress in Nuclear Science and Technology* (2013).
- [14] R.A. Forrest and J. Kopecky. “EASY-2005: Validation and new tools for data checking”. In: *Fus. Eng. Des* 82 (2007), p. 2471.
- [15] W.J. Huang et al. “The AME2016 atomic mass evaluation (I)”. In: *Chinese Physics C* 41, 3 (2017), p. 030002.
- [16] S. Goriely and J.M. Pearson. In: *Phys. Rev. Lett.* 102 (2009), p. 152503.
- [17] S. Goriely et al. “The Gogny-Hartree-Fock-Bogoliubov nuclear-mass model”. In: *European Physical Journal A* 52 (2016), p. 202.
- [18] P. Moller et al. “Nuclear ground-state masses and deformations: FRDM(2012)”. In: *Chinese Physics* 109 (2016), pp. 1–204.
- [19] J. Duflo and A. Zuker. In: *unpublished* ().
- [20] Arjan J Koning, Stéphane Hilaire, and Stéphane Goriely. “Global and local level density models”. In: *Nuclear Physics A* 810.1-4 (2008), pp. 13–76.
- [21] T. Ericson. In: *Adv. Phys.* 9 (1960), p. 425.
- [22] A. Gilbert and A.G.W. Cameron. In: *Can. J. Phys.* 43 (1965), p. 1446.
- [23] W. Dilg et al. In: *Nucl. Phys.* A217 (1973), p. 269.
- [24] H. Baba. In: *Nucl. Phys.* A159 (1970), p. 625.
- [25] A.V. Ignatyuk, G.N. Smirenkin, and A.S. Tishin. In: *Sov. J. Nucl. Phys.* 21, no. 3 (1975), p. 255.
- [26] A. Mengoni and Y. Nakajima. In: *J. Nucl. Sci. Techn.* 31 (1994), p. 151.
- [27] W.D. Myers and W.J. Swiatecki. In: *Nucl. Phys.* 81 (1966), p. 1.
- [28] S. Goriely. In: *Nucl. Phys.* A605 (1996), p. 28.
- [29] S. Hilaire. In: *PhD thesis, unpublished* (1997).
- [30] S.F. Mughabghab and C. Dunford. In: *Phys. Rev. Lett.* 81 (1998), p. 4083.
- [31] M.K. Grossjean and H. Feldmeier. In: *Nucl. Phys.* A444 (1985), p. 113.
- [32] P. Demetriou and S. Goriely. In: *Nucl. Phys.* A695 (2001), p. 95.
- [33] A.V. Ignatyuk, K.K. Istekov, and G.N. Smirenkin. In: *Sov. J. Nucl. Phys.* 29, no. 4 (1979), p. 450.
- [34] A.V. Ignatyuk et al. In: *Phys. Rev.* C47 (1993), p. 1504.
- [35] O.T. Grudzevich et al. In: *Proc. Nuclear Data for Science and Technology (Mito, JAERI)* (1988), p. 187.

- [36] A.S. Iljinov et al. In: *Nucl. Phys.* A543 (1992), p. 517.
- [37] A.R. Junghans et al. In: *Nucl. Phys.* A629 (1998), p. 635.
- [38] G. Hansen and A.S. Jensen. In: *Nucl. Phys.* A406 (1983), p. 236.
- [39] S. Goriely, F. Tondeur, and J.M. Pearson. In: *Atom. Data Nucl. Data Tables* 77 (2001), p. 311.
- [40] S. Goriely, S. Hilaire, and A.J. Koning. “Improved microscopic nuclear level densities within the HFB plus combinatorial method”. In: *Phys. Rev. C* 78 (2008), p. 064307.
- [41] S. Hilaire et al. “Temperature-dependent combinatorial level densities with the DIM Gogny force”. In: *Physical Review C* 86 (2012), p. 064317.
- [42] D.M. Brink. In: *Nucl. Phys.* 4 (1957), p. 215.
- [43] In: *Phys. Rev.* 126 (1962), p. 671.
- [44] J. Kopecky and M. Uhl. In: *Phys. Rev. C* 41 (1990), p. 1941.
- [45] M. Uhl J. Kopecky and R.E. Chrien. In: *Phys. Rev. C* 47 (1993), p. 312.
- [46] S. Goriely and E. Khan. “Large-scale QRPA calculation of E1-strength and its impact on the neutron capture cross section”. In: *Nucl. Phys. A* 706 (2002), pp. 217–232.
- [47] S. Goriely, E. Khan, and M. Samyn. “Microscopic + QRPA predictions of dipole strength for astrophysics calculations”. In: *Nucl. Phys. A* 739 (2004), pp. 331–352.
- [48] S. Goriely. “Radiative neutron captures by neutron-rich nuclei and the r-process nucleosynthesis”. In: *Phys. Lett. B* 436 (1998), pp. 10–18.
- [49] I. Daoutidis and S. Goriely. “Large-scale continuum random phase approximation for astrophysics calculations”. In: *Phys. Rev. C* 86 (2012), p. 034328.
- [50] S. Goriely et al. “Gogny+HFB+QRPA dipole strength function and its application to radiative nucleon capture cross section”. In: *Phys. Rev. C* 98 (2018), p. 014327.
- [51] B.L. Berman et al. In: *Phys. Rev.* 162 (1967), p. 1098.
- [52] J. Kopecky. In: *private communication* ().
- [53] D.G. Gardner. In: *Neutron Radiative Capture, OECD/NEA Series on Neutron Physics and Nuclear Data in Science and Technology*, eds. A. Michaudon et al. (1984), p. 62.
- [54] M.B. Chadwick et al. In: *Phys. Rev. C* 44 (1991), p. 814.
- [55] Yi Xu and Stephane Goriely. “Systematic study of direct neutron capture”. In: *Phys. Rev. C* 86 (2012), p. 045801.
- [56] AJ Koning and JP Delaroche. “Local and global nucleon optical models from 1 keV to 200 MeV”. In: *Nuclear Physics A* 713.3-4 (2003), pp. 231–310.
- [57] A.J. Koning. “Extension of TALYS to 1 GeV”. In: *Proceedings of ND-2013, New York* (2014).
- [58] S. Typel, O. Riedl, and H.H. Wolter. In: *Nuc. Phys.* A709 (2002), p. 299.
- [59] S. Chiba et al. In: *Spec. Meet. on the nucleon nucleus optical model up to 200 MeV, Bruyeres-le-Chatel* (1996).
- [60] S. Watanabe. In: *Nucl. Phys.* 8 (1958), p. 484.

- [61] D.G. Madland. In: *Proceedings of a Specialists' Meeting on preequilibrium nuclear reactions, Semmering, Austria, February 10-12 1988* (), p. 103.
- [62] W.W. Daehnick, J.D. Childs, and Z. Vrcelj. In: *Phys. Rev C* 21 (1980), p. 2253.
- [63] J. Bojowald et al. In: *Phys. Rev. C* 38 (1988), p. 1153.
- [64] Y. Han, Y. Shi, and Q. Shen. In: *Phys. Rev. C* 74 (2006), p. 044615.
- [65] Haixia An and Chonghai Cai. In: *Phys. Rev. C* 73 (2006), p. 054605.
- [66] L. McFadden and G.R. Satchler. In: *Nucl. Phys.* 84 (1966), p. 177.
- [67] C. Grama P. Demetriou and S. Goriely. In: *Nuclear Physics A* 707 (2002), p. 253.
- [68] J. Glorius et al. "Experimental cross sections of  $^{165}\text{Ho}(\alpha, n)^{168}\text{Tm}$  and  $^{166}\text{Er}(\alpha, n)^{169}\text{Yb}$  for optical potential studies relevant for the astrophysical  $\gamma$  process". In: *Phys. Rev. C* 89 (6 2014), p. 065808. DOI: 10.1103/PhysRevC.89.065808. URL: <https://link.aps.org/doi/10.1103/PhysRevC.89.065808>.
- [69] V. Avrigeanu, M. Avrigeanu, and C. Mihailescu. "Further explorations of the  $\alpha$ -particle optical model potential at low energies for the mass range  $A \approx 45-209$ ". In: *Phys. Rev. C* 90 (4 2014), p. 044612. DOI: 10.1103/PhysRevC.90.044612. URL: <https://link.aps.org/doi/10.1103/PhysRevC.90.044612>.
- [70] E. Bauge, J.P. Delaroche, and M. Girod. In: *Phys. Rev. C* 63 (2001), p. 024607.
- [71] F. Maréchal et al. In: *Phys. Rev. C* 60 (1999), p. 034615.
- [72] H. Sheit et al. In: *Phys. Rev. C* 63 (2000), p. 014604.
- [73] E. Khan et al. In: *Nucl. Phys A* 694 (2001), p. 103.
- [74] E. Bauge et al. In: *Phys. Rev. C* 61 (2000), p. 034306.
- [75] J.P. Jeukenne, A. Lejeune, and C. Mahaux. In: *Phys. Rep.* C25 (1976), p. 83.
- [76] J.P. Jeukenne, A. Lejeune, and C. Mahaux. In: *Phys. Rev. C* 10 (1974), p. 1391.
- [77] J.P. Jeukenne, A. Lejeune, and C. Mahaux. In: *Phys. Rev. C* 15 (1977), p. 10.
- [78] J.P. Jeukenne, A. Lejeune, and C. Mahaux. In: *Phys. Rev. C* 16 (1977), p. 80.
- [79] E. Bauge, J.P. Delaroche, and M. Girod. In: *Phys. Rev. C* 58 (1998), p. 1118.
- [80] R.R. Scheerbaum. In: *Nucl. Phys.* A257 (1976), p. 77.
- [81] R.K. Tripathi, F.A. Cucinotta, and J.W. Wilson. In: *NASA technical paper 3621* (1997).
- [82] T. Tamura. In: *Rev. Mod. Phys.* 37 (1965), p. 679.
- [83] J.P. Delaroche. In: *Proceedings of the International Symposium on Nuclear Data Evaluation Methodology, ed. C.L. Dunford, October 12-16 1992, Brookhaven, USA* (1992), p. 347.
- [84] N. Olsson, E. Ramström, and B. Trostell. In: *Nucl. Phys.* A513 (1990), p. 205.
- [85] B.V. Carlsson. In: *Workshop on Nuclear Reaction Data and Nuclear Reactors: Physics, Design and Safety, edited by N. Paver, M. Herman and A. Gandini, March 13 - April 14 2000, Trieste Italy* (2001), p. 61.
- [86] E. Sh Soukhovitskii et al. In: *J. Phys. G: Nucl. Part. Phys.* 30 (2004), p. 905.
- [87] P.P. Guss et al. In: *Phys. Rev. C* 39 (1989), p. 405.
- [88] G.R. Satchler. "Direct Nuclear Reactions". In: *Oxford University Press, New York* (1983).



- [89] P.E. Hodgson. “Nuclear Reactions and Nuclear Structure”. In: *Clarendon Press, Oxford* (1971).
- [90] A. van der Woude. In: *Electric and Magnetic Giant Resonances in Nuclei, edited by J. Speth (World Scientific, Singapore, 1991)* (1991), pp. 99–232.
- [91] C. Kalbach. In: *Phys. Rev. C* 62 (2000), p. 44608.
- [92] AJ Koning and MC Duijvestijn. “A global pre-equilibrium analysis from 7 to 200 MeV based on the optical model potential”. In: *Nuclear Physics A* 744 (2004), pp. 15–76.
- [93] H. Gruppelaar, P. Nagel, and P.E. Hodgson. In: *Riv. Nuovo Cimento* 9, No. 7 (1986), p. 1.
- [94] E. Gadioli and P.E. Hodgson. “Pre-equilibrium nuclear reactions, Oxford Univ. Press”. In: (1992).
- [95] C. Kalbach. In: *Phys. Rev. C* 33 (1986), p. 818.
- [96] C.K. Cline and M. Blann. In: *Nucl. Phys. A* 172 (1971), p. 225.
- [97] J. Dobeš and E. Běták. In: *Zeit. Phys. A* 310 (1983), p. 329.
- [98] E. Běták and J. Dobeš. In: *Zeit. Phys. A* 279 (1976), p. 319.
- [99] C.Y. Fu. In: *Nucl. Sci. Eng.* 86 (1984), p. 344.
- [100] C. Kalbach. In: *Phys. Rev. C* 32 (1985), p. 1157.
- [101] C. Kalbach. In: *Phys. Rev. C* 33 (1986), p. 818.
- [102] C. Kalbach-Cline. In: *Nucl. Phys. A* 210 (1973), p. 590.
- [103] F. C. Williams. In: *Nucl. Phys. A* 166 (1971), p. 231.
- [104] J.M. Akkermans and H. Gruppelaar. In: *Phys. Lett.* 157B (1985), p. 95.
- [105] H. Gruppelaar. In: *IAEA Advisory Group Meeting on Basic and Applied Problems on Nuclear Level Densities, (Brookhaven National Laboratory report, 1983)* (1983), p. 143.
- [106] C. Kalbach. In: *Phys. Rev. C* 71 (2005), p. 034606.
- [107] C. Kalbach. In: *private communication* (2007).
- [108] A.J. Koning and J.M. Akkermans. In: *Ann. Phys.* 208 (1991), p. 216.
- [109] C. Kalbach. “Systematics of continuum angular distributions: Extensions to higher energies”. In: *Phys. Rev. C* 37 (6 1988), pp. 2350–2370. DOI: 10.1103/PhysRevC.37.2350. URL: <https://link.aps.org/doi/10.1103/PhysRevC.37.2350>.
- [110] E Raeymackers et al. “Light charged particle production in the interaction of fast neutrons (25–65 MeV) with uranium nuclei”. In: *Physical Review C* 68.2 (2003), p. 024604.
- [111] A. A. Cowley et al. “Preequilibrium proton emission induced by 80 and 120 MeV protons incident on  $^{90}\text{Zr}$ ”. In: *Phys. Rev. C* 43 (2 1991), pp. 678–686. DOI: 10.1103/PhysRevC.43.678. URL: <https://link.aps.org/doi/10.1103/PhysRevC.43.678>.
- [112] W. Scobel et al. “Preequilibrium (p,n) reaction as a probe for the effective nucleon-nucleon interaction in multistep direct processes”. In: *Phys. Rev. C* 41 (5 1990), pp. 2010–2020. DOI: 10.1103/PhysRevC.41.2010. URL: <https://link.aps.org/doi/10.1103/PhysRevC.41.2010>.
- [113] J.W. Tepel, H.M. Hofmann, and H.A. Weidenmüller. In: *Phys. Lett.* 49B (1974), p. 1.
- [114] H.M. Hofmann et al. In: *Ann. Phys. (N.Y.)* 90 (1975), p. 403.

- [115] H.M. Hofmann et al. In: *Zeit. Phys.* A297 (1980), p. 153.
- [116] P.A. Moldauer. In: *Phys. Rev.* C14 (1976), p. 764.
- [117] P.A. Moldauer. In: *Nucl. Phys.* A344 (1980), p. 185.
- [118] J.J.M. Verbaarschot, H.A. Weidenmüller, and M.R. Zirnbauer. In: *Phys. Rep.* 129 (1985), p. 367.
- [119] S Hilaire, Ch Lagrange, and AJ Koning. “Comparisons between various width fluctuation correction factors for compound nucleus reactions”. In: *Annals of Physics* 306.2 (2003), pp. 209–231.
- [120] H. Gruppelaar and G. Reffo. In: *Nucl. Sci. Eng.* 62 (1977), p. 756.
- [121] M.B. Chadwick et al. In: *Phys. Rev.* C50 (1994), p. 996.
- [122] A.J. Koning and M.B. Chadwick. In: *Phys. Rev.* C56 (1997), p. 970.
- [123] P Romain, B Morillon, and AJ Koning. “Neutron actinides evaluations with the TALYS code”. In: *NEMEA-3 Neutron Measurements, Evaluations and Applications* 25 (2006), p. 113.
- [124] E.V. Gai et al. In: *Physics and Chemistry of Fission, I.A.E.A., Vienna* (1969), p. 337.
- [125] A. Mamdouh et al. In: *Nucl. Phys.* A679 (2001), p. 337.
- [126] A. J. Sierk. In: *Phys. Rev.* C33 (1986), p. 2039.
- [127] S. Cohen, F. Plasil, and W. J. Swiatecki. In: *Ann. of Phys.* 82 (1974), p. 557.
- [128] A. D’Arrigo et al. In: *Journ. Phys. G* 20 (1994), p. 365.
- [129] M. Sin, R. Capote, and A. Ventura. In: *Phys. Rev.* C74 (2006), p. 014608.
- [130] K.-H. Schmidt and B. Jurado. In: *Phys. Rev.* C82 (2011), p. 014607.
- [131] M.C. Duijvestijn, A.J. Koning, and F.-J. Hamsch. In: *Phys. Rev.* C64 (2001), p. 014607.
- [132] U. Brosa, S. Großmann, and Müller. In: *Phys. Rep.* 197 (1990), p. 167.
- [133] W.D. Myers and W.S. Swiatecki. In: *Ark. Fysik* 36 (1967), p. 593.
- [134] B.D. Wilkins, E.P. Steinberg, and R. R. Chasman. In: *Phys. Rev.* C14 (1976), p. 1832.
- [135] F. Vivès et al. “Investigation of the fission fragment properties of the reaction  $^{238}\text{U}$  (n,f) at incident neutron energies up to 5.8 MeV”. In: *Nuclear Physics A* 662 (2000), pp. 63–92. DOI: 10.1016/S0375-9474(99)00413-3.
- [136] S. Nagy et al. “Mass distributions in monoenergetic-neutron-induced fission of  $^{238}\text{U}$ ”. In: *Phys. Rev. C* 17 (1 1978), pp. 163–171. DOI: 10.1103/PhysRevC.17.163. URL: <https://link.aps.org/doi/10.1103/PhysRevC.17.163>.
- [137] T. C. Chapman et al. “Fission product yields from 6-9 MeV neutron-induced fission of  $^{235}\text{U}$  and  $^{238}\text{U}$ ”. In: *Phys. Rev. C* 17 (3 1978), pp. 1089–1097. DOI: 10.1103/PhysRevC.17.1089. URL: <https://link.aps.org/doi/10.1103/PhysRevC.17.1089>.
- [138] J. Kopecky et al. “Revisions and extensions of neutron capture cross-sections in the European Activation File EAF-3”. In: *ECN-C-92-051, July 1992* (1992).
- [139] S.F. Mughabghab. “Atlas of Neutron Resonances, 5<sup>th</sup> edition, Elsevier Publisher”. In: (2006).
- [140] S. Lemaire et al. In: *Phys. Rev.* C72 (2005), p. 024601.

- [141] J.A. Holmes et al. In: *At. Data Nucl. Data Tables* 18 (1976), p. 306.
- [142] M. Segawa et al. “Neutron capture cross sections of  $^{186}\text{Os}$ ,  $^{187}\text{Os}$ , and  $^{189}\text{Os}$  for the Re-Os chronology”. In: *Phys. Rev. C* 76 (2 2007), p. 022802. DOI: 10.1103/PhysRevC.76.022802. URL: <https://link.aps.org/doi/10.1103/PhysRevC.76.022802>.
- [143] H. Bateman. In: *Proc. Cambridge Phil. Soc.* 15 (1910), p. 423.
- [144] W. Rubinson. In: *J. Chem. Phys.* 17 (1949), p. 542.
- [145] W. R. Leo. “Techniques for Nuclear and Particle Physics Experiments”. In: *Springer Verlag, Berlin* (1994).
- [146] Nolte et al. In: *Phys. Rev. C* 36 (1987), p. 1312.
- [147] V. Avrieanu et al. In: *Phys. Rev. C* 49 (1994), p. 2136.
- [148] S. Goriely and J.-P. Delaroche. “The isovector imaginary neutron potential: A key ingredient for the r-process nucleosynthesis”. In: *Phys. Lett. B* 653 (2007), pp. 178–183.
- [149] Shin Okumura et al. “ $^{235}\text{U}(n,f)$  Independent fission product yield and isomeric ratio calculated with the statistical Hauser-Feshbach theory”. In: *Journal of Nuclear Science and Technology* 55 (2018), p. 1009.
- [150] M.J. Lopez Jimenez, B. Morillon, and P. Romain. In: *Ann. Nucl. Energy* 32 (2005), p. 195.
- [151] B. Morillon, H. Duarte, and P. Romain. “New BRC Neutron Evaluations of Actinides with the TALYS code : Modelization and First Validation Tests”. In: *Proceedings of the 4th International Workshop on Nuclear Fission and Fission-Product Spectroscopy, Cadarache, France, May 13-16, AIP Conference Proceedings, A. Chatillon, H. Faust, G. Fioni, D. Goutte, H. Goutte [Eds]* (2009), p. 79.
- [152] D. E. Cullen. *PREPRO 2018 - 2018 ENDF-6 Pre-processing codes*. Tech. rep. Technical report IAEA-NDS-39 (Rev.18), IAEA (2018). 2017.
- [153] AJ Koning and MB Chadwick. “Microscopic two-component multistep direct theory for continuum nuclear reactions”. In: *Physical Review C* 56.2 (1997), p. 970.
- [154] A.J. Koning et al. “TENDL: Complete Nuclear Data Library for innovative Nuclear Science and Technology”. In: *Nuclear Data Sheets* 155 (2019), p. 1.



## A. Log file of changes since TALYS-1.0

Development of TALYS since release of TALYS-1.0

- December 21, 2007

\*\*\*\*\* Release of TALYS-1.0 \*\*\*\*\*

- January 11, 2008

Repaired little bug for angular distributions in isomeric reactions in  
angleout.f, discovered by Neil Summers, LLNL

- January 17 2008

Added protection for average radiative width in resonancepar.f  
gamgam is never larger than 10.  
Only required for extreme cases.

- January 22 2008

Suggestion from Stephane Goriely:

Added HFB deformations, consistent with the HFB masses

Added keyword deformmodel to choose between Moeller deformations  
(deformmodel 1) and HFB deformations (deformmodel 2)

- March 4 2008

Suggestion by Emmeric Dupont: small format change for writing D0

in partable.f

- March 4 2008

Advice of Stephane Goriely:

Removed unphysical behavior of pre-equilibrium gamma emission for extremely neutron rich nuclides. Changed emissionrate.f, emissionrate2.f and preeqttotal.f

- March 4 2008

Added very small preequilibrium contribution to inelastic cross section, in preeqcorrect.f, to prevent discontinuous inelastic cross sections.

- March 11 2008

Emmeric Dupont spotted a small omission in partable.f for the second GDR Lorentzian. Solved.

- March 13 2008

Removed an error for Kopecky-Uhl gamma-ray strength functions. They are now computed at every incident energy since the model is incident energy dependent.

- March 29 2008

Introduced keywords aadjust, gnadjust, gpadjust, gamgamadjust (all default 1.) to make relative changes to level density and other parameters. This avoid looking up the default values of a, etc. and changing them.

- April 8 2008

Stephane Hilaire found an important initialization problem for the case 'projectile 0'; the maximum l-value was zero instead of gamma. This is repaired in densprepare.f

- April 18 2008

Changed a few 0 energies in the file hbstates.eo by energy 0.01 as given in Maslov's report. This was pointed out by Emmeric Dupont.

- May 5 2008

---

Error with 'segment' variable in spectra.f. Found by Ian Thompson.

- May 19 2008

Added extra if statement for calculation of radiative widths in nuclides.f. Test case 9 crashed for Vittorio Zecca.

- May 24, 2008

Corrected gamma-ray branching ratios with electron conversion, following advice of Olivier Bersillon and Arjan Plompen.

Default is now 'electronconv y'.

Also the branching ratios in the levels database have changed.

- June 20 2008

Pascal Romain found a problem with indexing optical model files, for cases where more than 1 external potential per particle is given. Therefore, the array index for omplines has been enlarged to 3 dimensions.

- June 30 2008

Small normalization problem solved for pre-equilibrium. The pre-equilibrium cross section could be slightly negative, and this is repaired in preeq.f, preeqtot.f and binary.f. Found by Stephane Hilaire.

-July 29 2008

Solved a bug found by Arnd Junghans. Small change in basicinitial.f where lmax is set to gamma for gamma's. This caused problems for calculations with ompall y.

- August 1 2008

Added the possibility to use microscopic particle-hole configurations, as generated from Stephane Hilaire's HFB software. A directory structure/density/ph has been added in the nuclear structure database, which contains 2-component particle-hole state densities for many configurations per nuclide. A new keyword has been introduced, phmodel, where phmodel 1 is used for the (default) analytical densities and phmodel 2 is used for the tabulated densities.

Various subroutines were generalized to accommodate the choice between these different types of ph-densities. A subroutine phdensitytable.f was added to read in the values from the structure database.

Inside `phdens.f` and `phdens2.f` there is now a choice for the model.

- August 4 2008

Added a keyword `endfecis`, similar to `inccalc` and `eciscalc`, such that calculations for a more precise ENDF-6 energy grid can be kept from a previous run.

- August 14 2008

Added a keyword `omponly`, so that Talys runs can be done in which ONLY an optical model calculation is done, i.e. Talys simply acts as a driver for ECIS.

- August 20 2008

Stephane Goriely sent a table with ground state and spins predicted by HFB calculations. This has now been added to the nuclear structure database. If the ground state spin and/or parity is not known experimentally, we now take it from the HFB table instead of a simple  $0^+$  (even) or  $0.5^+$  (odd) assignment.

- August 20 2008

A keyword `jlmmode` has been added to enable different normalizations for the imaginary potential, as explained in

‘‘The isovector imaginary neutron potential:

A key ingredient for the r-process nucleosynthesis’’,

S. Goriely, , J.-P. Delaroche, Physics Letters B 653 (2007) 178

Obviously, this keyword only works if `'jlmomp y'`.

- August 20 2008

On advice of Stephane Goriely, the possibility to include Pygmy resonance parameters for gamma-ray strength functions was included. Three new keywords `'epr'`, `'gpr'` and `'tpr'` were defined.

- September 6 2008

Repaired small error with `nbins` in `input2.f`

- September 11 2008

Added correction factor for gamma-ray strength function to take into account isospin -forbidden transitions, and the associated `fiso` keyword. Also included strength function model 5 for Goriely's hybrid strength



model.

Added a keyword `micro` to do a completely microscopic calculation, i.e. without any phenomenological models or adjustment

- September 21 2008

Added a keyword `'best'` to adopt a set of adjusted best parameters for a particular reaction. Such parameter sets can be stored in the `structure/best` directory.

- October 17 2008

Added the flexibility to use user-defined thermal capture,  $(n,p)$ , and  $(n,\alpha)$  cross sections. Also created a new subroutine `thermalxs.f` to handle these values there.

- November 1 2008

Changed the default for multiple preequilibrium reactions, i.e. `mpreeqmode 2` is now the default. This means that the transmission coefficient method of Chadwick is used instead of multiple exciton model. This may increase the speed of TALYS at higher energies by several factors.

- November 17 2008

Corrected small database reading bug for `strength=5` in `gammapar.f`. Discovered by Stephane Goriely.

Stephane Goriely discovers a Lab/CM bug in his `stellarrate` subroutine which may be non-negligible for light nuclides. Corrected version adopted.

- December 1 2008

Solved a problem encountered by Arjan Plompen for JLM based exciton model transitions in `bonetti.f` (fluctuating behaviour at low outgoing energies). For negative energies, the JLM results are normalized to the phenomenological imaginary potential. This only applies if `'preeqmode 3'` and `'jlmomp y'`.

- December 2 2008

External optical model input files can now also be made adjustable through the `rvadjust`, etc. keywords. Changed `opticaln.f`, ... `opticala.f` for this.

- December 16 2008

Implemented several corrections and options by Pascal Romain for fission calculations:

- Increased maximum possible number of rotational levels to 20 in checkvalue.f
- Possibility to read in head band and class2 states for 3 barriers in fissionpar.f
- Increased maximum number of incident energies to 500 in talys.cmb
- Increased maximum number of rotational states to 700 in talys.cmb
- Updated the subroutine tfission.f with improved equations for the treatment of class-II states.
- Re-introduced the keyword colldamp. If 'colldamp y' fission calculations according to the method of Pascal Romain of CEA Bruyeres-le-Chatel are performed. This means different options for the spin cutoff parameter and the collective enhancement. Also different defaults for some level density parameters are used.

- January 13 2009

TALYS can now produce a table with average resonance parameters, for the description of the Unresolved Resonance Range (URR). A keyword 'urr' has been added. If 'urr y', a file called urr.dat will be produced, which for each incident energy contains the relevant neutron and gamma strength functions S, mean level spacings  $D(J,l)$  and decay widths  $G_n(J,l)$  and  $G_g(J,l)$ . To enable this, the subroutines radwidththeory.f and spr.f are now called for every incident neutron energy.

- February 9 2009

Amir A Bahadori found a problem for emission spectra in a natural element calculation. The emission energy grids for the various isotopes were out of sync. I revised the subroutine natural.f to solve this.

- February 20 2009

Added the possibility to normalize certain reaction channels in TALYS directly to experimental or evaluated data. A new subroutine normalization.f was written and the keyword 'rescuefile' was added. With a "rescue file" normalization factors can be given on an incident energy grid, after which the cross sections are normalized. The difference is then corrected for in the elastic cross section. Use with care.

---

- March 20 2009

Change default of including class 2 states. The default is now class2 y.

- March 20 2009

Corrected little error in densityout.f. The name of the nucleus was written incorrectly to the ld\* files

- April 1 2009

Change the default for deformed OMP calculations: If the incident particle is a neutron or a proton, only for that particle (as projectile and ejectile) a deformed OMP is adopted. The default for the other particle is set to false. As usual this can all be overwritten with the 'rotational' keyword.

- April 6 2009

Ian Thompson found unphysical oscillatory behavior for Al26(n,n') excitation functions. The cause of this lies in a too large pre-equilibrium alpha contribution. Therefore, I changed some C/A terms in preeqcomplex.f into  $C/(\max(A,40))$ .

- April 9 2009

Added isotopic specification for natural targets, for recoil output files on advice of Reto Trappitsch. Changed recoilout.f and natural.f.

- April 14 2009

For very short-lived target nuclei the definition of 'isomer' is overwritten with the lifetime of the target level. This means that if e.g. the lifetime of the target nucleus is 1 ms, then all other levels with a lifetime longer than 1 ms will be treated as "isomer". This overwrites the isomer default or value given in the input. Change made in levels.f.

- April 16 2009

Changed maximum value of enincrow in grid.f to prevent array overflow. Only relevant for very exotic nuclei.

- April 23 2009

Updated preeqcomplex.f after discussions with Connie Kalbach on complex

particle pre-equilibrium emission. Changed lab energy into CM energy at a few formula's

- April 23 2009

Made a correction in recoilinit.f. For reactions with very large Q-values, e.g. Li-7(d,n) the recoil energy grid was not well defined. Problem encountered by Markus Lauscher.

- April 29 2009

Added the latest mass and deformation tables from Stephane Goriely, as published in Phys Rev Lett 102, 152503 (2009).

- April 29 2009

Included default normalization factors for microscopic fission calculations as published in Goriely et al Phys Rev C79, 024612 (2009).

- April 29 2009

Introduced logarithmic interpolation for tabulated level densities, instead of linear. Changed densitot.f and phdens.f, phdens2.f.

- May 5 2009

Updated preeqcomplex.f after discussions with Connie Kalbach: included pairing correction

- May 5 2009

Generalized nuclear mass, deformation and discrete level tables. Nuclear masses are now read from different subdirectories instead of from one file. This will allow easier addition of alternative mass models in the future.

Also the number of theoretical deformation parameter databases is extended. For both masses and deformation parameters, HFB calculations based on the Gogny D1M force are now also an option.

For discrete levels, HFB based ground state spins and parities are added to the discrete level database for nuclides for which no experimental information exists.

The nuclear structure database, masses.f deformation.f and levels.f were changed for this.

- May 10 2009

---

Further changes to the masses, deformation, and ground state property tables. I removed the 'deformmodel' keyword again (introduced after the release of TALYS-1.0, so no problems with angry users). It is obvious that theoretical deformation parameters should be taken from the same model as that of the masses. The structure/deformation directory is also cleaned. It now only consists of the subdirectory exp/ where the coupling schemes for coupled channels (and possibly deformation parameters) are given. The theoretical deformation parameters are now found in the mass tables. However, they can be overruled by other values in the deformation/exp files. In most cases however, only the coupling scheme and vibrational deformation parameters (from Raman and Kibedi) remain.

For all (or at least most) stable nuclides above  $Z=30$  and actinides a coupling scheme is provided in deformation/exp. This will soon be finalized for all nuclides, enabling default coupled-channel calculations.

I also changed the numbering of the massmodels.

- 0: Duflo-Zuker (analytical formula for the masses, no deformation and ground state spins, instead, take these from 'massmodel 2')
- 1: Moller
- 2: HFB-17 (Skyrme force): this is the default when exp data is not available
- 3: HFBD1M (Gogny force)

- May 11 2009

The keywords "addelastic" and "adddiscrete", to fold in direct peaks in secondary spectra, for easy comparison with experiment, have been made energy dependent, similar to that of e.g. the "preequilibrium" keyword. Possibilities are now e.g.

addelastic y  
addelastic n  
addelastic 25.

and similarly for adddiscrete.

- May 18 2009

After one year, I ran the monkey code again, and included all new keywords for testing. This means running Talys with random input files with about 300 keywords, and the requirement that it does not crash, run out of array boundaries etc. A few minor problems were solved in several subroutines.

- May 19 2009

Introduced new keyword strengthM1, to vary between M1 strength functions. The default is strengthM1 = 2 (RIPL-2 prescription) while a value of 1 corresponds to the older prescription of RIPL-1.

- May 22 2009

Adopted the Duflo-Zuker formula from 1996 (from the RIPL-2 website) and replaced the older version. This is only used for masses outside the masstables.

- May 22 2009

Adopted the suggestion of David Perticone to use the longest-lived nuclide as default for natural element calculations for elements without stable isotopes. In other words 'element Tc' in combination with 'mass 0' will no longer give an error message but instead A=98 will be adopted.

- May 22 2009

Added the isotope number to double-differential spectra in the case of natural targets. David Perticone discovered that this was missing.

- June 9 2009

Added the possibility to give 'nbins 0' in the input. In this case, TALYS tries to adopt a clever number of bins, depending on the incident energy, rather than a constant number of bins (the default).

- June 9 2009

Since all theoretical deformations are now in the mass directory, there is only one set of data left in the structure/deformation/, the subdirectory exp/ is therefore removed, and the structure/deformation/ directory now merely contains the coupling schemes. Added many coupling schemes in this directory

- June 9 2009

Removed an input inconsistency for D0. Both in the input and from the structure database they are now read in keV.

- August 5 2009

---

Improved some of the ECIS flags for low incident energies for deformed nuclides, according to suggestions of Pascal Romain. Also introduced the keyword 'soswitch' to set the energy where deformed spin-orbit calculations and sequential iterations begin.

Introduced the keyword 'coulomb' to (de-)activate Coulomb excitation calculations by ECIS

Reprogrammed the energy dependence of fstrength.f (again!), after discussion with Pascal Romain and Roberto Capote.

- August 24 2009

Solved a problem with angular distributions for reactions on isomeric states. Incorrect levels were read in (in other words, elastic and inelastic angular distributions were mixed.)

- October 28 2009

Included new thermal cross section database of Mughabghab in structure database. This will change the low energy extension down to thermal energies.

- December 1 2009

Added new keywords rvadjustF, avadjustF, to enable energy-dependent geometry parameters for the optical model. This is useful for local optimization of the optical model. We can now slightly deviate from constant values, in a smooth manner, and for a restricted energy range given by the user.

- December 11 2009

Added a keyword bestpath to allow a different directory to store input files with best TALYS input parameters

- December 17 2009

Smoothened microscopical particle-hole states read from table.





## B. Log file of changes since TALYS-1.2

Development of TALYS since release of TALYS-1.2

- December 23, 2009

\*\*\*\*\* Release of TALYS-1.2 \*\*\*\*\*

- January 6, 2010

Changed some of the defaults for 'micro y' (fully microscopic option).  
Among others, the phmodel is kept to 1. Also 'strength 4' if 'micro y'.

- January 8, 2010

Tiny correction by Stephane Goriely

in levels.f, line 153

```
    if (massmodel.ne.2) then
```

has been changed into

```
    if (massmodel.le.1) then
```

since in the case of D1M (massmodel=3), a spin and parity is  
also assigned.

- January 21, 2010

Solved a problem with Elow (eninclo in the source code) found by Dimitri  
Rochman: If the highest incident energy is smaller than Elow, Elow is set  
to zero (otherwise there would be no output of cross sections).

- February 1, 2010

Corrected an indexing problem for exclusive channels in channels.f  
For exotic nuclei sometimes the exclusive channel does not open up  
at the right moment, so the idnum index is incorrect. This caused  
a wrong extrapolation at thermal energies for Th-150(n,n')

Added a small output extension for the branching ratios of  
residual products

- February 24 2010

Added a correction for the indexing of exclusive channels  
(see also Feb 1). In the case of using 'rescue' files the capture  
channel may appear to be closed while the (n,gf) channel, which is  
not normalized, remains open. Variables chanopen, in channels.f,  
and chanexist, in strucinitial.f, are now reinitialized to true  
for the capture channel.

Nikolaj Filatov found a few small bugs, or potential problems,  
in the Brosa subroutines. for fission yield calculations.  
These have now been corrected

- March 2 2010

Solved a precision problem in molprepare.f for width fluctuation  
calculations Very small numbers were multiplied with very large numbers,  
leading to NaN's in some cases.

The thermal (n,a) cross section in the thermal database was a factor a  
million too high. This is corrected.

- March 7 2010

Erich Ormand found a Bug in Fu's pairing correction, important if  
pairmodel 1 (the default is pairmodel 2). The formula contains a factor  
(U/delta)\*\*0.68 and not (U/delta\*0.68) as coded. Formula corrected.

- March 11 2010

Solved an old problem, first mentioned by Arjan Plompen and now also by  
Alexander Konobeyev: Different compilers produce different answers for  
the gamma spectrum at low energies. This was visible in sample case 6.  
Solved this by a small correction in specemission.f and compemission.f.

---

After advice of Yan Shengquan, I added a print of the compound nucleus formation cross section per spin and parity value. This will only be done if flagcheck y.

- March 12 2010

On the advice of Stephane Hilaire, S(Zcomp,Ncomp,1) was changed into S(Zcomp,Ncomp,k0) in the Kopecky-Uhl generalized Lorentzian part of fstrength.f

- March 26 2010

Separated subroutine preeqcomplex into subroutines stripping, knockout and breakup.

- March 28 2010

Implemented Kalbach's new model for break-up reactions, documented in a FENDL-3 report (March 2010). For this, the Cbreak keyword was introduced, similar to Cknock and Cstrip. The new model is in subroutine breakup.

- March 30 2010

After a problem discovered by Mihai Horoi, and on the advice of Stephane Goriely, the limits to take transmission coefficients and cross sections (popeps, etc.) into account are lowered for astrophysical calculations.

- April 9 2010

Repaired a problem found by Tak Pui Lou. If 'compound n', as in sample case 9, no subroutine or function calculating gamma ray strength functions should be called. This was ensured by changing an if statement near the end of incident.f

- April 14 2010

Erich Ormand found a bug for the exciton model emission rates for the use of pairmodel=1 (Fu's pairing correction). The exciton number of the residual system should be taken in preeqpair, when appropriate. Changed emissionrate.f and emissionrate2.f for this.

- May 20 2010

Added the recoil flag to the endf.inf file, for ENDF evaluations

- June 25 2010

Ronald Murphy discovered an error for angular distributions to discrete states for reactions other than inelastic scattering. This problem occurs for low incident energies of complex particles such as alpha's. Subroutine angdis.f was corrected for this.

- June 28 2010

Ronald Murphy discovered another error for continuum angular distributions. An if statement was removed from spectra.f

- September 2 2010

Generalized input of characters: filenames and other strings can now also be given in uppercase

- December 7 2010

An input file of Mahdi Bakhtiari and Mahdi Sadeghi led me to a problem on interpolation of (input) tabulated OMP parameters with the optmod keyword. Solved in subroutine omppar.f. Also, they spotted an error in the manual on the outexcitation keyword that I corrected, and it TALYS now stops if a microscopic level density model (ldmodel 4,5) is used in combination with colenhance y (which only works for phenomenological models).

- December 27 2010

On the advice of Stephane Goriely, I wiped a problem with the JLM OMP under the carpet: for very neutron-rich nuclides the JLM potential by Bauge and Delaroche may produce negative reaction cross sections. Instead of stopping the TALYS calculation directly, we now set the reaction cross section to zero. Changed incidentread.f and directread.f for this.

- December 29 2010

Adopted Stephane Goriely's double folding alpha potential. A new subroutine foldalpha.f was added, and now alphaomp runs from 1 to 5. The default for the alpha OMP remains unchanged.

- January 3 2011

Extended the flexibility for using level densities. It is now possible,

---

in one and the same calculation, to use different level density models for different nuclides. If the input file contains e.g.

```
ldmodel 5
ldmodel 24 52 1
ldmodel 24 53 2
```

then ldmodel 5 will be used for all nuclides, with the exception of Cr-52 and Cr-53 for which ldmodel 1 and 2, respectively, will be used. The level density model thus becomes an adjustable parameter of integer type.

- January 5 2011

Changed the meaning of the keyword `adddiscrete`. For  $(n,p)\dots(n,\alpha)$  reactions discrete state cross sections are now always added to the energy spectrum of the cumulative  $(n,xp)\dots(n,xa)$  spectra.

- January 10 2011

Changed a small cross section limit in `preeqcorrect.f` to allow calculation of very small alpha-induced cross sections of astrophysical interest

- February 3 2011

Solved an error in the extrapolation of partial  $(n,p)$  cross sections to thermal values.

- February 22 2011

Solved a consistency error in the GSM in `densitypar.f`, found by Heedong Choi. After determination of the critical level density parameters, the global l.d. `alimit` should not be redefined.

- April 13 2011

Changed the lower input boundary for the Pygmy resonance width to 0.1 MeV, due to a problem found by Samantha Hammond.

- May 10 2011

Added 4 global deuteron OMP's, since the default model coming from standard Watanabe folding does not perform well. A keyword `'deuteronomp'` has been made to make a choice between the models:

```
deuteronomp 1 --> Watanabe potential
deuteronomp 2 --> Daehnick potential
```

deuteronomp 3 --> Bojowald potential  
deuteronomp 4 --> Han potential  
deuteronomp 5 --> Haixia An potential

- May 11 2011

Added flexibility in the OMP adjustment factors. If necessary, all geometrical parameters can now be independently varied (e.g. we no longer necessarily have  $rw=rv$ , etc.)

- May 11 2011

Adopted new egridastro.f subroutine from Stephane Goriely for a finer energy grid in astrophysical calculations

- June 27 2011

On the advice of Kevin Kelley, I allowed the combination 'ecissave y' and 'eciscalc n'. TALYS now checks whether transmission coefficients are available from a previous run. If not, it will create them.

- June 30 2011

Corrected the treatment of the URR: there was an error in the rules for possible angular momentum combinations in urr.f, discovered by Paul Koehler. I also included the calculation of the average gamma width for the URR in the correct place, as suggested by Gilles Noguere. All URR related actions in TALYS can now be found by searching for the variable flagurr.

- July 1 2011

Included the possibility to calculate the effective cross section for integral activation measurements, by folding the excitation functions by an experimental flux. In talys/structure/flux, we have stored more than 40 spectra, coming from the EASY package, which have been used in past activation benchmarks. Such benchmark activities should eventually be performed using (processed) nuclear data libraries, especially for reactions containing a low energy resonance range, but it is helpful to get an idea of the performance directly with TALYS as well. A new keyword 'integral' has been introduced for this.

- September 19 2011

Adopted Gilles Noguere's subroutines for the calculation of unresolved resonance parameters. The urr keyword now acts like the

---

widthfluc keyword: it can be y, n or a value between 0 and 20 MeV for the on-set of the URR calculations. For more flexibility, also the keyword lurr (maximal l-value for URR calculations) and urrnjoy (to normalize capture and fission widths according to the NJOY URR method). The URR subroutines are urr.f, urrout.f, csunr2.f, unfac.f, gnrl.f and strengthfunc.f

- September 19 2011

In addition to making ldmodel nuclide-dependent, the same was done for the colenhance keyword, i.e. for every nuclide in the chain the collective enhancement for the level density can be turned on and off.

- November 17 2011

Added the possibility to print the direct, pre-equilibrium and compound components of each cross section in the output files. This can be enabled with a new keyword: components

- December 22 2011

Added the possibility to calculate Maxwellian averaged cross sections (MACS). This was already implicitly available for the calculation of astrophysical reaction rates, but now with the keywords 'astroE' and 'astroT' the MACS can be calculated at a user-specified energy, e.g. 30 keV, or temperature.





## C. Log file of changes since TALYS-1.4

Development of TALYS since release of TALYS-1.4

- December 28, 2011

\*\*\*\*\* Release of TALYS-1.4 \*\*\*\*\*

- December 30, 2011

Introduced non-equidistant binning for the excitation energy grid. The excitation energies can now be given on a logarithmic grid, and this has also become the default. It allows to use less excitation energy bins while not losing precision. The old situation, of TALYS-1.4 and before, can be invoked with the new 'equidistant' keyword, but from now on the default will be 'equidistant n'.

Especially for cross sections above 100 MeV the improvement is significant: smoothness and more reliable absolute values.

TALYS can now technically be used up to 600 MeV, though I did not validate that with experimental data yet.

- January 2, 2012

Added the extra MT numbers 152-200, introduced in the ENDF-6 format in 2010, in the normalization subroutine.

- January 9, 2012

Multiplied the normalization factor for stripping reactions for

helions with a factor of 5 to better fit (n,h) data.

- January 16, 2012

Added the statement "Positive parity" to the level density output file in case of parity-dependent level densities, on the advice of Goran Arbanas.

- January 17, 2012

Solved a problem reported by Patric Granholm.  
At energies far below the Coulomb Barrier, some excitation functions showed discontinuities. This was due to a too strict choice of pre-equilibrium spin distributions in subroutine binary.f.  
A popepsA constant is now added.

- January 17, 2012

Turned a few TALYS-errors into warnings (TALYS no longer stops) in integral.f. Also changed the naming of the associated data into talys/structure/integral in which experimental data and fluxes can be found.

- January 21, 2012

Patric Granholm's problem of jan 17 motivated me to extend the 'preeqspin' keyword. There are now 3 possibilities:  
preeqspin 1 (equivalent to the old 'preeqspin n'): the pre-eq spin distribution is made equal to the relative spin-dependent population after compound nucleus emission  
preeqspin 2: the spin distribution from total level densities is adopted  
preeqspin 3 (equivalent to the old 'preeqspin y'): the pre-eq spin distribution is based on particle-hole state densities  
The input is backward compatible, i.e.  
preeqspin y is transformed into preeqspin 3  
preeqspin n is transformed into preeqspin 1  
Total number of possibilities: preeqspin can be equal to y,n,1,2,3

- February 1, 2012

Added more adjustment flexibility. With the tljadjust keyword the transmission coefficients for the outgoing channel can be changed in a certain energy range.

- February 3, 2012

---

Added more adjustment flexibility. With the 'branch' keyword the branching ratio for any discrete level in the database can easily be overwritten in the user input. This may be necessary to adjust isomeric cross sections.

- February 13, 2012

Corrected a bug found by Yuki Watanabe: for charged-particle elastic and inelastic scattering the compound components are not written to output. Corrected `angdis.f` for this.

- February 13, 2012

Fixed a problem with the calculation of  $D_0$  for the case of incident photons, (used for level density studies) found by Stephane Goriely. I changed `dtheory.f` for this.

- February 14, 2012

Removed a bug in `rldm.f`, introduced just before the release of TALYS-1.4: A `min()` function was introduced to prevent an array running out of bounds. However, this gives huge fission cross sections for neutrons on subactinides (or any other case where the RLDM is needed), as discovered by Stephane Goriely.

- February 16, 2012

On the advice of Stephane Goriely, I made a change in `fissionpar.f`: if no fission barriers are found in the tables with phenomenological values, first the tables of Mamdouh (fismodel 2) are scanned for values, before an analytical model is used.

- February 20, 2012

After a question by Yuki Watanabe, I decided to introduce the 'cpang' keyword, to enable the calculation of discrete level compound nucleus angular distributions for incident charged particles

- February 25, 2012

Changed the order of reading lines from files of the `best/` directory such that the "best" parameters are now also used for natural targets.

- March 6, 2012

Added radiative capture model for direct and semi-direct capture cross

section. For this the 'racap' keyword was made. By default, 'racap n'.

- March 11, 2012

Added new microscopic level densities, this time generated from temperature-dependent HFB calculations and the Gogny D1M nucleon force. This option is now 'ldmodel 6'

- March 27, 2012

Automatically increased the number of included discrete levels for a highly excited isomer.

- March 27, 2012

To increase fitting flexibility I added the keywords egradjust, ggradjust, sgradjust, epradjust, gpradjust, spradjust, E0adjust, Exmatchadjust, Tadjust, fisbaradjust and fishwadjust, all with a default value of 1.

- March 27, 2012

About 70 keywords can now have energy-dependent values. After the keyword and its value are given in the input, an energy range can be specified over which the corresponding model parameter is altered.

- March 29, 2012

Incident energies can now be given in any order, not necessarily in ascending order. They will be sorted first. Also double energy points will be removed. TALYS will thus no longer stop when this is provided.

- April 11 2012

Solved an output bug found by Mahdi Baktiari. At energies far below the Coulomb barrier the direct elastic angular distribution for charged particles is not copied into the total column.

- April 21 2012

Changed nexout into nexout2 in do loop around line 70 in specemission.f That error has been there for several years.

- April 26 2012

Generalized a few energy grids, so that TALYS can be used up to 600 MeV.

- May 7 2012

Vivian Demetriou found a problem in the alpha folding potential which I corrected.

- August 12 2012

Added keyword 'production' for (medical) isotope production.  
Added decay data subroutine.

- August 15 2012

Extended nucleon OMP to 1 GeV. For this the keywords Vinf, Ejoin, w3adjust and w4adjust give some extra flexibility. TALYS can now technically be used up to 1 GeV.

- August 28 2012

Added all other routines for medical isotope production. With the 'production' keyword, excitation functions are automatically transferred into isotope production rates in MBq. Various extra keywords are included for flexibility.

- September 11 2012

Corrected an error for particle multiplicities found by Pedro Arce. The yields should be defined relative to the nonelastic cross section and not to the reaction cross section. This makes a difference at low energies and will change charged-particle nuclear data libraries, which use this information.

- September 15 2012

Added possibility to locally adjust the OMP parameters for outgoing channels only, with the keyword 'incadjust'. If 'incadjust n' then OMP adjustments are not performed for the incident channel. This gives more fitting flexibility. Physical argument is that the magnitude of the approximation to regard inverse channels as occurring from the ground state, is unknown. (temperature dependent OMP). The default is however still 'incadjust y'.

- September 15 2012

Arnd Junghans discovered a reading error in the RIPL file for S0. Some of the values do not have a decimal point, resulting in a factor

of 100 difference for some nuclides. Fortunately S0 is not used, apart from reference, but the error is now corrected anyway. At his advice, I also removed the printing of the neutron strength function at the wrong place (it was in the middle of the gamma data).

- October 6 2012

Introduced a keyword Liso to enforce the isomeric number of the target (by default it is determined from the discrete level database).

- October 15 2012

Added a keyword fisfeed to output the fission per excitation energy bin. This allows to couple TALYS with a fission yields code, like GEF. To enable this, a few outlets in multiple.f had to be made.

- December 10 2012

Changed a statement in the output about level densities. The 'total' column is given per parity. (issue discovered by Paul Koehler).

- December 21 2012

Revised the direct radiative capture model together with Stephane Goriely. Three new keywords were introduced: ldmodelracap, sfexp and sfth.

- December 21 2012

The resulting intermediate code version is called TALYS-1.50

- March 19 2013

Complemented RIPL discrete level database with levels calculated from microscopic densities. A new keyword 'disctable' was introduced to include or exclude such additions. The default is now 'disctable 1', i.e. RIPL discrete levels file + levels added up to level number 100 if missing from RIPL.

- April 27 2013

Changed 'ecis06' into 'ecis' for filenames etc., to make the filenames independent of the version of ECIS.

- May 18 2013

---

Increased boundaries for keyword `massexcess` from 0.500 to 0.600 a.m.u.

- May 20 2013

Put an if statement around loop 150 in `comptarget.f` to solve an NaN problem found by Satoshi Chiba for astrophysical calculations.

- May 21 2013

Enforced that the non-elastic cross section is non-negative by putting a max around `xsnonel` in `binary.f`

- May 21 2013

Changed an `.lt.` into an `.le.` in `tripathi.f` which solves a strange ifort compilation problem.

- May 27 2013

Problem found by Mahdi Bakhtiari for elastic scattering angular distributions for natural targets. For charged particles these were not summed to the natural value. Corrected `natural.f`

- June 4 2013

Wrote more parameters to `parameters.dat` file

- June 24 2013

Corrected two problems found by Peter Mohr: if `cpang y` then the charged-particle compound elastic angular distribution should be anisotropic (though symmetric). Corrected `angdis.f` for this. Also, for incident charged particles, direct discrete contributions are not taken into account if a coupling scheme is given. The default for the 'spherical' keyword is changed for this.

- June 30 2013

Added call for GEF subroutine, created by Vasily Simutkin and Michail Onegin, to `massdis.f` to use the Schmidt-Jurado model for fission observables. Introduced the `fymodel` keyword to choose FY model `fymodel 1= Brosa`, `fymodel 2 = GEF`. `fymodel 1` is now also the default.

- August 18 2013

As requested by Paul Koehler, added the average alpha and proton widths in the output of the URR subroutines.

- August 22 2013

Disabled the renormalization of the URR by the NJOY method, as a default at least, to avoid unstable results for the competitive width.

- September 24 2013

Added radiounit and yieldunit keywords to specify units for medical isotope production

- October 11 2013

Repaired an error in optical model adjustment, discovered by Vlad and Marilena Avrigeanu, the Coulomb radius was not adjusted with radjust.

- December 1 2013

Added new version of GEF subroutine, created by Vasily Simutkin and Michail Onegin, called geftalys.f. Fission yields, P(nu), nubar, and nu as function of Z and A are added to the output. Introduced two extra keywords: outfy, to give detailed output of the GEF calculation, and gefran, for the number of random events to be fed to GEF.



## D. Log file of changes since TALYS-1.6

Development of TALYS since release of TALYS-1.6

- December 23, 2013

\*\*\*\*\* Release of TALYS-1.6 \*\*\*\*\*

- January 10, 2014

Stephane Goriely found an error in densitytable.f where for ldmodel=6 no parity-dependence was included. This is corrected.

- January 23, 2014

Introduced extra adjustment to pairing shift with keyword Pshiftadjust. This works for the Backshifted Fermi Gas model and avoids looking up the actual value for Pshift for adjustment.

- January 26, 2014

Extended the range of gamgamadjust. Lower limit is now 0.01.

- February 10, 2014

Fixed a problem for FY calculation spotted after calculating a test case for Charlotte Duchemin.

An extra check is done in massdis.f to speed up the FY calculations

considerably. Charlotte also noted that not all FP files are present. A correction in `massdisout.f` was made. In addition, total FP cross sections per A (cumulative cross section) are now given.

- February 13, 2014

Re-introduced the possibility to fit average radiative width for masses below  $A=40$ . This is now performed if the `gamgamadjust` keyword is explicitly given in the input.

- February 21, 2014

Added low energy resonance cross sections to the output. For this, the keyword 'resonance' was introduced. If resonance y, TALYS will read in resonance parameters, from various possible sources, and call the RECENT code of Red Cullen's PREPRO package, which is included as a subroutine. Low energy pointwise resonance cross sections will be added to channels like total, elastic, fission and capture. Choices for various libraries can be made with the `reslib` keyword, which can be equal to default, `endfb7.1`, `jeff3.1`, `jendl4.0`

- March 10, 2014

Extended the resonance option with the SIGMA1 code, for resonance broadening, and GROUPIE, for groupwise cross sections, of the PREPRO package. Introduced the new keywords 'Tres' and 'group' for this. Giving e.g. `Tres 100.`  
`group y`  
will produce groupwise cross sections at 100 Kelvin Doppler broadening.

- April 1, 2014

Sort of solved an indexing problem for exclusive channels, found after a crash (indeed!) of TALYS with an input case of Arjen van Vliet. When `maxchannel` is high, e.g. 8, so many channels are open that an indexing problem occurs. The ultimate correction has not been found but a small change in `channels.f`, when we jump out of the loop if the index reaches `numchantot-10`, seems to do the trick.

- April 1, 2014

Solved a problem discovered by Ismail Sarpun. The GSM model was insensitive to variations of the level density parameter  $a$  in the input. This is because the GSM model contains a critical level density parameter

---

a\_crit that is solved by iteration.  
Now 'a' from the input is set equal to a limit.

- May 29, 2014

Solved a problem for medical isotope production found by Will Webster. TALYS was not protected against using a very fine incident energy. This has now been solved and up to 2000 incident energies can be used. Also some indices for the radioactive decay chain were wrongly assigned, which is solved now.

- June 21, 2014

Added the absorption cross section to the output. This was missing so far. The absorption cross section is the sum of all partial cross sections that do not have the initial particle (usually a neutron) in the exit channel. While it is not needed for any application, EXFOR contains some data sets that can be compared with this.

- July 7, 2014

Added a keyword bestend to insert a structure/best file at the end of the input file instead of at the beginning. This gives more flexibility for overruling parameter values.

- July 14, 2014

Replaced HFB masses by HFB24.

- July 14, 2014

Replaced microscopic  $\rho$ -densities by latest version from Stephane Hilaire.

- July 14, 2014

Added strength 6,7,8 models for gamma-ray strength functions. Added a new keyword Efile to read in user-defined gamma-ray strength functions.

- July 29, 2014

Added keyword Estop to stop calculation if incident energy gets above Estop (while not changing the file with incident energies which may go beyond Estop).

- August 6, 2014

Changed acceptable range for `aadjust`, `gnadjust` and `gpadjust` into 0.1 - 10. Also extended a few other ranges to accommodate random variations for TASMAn.

- August 8, 2014

Continue calculation if number of built rotational or class2 states exceeds `numrot` (currently 700). Instead of an error message we simply stop building more states.

- August 18, 2014

Solved a problem for medical isotope production found by Ismail Sarpun. The wrong lifetime is read from the decay data file for the ground state if a nuclide has more than 1 isomer. Solved this in `decaydata.f`.

- August 27, 2014

Extended the range of elements up to  $Z=124$ , with nuclear symbols `Rg(111)`, `Cn(112)`, `F1(114)` and `Lv(116)`, `B3(113)`, `B5(115)`, `B7-9(117-119)`, `C0-4(120-124)`

- August 28, 2014

Unified the output in separate files a bit more. Everything is now in `1p,e12.5` format. This makes processing by TEFAL and TASMAn easier.

- September 8, 2014

Added the stand-alone Fortran version of the GEF code as a subroutine to enable, in principle, the evaporation of fission fragments by TALYS. For this, TALYS "loops over itself", i.e. it is restarted internally for every excited fission fragment. The option for this is `fymodel 3`.

- October 8, 2014

Added extra security in `compound.f` to prevent division by zero in the case of fission. A rare NaN by TALYS was found for 160 MeV helions on Bi-203.

- October 22, 2014

Re-indexed the excitation energies in case of an incident population grid. This was corrected before but not written down.

---

Olivier Delaune pointed me to that error of TALYS-1.6.  
Created a new keyword popMeV. If popMeV y, the population grid is given as a spectrum per MeV (like before). If popMeV n (the default), the table represents pure histograms.

- November 25, 2014

Added alpha global OMP by Avrigeanu et al and Nolte et al.  
The range for alphaomp has now been extended to 8.

- December 21, 2014

Added keyword bestbranch to enable the adoption of adjusted branching ratios from the structure/levels/branch/ directory, for better description of isomeric cross sections and gamma-ray cross sections.

- April 13 2015

Corrected an error in the manual, the default GSF is Kopecky-Uhl for incident neutrons only, otherwise it is Brink-Axel. Found by David Walz.

- May 1 2015

Started implementing breakup model by Marilena Avrigeanu. There is now a keyword breakupmode which can have the values 1 (Kalbach) or 2 (Avrigeanu). The implementation is not yet complete.

- May 7 2015

On the advice of Stefan Buller, I added a keyword outbinspectra, which prints the emission spectra per compound nucleus bin. Obviously, the default is outbinspectra n, since it generates a lot of output.

- September 6 2015

Solved a problem for astrophysical reaction rates, found by Stylianos Nikas. The non-equidistant binning of excitation energies was not properly coded in partfunc.f leading to wrong astrophysical reaction rates, especially for high temperatures.

- September 24 2015

Defined deltaEx(0) to be half the energy of the first discrete level. It used to be zero.

- October 3 2015

Added keyword `ffspin` to enable a choice between spin dependent excitation functions for excited fission fragments as emerging from GEF, or assignment of spin distributions by TALYS

Added keyword `skipCN` to skip the decay of a compound nucleus altogether.

- October 10 2015

Generalized GSM for fission according to suggestions of Michael Onegin. Various level density parameters for the GSM now have an extra fission barrier index.

## E. Log file of changes since TALYS-1.8

Development of TALYS since release of TALYS-1.8

- December 28, 2015

\*\*\*\*\* Release of TALYS-1.8 \*\*\*\*\*

- January 21, 2016

Dimitri Rochman managed to crash TALYS, a rare feat indeed!  
For ldmodel 6, it was possible to have an undefined value for D0,  
leading to possible incident energies of 0 MeV, which crashes ECIS.  
Therefore a minimum value for enincrow, 1.e-11 MeV, was set in grid.f

- February 12, 2016

Solved an error found by Vlad Avrigeanu. Contrary to what is stated in  
the manual, alphaomp was NOT yet the default in TALYS-1.8.  
To use the best alpha OMP, you have to explicitly put alphaomp 6.  
This is solved now in input4.f

- June 1, 2016

Implemented the possibility to read external files for M1 strength  
functions, and also the possibility to scale them with the etable and  
ftable keywords.

Changes were suggested by Stephane Goriely. etable and ftable are now  
extended, e.g. etable 40 93 1.2 M1

A new keyword `mfile` was introduced.

- June 2, 2016

Replace `RECENT`, `SIGMA1` and `GROUPIE` subroutines of `PREPRO-2013` to `PREPRO-2015`.

- July 9, 2016

Revised Kalbach parameterization for break up reactions, according to the latest draft of her *Phys Rev C* paper.

- July 16, 2016

Added keyword `massdir` to add your own `masstables` to the `talys/structure/mass/` directory

- September 7, 2016

Corrected an error in the deformed OMP for actinides of Soukhovitskii et al, after a problem was reported by Fabio Zeisler. The Fermi energy in the OMP definition was taken from the global `KD03` formula, while the Soukhovitskii OMP uses specific Fermi energies per nucleus.

- September 19, 2016

Corrected an error found by Bernard Erasmus: medical isotope production Results were different for a second run in which the TALYS results were already precalculated. Solved this by initializing the discrete levels for non-binary channels (this was omitted when 'reaction n').

- November 2, 2016

Stephane Goriely added 2 new keywords `upbendc` and `upbende` to adjust gamma-ray strength functions at low energy.

- November 8, 2016

Stephane Goriely added 2 new keywords `astroex` and `nonthermlev` to account for cross sections to excited states.

To accommodate this I introduced variables `MaxZastro` and `MaxNastro` to keep the memory for `astro` arrays under control.

- November 16, 2016

Solved a problem for low lying  $0^+$  states. `ECIS` seems to produce



---

inelastic cross sections larger than the reaction cross section.  
In case of a weakcoupling model, if the first discrete level is a 0+ we replace it by 2+.

- November 28, 2016

Added a loop over astrophysical temperatures in resonance.f.  
Low energy resonance capture is now broadened by prepro at various temperatures to obtained physically more realistic reaction rates.

- December 8, 2016

Introduced the trim() option everywhere for characters so the length of pathnames no longer needs to be recorded, a nasty F77 feature.

- December 15, 2016

Added p-wave radiative width and D1, on the advice of Stephane Goriely.

- March 12 2017

Renamed the files of the entire structure/ directory. Instead of using e.g. z010.dat it is now called Ne.lev for the discrete levels (Ne for the element, .lev for the extension) and similarly in all other databases. The filenaming is now consistent all over the TALYS code and its satellite codes.

- April 3, 2017

Improved the output of isomeric fission yield ratios, on the advice of Satoshi Chiba.

- May 3, 2017

Repaired a bug found by Ann-Cecilie Larsen and solved by Stephan Goriely on equidistant energy grid in astrophysical calculations in partfunc.f

- May 5, 2017

Must repair bug found by Dmitry Smychko: adopt ENDSF suggested spin in all cases.

Must repair bug found by Steven Gardiner: K39 has even spin levels, maybe more general problem

- September 22, 2017

Separated quasi-deuteron calculation in separate subroutine

- November 11, 2017

Replaced PREPRO subroutines by 2017 versions.

## F. Log file of changes since TALYS-1.9

Development of TALYS since release of TALYS-1.9

- December 21, 2017

\*\*\*\*\* Release of TALYS-1.9 \*\*\*\*\*

- January 30, 2018

Imported new PSF D1M + QRPA M1 data tables in structure directory. Adopted two keywords, 'upbend' to enable or disable the low-energy upbend formula of the PSF (default n) by Stephane Goriely. Also introduced the 'upbendf' parameter for this formula.

The upbendc, upbende and upbendf parameters now have a default, which are likely to change in the near future.

- February 2, 2018

Removed a few redundant quantities from massdis.f and ffevap.f, like pre-neutron nubar (which is nonsense)

- February 13, 2018

Corrected an error found by Stephane Goriely. Due to a trim() error in densitytable.f tables for elements with only one symbol, like U, are not read for ldmodel 4, 5 and 6.

Corrected that. I expect more comments about this one!

- March 14, 2018

Solved a few errors in medical isotope production, found by Jiri Ulrich. The Tcool keyword was not working properly. Solved that in input5.f Also some suggestions on time steps and the use of natural targets were adopted.

- April 13, 2018

Error found by Dimitri Rochman, the combination expmass n, massmodel 0 did not work. Solved in masses.f

- July 6, 2018

Expanded the output for decay from each continuum and discrete bin. If outpopulation y, which is automatically enabled when outbasic y, then an extra keyword can be given 'outdecay y', which will output the precise Ex,J,pi dependent decay from each mother state in the CN decay. This was added on the advice of Stephane Goriely to better study reactions used by the Oslo group for the determination of PSF's.

- March 12, 2019

Added SML0 PSF tables from Stephane Goriely and Vladimir Plujko. Also changed some of the default parameter for upbend formula for PSF. strength 9 now added for SML0.

- March 15, 2019

Added the fission width average fission lifetime, and fission level density to the specific output of the fission transmission coefficients.

- April 7, 2019

Corrected an error found by Stephane Goriely on the calculation of the average emission energy.

The discrete levels were not properly taken into account.

This quantity was only printed in the basic output of TALYS (and not used for anything). Now, however, when 'outspectra y', files like Eaverage.n will be created containing the average emission energy per particle.

- April 9, 2019

With 'filedensity y' TALYS now also prints the level density tables exactly in the form as they are used as input from the nuclear structure

---

data tables, these files are called e.g. nld041093.tab.

- May 28, 2019

Solved an error in levels.f: if the isomer of a residual nucleus is above the maximum level included it is brought down to that maximum level. So far this was only implemented for the target nucleus, i.e. inelastic scattering.

- June 9 2019

Added normalization of average number of neutrons to massdis.f

- June 23 2019

Solved a bug discovered by Rob Hoffman. In input5, for the gp parameter (single particle state density for protons) gn was read instead of gp. Since for fitting we often use gpadjust this bug got unnoticed for years!

- June 23 2019

Added a new keyword 'psfglobal' to use only global photon strength functions and parameters. By default 'psfglobal n' meaning that the best GDR parameters, tuned PSF tables etc are used. This keyword was especially introduced for two tables for the SML0 model, one optimized and one global, but is applicable in general.

- August 13 2019

Included fission partial damping model constructed by Guillaume Scamps and Stephane Goriely. This should bring TALYS closer to the capabilities of EMPIRE for fission prediction. For this, the keywords, fispartdamp, bdamp and bdampadjust were added.

- December 16 2019

Included parameters wtable and wtableadjust to change the width of a tabulated PSF.



## G. Log file of changes since TALYS-1.95

Development of TALYS since release of TALYS-1.95

- December 30, 2019

\*\*\*\*\* Release of TALYS-1.95 \*\*\*\*\*

- January 4, 2020

Added keyword 'soukho' to allow disabling of Soukhovitskii OMP for actinides. Now parameters relative to KD03 can be given.

- January 16, 2020

Implemented Okumura model for fission yields and subsequent particle evaporation for fission neutrons. Keyword fymodel has been extended with option '4' for this model. Also added fymodel 5 which allows to enter FF population tables.

- February 5, 2020

Introduced keyword 'yieldfile' to allow input of fission fragment yields per pair, and their respective TXE and TKE values. This serves as input for the Okumura model to evaporate each excited fission fragment individually.

- March 2, 2020

Improved the calculation for a populated nucleus, needed after tests done with Toshihiko Kawano on the depletion of fission products. The flux of the populated nucleus is now better mapped onto the decay scheme. Subroutine `excitation.f` was changed for that. The number of energy bins can now also be equal to 1, for one starting excited level.

- March 23, 2020

Changed the default to 'equidistant y' after binning problems observed by Sandor Sudar at high energy proton reactions.

- March 27, 2020

Added keyword `gadjust` as 1-component version of `gnadjust` and `gpadjust`. This can be used for consistent variation of `gn` and `gp` for the 2-component exciton model.

- May 24, 2020

Removed a bug in the reading of an external, tabulated optical model file. The first line of that file should contain Z, A, number of lines and (optional) Fermi energy. This did not hurt the sample case for `n + U238` which is why the error got unnoticed for so long.

- May 27, 2020

TALYS can now read RIPL OMP, which is for sure necessary for actinides and probably helpful for other cases as well. The initial version has been tested for important actinides.

- May 31, 2020

Extended isospin forbidden transitions for multiple emission using 'fisom' keyword on advice of Stephane Goriely.

- Aug 17, 2020

Reprogrammed the isospin transition cases by making a new subroutine `isotrans.f`.

- Sep 27, 2020

Improved the output of level densities after suggestions by Khashane Malatji: the total parity-dependent level densities were not properly tabulated. Made a new subroutine `densitytotP` for this, which only affects the level density output, not the reaction calculations.



---

- Oct 19, 2020

On the advice of Jean-Francois Lemaire, revised various subroutines for the FY and PFNS calculations, There was an error in the CM to LAB integration boundaries and the FY and spin distribution normalizations.

- Dec 22, 2020

Added a 'block' keyword to write large amount of output data in output blocks instead of individual files. With 'block n' the old situation is retained: All spectra, angular distributions etc are stored in an individual file per incident energy. We use 'block y' now to create ENDF files, as it does not produce a huge amount of files.

- Feb 1, 2021

Corrected more coding on FY, PFNS and nubar etc. in massdis.f and ffevap.f on the advice of Jean-Francois Lemaire.

- Feb 15, 2021

Include Marilena Avrigeanu's deuteron break-up model. For this, the structure/ database was extended with the TENDL neutron and protons cross sections.

- March 1, 2021

Corrected pre-neutron FY table on advice of Ali Al-Adili. Also added average gamma and neutron energy per FF in the output.

- April 26, 2021

Changed the default for subactinide fission. Now fismodel 3 is used for charged-particle induced reactions on all targets and for neutrons on subactinides, which gives a dramatic improvement of the fission cross sections.

- April 28, 2021

Added the option to read in fission fragment distributions from the structure database, after the provision of FF distributions from Ali Al-Adili and Fredrik Nordstroem.  
Added the keyword 'ffmodel' to chose between models of FF distribution.

- June 6, 2021

Updated the resonance parameter files in talys/structure/resfiles. Now resonance parameters can be read for tendl.2019 (default), endfb8.0, jeff3.3, cendl3.2 and jendl4.0.

- July 13, 2021

Changed default parameters for break-up reactions for (d,p) and stripping reactions for (alpha,n) to improve the global quality of the cross section fits. This extra adjustment is done at the top of breakup.f.

- August 23, 2021

Added a subroutine for Wahl systematics for nubar, which is now written to output by TALYS also without an explicit FF to FY calculation. Separated some of the calculation of nubar for total and nu(A) for that in nudisout.f and nubarout.f. Made keywords Cnubar1 and Cnubar2 to alter systematic value for nubar.

- September 1, 2021

Enabled the reading in of so called wtable parameters to change the width of tabulated photon-strength functions.

- September 6, 2021

Changed the default in TALYS for photon strength functions after testing with Stephane Goriely. The default is now strength 9 (SMLO), strengthm1 3 with 'upbend y'. This is applied for all incident particles, improving the default prediction for (n,g), (g,n), (p,g), (a,g) considerably. Automatic wtable parameters are read in for adjustment to capture cross sections in the fast neutron range.

- September 9, 2021

Added new keyword 'ldmodelCN' to assign a particular level density model to the compound nucleus only. So e.g. ldmodelCN 1 and ldmodel 2, simultaneously in the input file will use CTM for the CN and BFM for all other nuclides in the calculation. The same can be achieved by e.g. ldmodel 1 26 57 and ldmodel 2, but the new option avoids having to specify the Z,A of the compound nucleus.

- September 28, 2021

Added keywords 'ngfit', 'nnfit', 'nafit', by default disabled,

to use fitted nuclear model parameters. If 'best y' then 'ngfit y' by default.

- November 4, 2021

Added Iwamoto model for PFNS, which is a variant of the Los Alamos model. Added the keyword 'pfnsmodel' to enable it instead of PFNS coming from explicit FF evaporation. Also added the keywords 'Tadjust' and 'Fsadjust' to adjust the temperature and number of scission neutrons in this model.

- December 29, 2021

Added keywords vfiscoradjust and betafiscoradjust to enable changes relative to vfiscor and betafiscor.



## H. Derivation of isotope production equations

Here we derive the solutions (19.10) of the differential equations (19.6)-(19.9). Eq. (19.10) has been implemented in ISOTOPIA.

For the target nuclide we have

$$\frac{dN_T(t)}{dt} = -R_T N_T(t). \quad (\text{H.1})$$

This is easily integrated to give

$$N_T(t) = N_T(0) e^{-R_T t}. \quad (\text{H.2})$$

For every parent nuclide we have

$$\frac{dN_p(t)}{dt} = R_{T \rightarrow p} N_T(t) - \lambda_p N_p(t). \quad (\text{H.3})$$

Multiplying both sides with  $e^{\lambda_p t}$  gives

$$e^{\lambda_p t} \frac{dN_p(t)}{dt} = e^{\lambda_p t} [R_{T \rightarrow p} N_T(t) - \lambda_p N_p(t)], \quad (\text{H.4})$$

or,

$$e^{\lambda_p t} \frac{dN_p(t)}{dt} + e^{\lambda_p t} \lambda_p N_p(t) = e^{\lambda_p t} R_{T \rightarrow p} N_T(t), \quad (\text{H.5})$$

or,

$$\frac{d[e^{\lambda_p t} N_p(t)]}{dt} = e^{\lambda_p t} R_{T \rightarrow p} N_T(t). \quad (\text{H.6})$$

Integrating this, after insertion of Eq. (H.2), gives

$$\begin{aligned} e^{\lambda_p t} N_p(t) &= N_T(0) R_{T \rightarrow p} \int dt e^{(\lambda_p - R_T)t} \\ &= \frac{N_T(0) R_{T \rightarrow p}}{\lambda_p - R_T} e^{(\lambda_p - R_T)t} + C, \end{aligned} \quad (\text{H.7})$$

or,

$$N_p(t) = \frac{N_T(0) R_{T \rightarrow p}}{\lambda_p - R_T} e^{-R_T t} + C e^{-\lambda_p t}. \quad (\text{H.8})$$

Inserting the initial condition  $N_p(t = 0) = 0$  gives

$$0 = \frac{N_T(0) R_{T \rightarrow p}}{\lambda_p - R_T} + C, \quad (\text{H.9})$$

giving the solution for the parent nuclide,

$$N_p(t) = \frac{N_T(0) R_{T \rightarrow p}}{\lambda_p - R_T} \left[ e^{-R_T t} - e^{-\lambda_p t} \right]. \quad (\text{H.10})$$

For every isotope  $i$ , directly produced by a nuclear reaction on the target isotope or from decay of the parent nuclide, we have

$$\frac{dN_i(t)}{dt} = \lambda_{p \rightarrow i} N_p(t) + R_{T \rightarrow i} N_T(t) - \lambda_i N_i(t). \quad (\text{H.11})$$

Multiplying both sides with  $e^{\lambda_i t}$  gives analogous to Eqs. (H.4)-(H.6),

$$\frac{d[e^{\lambda_i t} N_i(t)]}{dt} = e^{\lambda_i t} [\lambda_{p \rightarrow i} N_p(t) + R_{T \rightarrow i} N_T(t)], \quad (\text{H.12})$$

or, after insertion of the solutions for  $N_T$ , Eq. (H.2), and  $N_p$ , Eq. (H.10),

$$\frac{d[e^{\lambda_i t} N_i(t)]}{dt} = e^{\lambda_i t} \left[ \lambda_{p \rightarrow i} \frac{N_T(0) R_{T \rightarrow p}}{\lambda_p - R_T} (e^{-R_T t} - e^{-\lambda_p t}) + N_T(0) R_{T \rightarrow i} e^{-R_T t} \right]. \quad (\text{H.13})$$

Integrating this gives

$$\begin{aligned} e^{\lambda_i t} N_i(t) &= \frac{N_T(0) \lambda_{p \rightarrow i} R_{T \rightarrow p}}{\lambda_p - R_T} \left[ \int dt e^{(\lambda_i - R_T)t} + \int dt e^{(\lambda_i - \lambda_p)t} \right] \\ &+ N_T(0) R_{T \rightarrow i} \int dt e^{(\lambda_i - R_T)t} + C, \end{aligned} \quad (\text{H.14})$$

or,

$$\begin{aligned} N_i(t) &= \frac{N_T(0) \lambda_{p \rightarrow i} R_{T \rightarrow p}}{\lambda_p - R_T} \left[ \frac{e^{-R_T t}}{\lambda_i - R_T} + \frac{e^{-\lambda_p t}}{\lambda_i - \lambda_p} \right] \\ &+ \frac{N_T(0) R_{T \rightarrow i}}{\lambda_i - R_T} e^{-R_T t} + C e^{-\lambda_i t}. \end{aligned} \quad (\text{H.15})$$

Inserting the initial condition  $N_i(t = 0) = 0$  gives

$$0 = \frac{N_T(0) \lambda_{p \rightarrow i} R_{T \rightarrow p}}{\lambda_p - R_T} \left[ \frac{1}{\lambda_p - R_T} + \frac{1}{\lambda_i - \lambda_p} \right] + \frac{N_T(0) R_{T \rightarrow i}}{\lambda_i - R_T} + C, \quad (\text{H.16})$$

so we finally obtain

$$N_i(t) = N_T(0) \frac{R_{T \rightarrow i}}{\lambda_i - R_T} [e^{-R_T t} - e^{-\lambda_i t}] + N_T(0) \frac{\lambda_{p \rightarrow i} R_{T \rightarrow p}}{\lambda_p - R_T} \left[ \frac{e^{-R_T t} - e^{-\lambda_i t}}{\lambda_i - R_T} - \frac{e^{-\lambda_p t} - e^{-\lambda_i t}}{\lambda_i - \lambda_p} \right]. \quad (\text{H.17})$$

The three solutions of interest are thus given by Eqs. (H.2), (H.10), and (H.17).

# I. GNU GENERAL PUBLIC LICENSE

Version 3, 29 June 2007

## I.1 Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program—to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be

marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

## I.2 Terms and Conditions

### 0. Definitions.

“This License” refers to version 3 of the GNU General Public License.

“Copyright” also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

“The Program” refers to any copyrightable work licensed under this License. Each licensee is addressed as “you”. “Licensees” and “recipients” may be individuals or organizations.

To “modify” a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a “modified version” of the earlier work or a work “based on” the earlier work.

A “covered work” means either the unmodified Program or a work based on the Program.

To “propagate” a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To “convey” a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays “Appropriate Legal Notices” to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

### 1. Source Code.

The “source code” for a work means the preferred form of the work for making modifications to it. “Object code” means any non-source form of a work.

A “Standard Interface” means an interface that either is an official standard defined by a



recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The “System Libraries” of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A “Major Component”, in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The “Corresponding Source” for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work’s System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

## 2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

## 3. Protecting Users’ Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures. When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit

operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

#### 4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

#### 5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

- (a) The work must carry prominent notices stating that you modified it, and giving a relevant date.
- (b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".
- (c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.
- (d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

#### 6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

- (a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.
- (b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable

- physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.
- (c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.
  - (d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.
  - (e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A “User Product” is either (1) a “consumer product”, which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, “normally used” refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

“Installation Information” for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM). The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed.

Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

#### 7. Additional Terms.

“Additional permissions” are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

- (a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or
- (b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or
- (c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or
- (d) Limiting the use for publicity purposes of names of licensors or authors of the material; or
- (e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or
- (f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered “further restrictions” within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating

where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An “entity transaction” is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party’s predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A “contributor” is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor’s

“contributor version”.

A contributor’s “essential patent claims” are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, “control” includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor’s essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a “patent license” is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To “grant” such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. “Knowingly relying” means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient’s use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is “discriminatory” if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

#### 12. No Surrender of Others’ Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that

contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License “or any later version” applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy’s public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO

USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS