TASMAN-2.0

Statistical software for TALYS: Uncertainties, sensitivities and optimization

Arjan Koning

s

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/Bruyères-le-Châtel and Los Alamos National Laboratory on the development and computational implementation of nuclear reaction models. He has led several 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 for 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 2023, his h-index is 57 with more than 21 000 citations.

Although Arjan is currently in a managerial role, he aims to keep his scientific creativity alive by maintaining and extending TALYS plus all products that emerge from that. Pleas from his friends to also spend time on other things are sometimes honoured.

Preface

TASMAN is a software package for statistical operations on the nuclear model code TALYS and other related software. TASMAN generates random distributions of input parameters for the TALYS code, then runs TALYS for each random input file, and collects the calculated results to obtain probability distributions of basically all output quantities of TALYS. From this, statistical information such as covariance matrices for parameters and cross sections can be constructed. Also sensitivity profiles can be obtained. With the same technique, automatic optimization ("parameter search") of the TALYS results to experimental data is possible, so the best possible fit to experiment can be obtained. For all this, the entire EXFOR database, and several ENDF data libraries for that matter, are linked to TASMAN. The idea to make TASMAN was born in 2004, when I was using the TALYS and TEFAL codes for the production of nuclear data libraries. What was missing was a method to assign uncertainties to our model-calculated results, and the associated absence of covariance data in the data libraries. Also, manual fitting of TALYS results to experimental data is sometimes tedious, calling for a more automated optimization method. For both purposes, input generation, reading and subsequent processing of many TALYS output files is required, and this is implemented in TASMAN.

At certain moments in time, a well-defined version of TASMAN is frozen and subjected to extensive verification and validation procedures. You are now reading the manual of version 2.0.

License, contact and reference

As mentioned on the first page and in the source code, TASMAN falls in the category of MIT License software.

In addition to the MIT *terms* I have a *request*:

- When TASMAN is used for your reports, publications, etc., please make a proper reference to the code. At the moment this is:
  A.J. Koning, D. Rochman, J.-Ch. Sublet, N. Dzysiuk, M. Fleming, and S. van der Marck,

TENDL: Complete Nuclear Data Library for innovative Nuclear Science and Technology, Nuclear Data Sheets 155,1 (2019).

The webpage for TASMAN is **nds.iaea.org/talys**.

Acknowledgements

Arjan Koning

Contents

1. Introduction

TASMAN is a computer code system for nuclear data uncertainty distributions based on the nuclear model code TALYS [1], and for automatic optimization of the TALYS results with respect to experimental data.

The assessment of uncertainties in nuclear reaction data is important for both basic physics and technological applications. First, it defines the current quality of both experimental and theoretical nuclear reaction physics. Experimental uncertainties represent a sort of limit of the current-day precision with which we can measure a particular observable, and in general these uncertainties have a systematical and a statistical component. Theoretical uncertainties would represent the extent to which we are able to cast nature into models and their associated parameters. Second, nuclear reaction data, such as cross sections, resonance parameters, energy spectra and angular distributions are of prime importance to the computational simulation of nuclear installations. A reliable assessment of the uncertainties in calculated integral reactor parameters depends directly on the uncertainties of the underlying nuclear data. To make the most impact for design calculations, nuclear data libraries should include uncertainty information. Starting from the covariance matrices of the basic nuclear data, error propagation in transport, reactor, activation, etc. codes enables to estimate the uncertainties of calculated design parameters, which has a profound impact on issues of general concern such as safety and economy.

An often heard complaint is that covariance nuclear data are scarce, and also that the covariance evaluation process can be rather complex and time-consuming. While the good habit of delivering error bars is followed by almost all experimental nuclear physicists, theoreticians do not always (well, more often than not) provide uncertainties, let alone a full covariance matrix, with their results. With the help of the TALYS nuclear model code and the current-day computer power, it is now possible to do something about this. We note that the basic ideas of the approach have first been outlined by Smith[2] and later by Bauge[3], Capote [4] and Koning [5], and we have implemented these and related methods in TASMAN to show that the method is actually feasible. The essential idea is to assume that each nuclear model parameter has its own uncertainty. Running TALYS

many times, whereby each time all elements of the input parameter vector are *randomly* sampled from a normal, uniform or general (numerical) distribution with a specific width for each parameter distribution, provides a lot of statistical information, such as exact uncertainty distributions, and also variances and a full covariance matrix. The basic objective behind the construction of TASMAN is to facilitate all this.

Moreover, TASMAN can also be used for automatic fitting of calculated results to experimental data. We use several optimization methods, Monte Carlo and deterministic, to find the optimal nuclear model parameters, i.e. the parameters that result in the best fit of experimental data for a particular reaction channel, for all reaction channels of a nucleus, or even for an entire nuclide range.

In sum, the following is possible with the TASMAN-TALYS combination:
- Production of uncertainty distributions and covariance matrices for cross sections, spectra, angular distributions etc.
- Production of uncertainty distributions and covariance matrices for nuclear model parameters.
- Production of sensitivity matrices based on perturbation or Monte Carlo, i.e. matrices that shed light on the dependence of all quantities that are calculated with TALYS on nuclear model parameters.
- Automatic optimization of the TALYS model parameters to obtain the optimal fit of experimental data, either for one reaction and/or nuclide or many nuclides simultaneously.
- Combination of optimization and probability distributions using Bayesian Monte Carlo.

As specific features of TASMAN we mention
- Production of TALYS input files with random input parameters, and then run TALYS, to enable covariance generation with the Bayesian Monte Carlo covariance method.
- Possibility to handle basically all of the output files of TALYS (cross sections, angular distributions, spectra, etc.) in an optimization scheme or for covariance production.
- Robustness, enabling the TASMAN-TALYS combination to be used in scripts that produce a huge amount of results.
- Production of ENDF-6 covariance data libraries by using the TASMAN-TALYS-TEFAL combination. TEFAL[6] is a code for the production of nuclear data libraries in ENDF-6 format. This is the backbone for the TENDL library.
- Setting TASMAN-TALYS-TEFAL in "Total Monte Carlo" mode[7], in which no covariance data are produced, but instead a statistical ensemble of random ENDF-6 data libraries for the same isotope, for exact uncertainty propagation. For a complete neutron data library, also the codes TARES, TAFIS and TANES are required.
- A transparent source program.
- Input/output communication that is easy to use and understand.
- A user manual.
- A few sample cases.

In the current implementation, TASMAN thus runs TALYS, and if required also other software, via system calls.

1.1   How to use this tutorial

For users that are interested in nuclear model based covariances, probably most of this manual is relevant, but you may check below whether parts can be skipped. After this Introduction, you will find

Chapter 2: Installation guide.
Chapter 3: The general rules for the input file.
Chapter 4: The software that is driven by TASMAN

Chapter 5: A description of the basic formulae for uncertainty generation and a discussion on the expected reliability of the method.
Chapter 6: Optimization to experimental data
Chapter 7: Reference Guide. A description of all keywords
Chapter 8: Outlook and conclusions

2. Installation and getting started

2.1 The TASMAN package

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

- *LICENSE* is the license file,
- *README.md* outlines the contents of the package,
- *code_build* and *path_change* are scripts that take care of the installation,
- *source/* contains the Fortran source code of TASMAN,
- *bin/* contains the TASMAN binary after successful compilation,
- *misc/* contains miscellaneous files used by TASMAN or related scripts,
- *parameters/* contains numerical probability distributions of parameters
- *doc/* contains the documentation: this manual in pdf format.
- *samples/* contains the input and output files of the sample cases.

In total, you will need only a few tens of Mb free disk space to install TASMAN. The code has so far *only* been tested on various Linux and MacOS systems, so we can not guarantee that it works on all operating systems. The only OS 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.f90*.

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

2.2 Installation

The installation of TASMAN is straightforward. You can download TASMAN via either git

- **git clone https://github.com/arjankoning1/tasman.git**

or by getting the tar file

- from **https://nds.iaea.org/talys/tasman.tar**

    - **tar zxf tasman.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 *tasman.setup* script, as follows

- edit *code_build* and set the first two variables: the name of your compiler and the place where you want to store the TASMAN executable.
- **code_build tasman**

An alternative installation option is

    - **cd tasman/source**
    - **make**

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

- **cd tasman/source**
- Ensure that TASMAN can read the binary directory with the TALYS and/or other executables (e.g. TEFAL for the production of ENDF-6 files). This is done in subroutine *machine.f90*. If *code_build* has not already replaced the path name in *machine.f90*, do it yourself. We think this is the only Unix/Linux machine dependence of TASMAN. We expect no complaints from the compiler.
- **gfortran -c *.f90 *.f**
- **gfortran *.o -o tasman**
- **mv tasman ../bin**

The above commands represent the standard compilation options. Consult the manual of your compiler to get an enhanced performance with optimization flags enabled. We note that the part of a total run that is taken by TASMAN generally takes only takes a few seconds so extreme optimization is not necessary. It is the underlying TALYS run that may be time consuming.

2.3   Verification and sample cases

If TASMAN 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. It also contains a subdirectory *new*, where we have stored the input files only and where the *verify* script will produce **your** output files. Of course, it is assumed that you have TALYS installed, see the TALYS manual, to do the underlying nuclear model calculations. If you want to do "Total Monte Carlo" (TMC), you will also need TEFAL and possibly other auxiliary codes (TARES, TAFIS and TANES). A full description of the keywords used in the input files is given in Chapter 3. The sample cases are scattered throughout this tutorial. Note that in each subdirectory a file with differences with our original output is created.

2.4   Getting started

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

tasman < **tasman.inp** > **tasman.out**

where the name *tasman.inp* is not obligatory: you can use any name for this file. Besides the general output file *tasman.out* various other output files are produced.

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 TASMAN and its users, we have constructed an input/output method which shields beginners from all the possible options for parameters that can be specified in TASMAN, while enabling at the same time maximal flexibility for experienced users.

The input handling of TASMAN is very similar to that of TALYS, in the sense that an input file of TASMAN also consists of keywords and their associated values. Even more, a correct input file of TASMAN can also be used as an input file for TALYS. There is however one essential difference. As the experienced TALYS user may know, TALYS input lines can be disregarded by letting them start with the '#' symbol, which is helpful for temporary disabling keywords or to insert comments. TASMAN is different in that it *reacts* to keywords that start with a '#'. As we will explain below, the keyword **#change** plays a central role in this.

As usual, you have the choice to give only a few keywords or many of them. Behind a simple input file there are default values for more than 100 parameters, options for level of detail, etc., that you may specify. When you use a minimal input file, you leave it to the author of TASMAN to choose all the parameters for you. If you want to use specific parameters or want to have more, or less, specific information in the data file, more keywords are required, such as in this example:

```
projectile n
element nb
mass 93
energy energies
rvadjust n 1.03
#change
#mode 1
m2constant 1.3 0.20
a 41 94 8.735
```

In this Chapter, we will first give the basic rules that must be obeyed when constructing an input file for TASMAN. Later in this tutorial, we give an outline of all the keywords, which have been categorised in several groups. Finally, we summarize all keywords in one table.

3.1 Basic input rules

First of all, there is one magical keyword for TASMAN that determines the entire execution of the code. As stated above, the TASMAN input syntax is basically equal to that of TALYS. However, as soon as you give the line **#change**, see the sample file above, TASMAN assumes that everything coming *below* that line is to be taken into account for uncertainty and/or optimization purposes, including lines that start with a single '#'. For example, in the input file above, **#mode 1** puts TASMAN in the mode of Monte Carlo uncertainty quantification, while the TALYS keywords **m2constant** and **a** will be randomly varied around their central values. The value of of the TALYS keyword **rvadjust** will *not* be varied since it is placed above **#change**. Similar to TALYS, successful execution of TASMAN 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, but some keywords need to be accompanied by more than one value 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 you use the same keyword more than once, the value of the last one will be adopted.
4. All characters can be given in either lowercase or uppercase.
5. A keyword *must* be accompanied by a value. To use default values, the keywords should simply be left out of the input file. The exception is **#change** which does not have a value.
6. An input line starting with a # in column 1 may be recognized as a TASMAN keyword. If it occurs *before* the input line with **#change**, it is treated as a comment. If it occurs after this line, it is treated as a TASMAN keyword, and it will be tested whether the keyword exists. If you want to give comments after **#change**, they should start with **##** on the input line, after which these lines will be neglected by TASMAN.
7. An input line may not exceed 132 characters.

If the above rules are not obeyed, TASMAN will give an error message for the first encountered problem and the execution will be stopped.

3.2 Parameter variation

TASMAN adds an important feature to a seemingly standard TALYS input file. If TALYS keywords are specified *after* the line with **#change**, we can add the allowed variation (uncertainty) to that parameter on the same line. This is done for the **m2constant** parameter in the input file above. The central value is 1.3 and we let TASMAN randomly sample values in a range of 20% around the central value. The added value thus concerns the *relative deviation*, given as a fraction of unity. Whether this deviation concerns the width of a Gaussian (the default) or a uniform sampling distribution depends on the user. We note that the other TALYS keyword in this input file, **a**, is also randomly varied, but here we use its default relative deviation, and we do not have to add a deviation to the central value. All TALYS parameters have a built-in relative deviation and Section 5.4 contains a list of all default variations. What TASMAN does in practice is to generate many random TALYS input files in which

• All input lines above **#change** are copied and left untouched,

- All TASMAN keywords starting with a '#' are, as usual, treated as comments by TALYS
- All TALYS keywords below **#change** take on random values inside the uncertainty range specified in the input or taken as default. These values are written to the many "random" TALYS input files.

Next, TASMAN gives a system call to run TALYS for each random input file and collects all the results for processing into statistical output. For example, the first random TALYS input file, generated from the TASMAN input file above, may look as follows

```
projectile n
element nb
mass 93
energy energies
rvadjust n 1.03
#change
#mode 1
m2constant 1.3426
a 41 94 8.602
```

and also in all the other random TALYS input files only the last two lines change. If required, hundreds or even thousands of such parameters can be varied in one and the same input file.

As explained in sample cases in Chapter 5, there is a more efficient way to vary TALYS parameters by a combination of the **#parameter**, **#partvary**, **#ZAvary**, **#keyvary**, **#Zdeep**, **#Adeep** keywords For example, with the following input file

```
projectile n
element nb
mass 93
energy energies
#change
#mode 1
#Zdeep 1
#Adeep 2
#parameters y
#partvary p
#partvary n
#keyvary etable
#keyvary ftable
#keyvary rvadjust
#keyvary avadjust
```

all TALYS parameters given with the **#keyvary** keyword will be varied up to a depth of $Z = 1, A = 2$ units away from the initial compound nucleus and/or for neutrons and protons. More combinations and examples are given in Chapter 5.

4. Software called by TASMAN

Obtaining statistical information by means of Monte Carlo sampling is basically quite simple: make sure you have an input generator for the code that produces the physical results of interest, run that code repeatedly with random input files, and collect the results to obtain averages, standard deviations, covariances etc. Depending on the interest of the user, TASMAN calls one, TALYS, or more codes. A short description of these codes will be given in this chapter.

4.1 The TALYS nuclear model code

The nuclear model code TALYS[1] is designed to analyse and predict nuclear reactions. The physical models implemented in TALYS have been validated for the simulation of nuclear reactions that involve neutrons, photons, protons, deuterons, tritons, ^{3}He- and alpha-particles, in the keV - 200 MeV energy range and for target nuclides of mass 19 and heavier. Formally, the ranges are even more extensive than this: Using external resonance information the code now produces meaningful results already in the meV range, and the code can even be used for targets as light as ^{6}Li and up to incident energies of 1 GeV. Note however that the optical and statistical models implemented in TALYS are known to break down for nuclides with masses below about 20. Also, above several hundreds of MeV the omission of pion production in TALYS will become obvious. Other limitations are the absence of a sound model for gamma-ray transitions at thermal energies, and explicit treatment of break-up reactions such as (d,p) to discrete states, and the latest models for quantum-mechanical pre-equilibrium emission. The code aims at quality and completeness at the same time: Sophisticated nuclear reaction models have been implemented, using a vast range of microscopical nuclear structure ingredients, but at the same time allows to add enough phenomenology to always produce desired answers.

TALYS has been extensively validated over the past decades in fields ranging from fundamental nuclear reaction studies to data evaluation for nuclear reactors, medical isotope production and

astrophysics, among others. This validation was not only performed by the authors but also by thousands of users who often do not hesitate to report both positive and negative feedback, including bugs, to the authors which has made this open source code very robust. Currently on average about 2 papers per day are published that use TALYS in some way, and that the total number of papers citing TALYS is now beyond 6000. An extensive description of the various applications of TALYS can be found in [1]. Fig. 4.2 gives a breakdown per application.
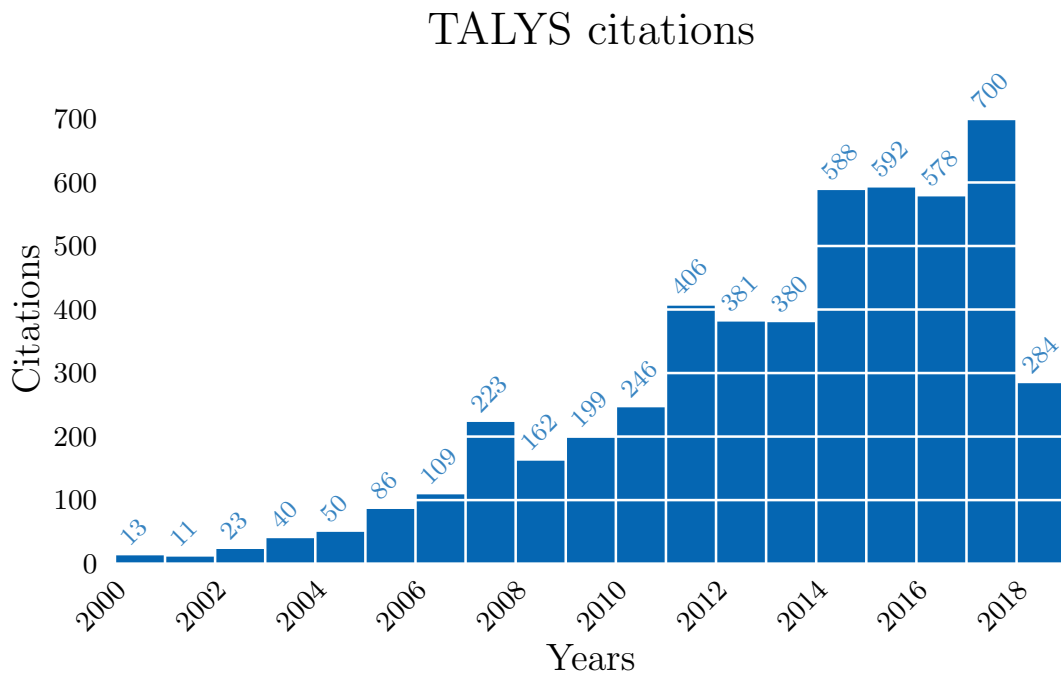
Figure 4.1: Time-dependence of TALYS citations, from the first beta release in 2001.

Since all models used for the nuclear reaction mechanisms implemented in TALYS are limited, it can not be expected that TALYS can ab initio reproduce high-quality experimental data, and therefore adjustable model parameters are needed. More than 400 different keywords are available, though not all for adjustable model parameters but also for model and output options etc., to let TALYS do what the user desires. And even with that, reproduction of experimental data may not always be successful, if the nuclear model is "too wrong". This holds especially for actinides and the modeling of the fission channel, for which it is still difficult to use TALYS as a routine production tool. An advantage of TALYS is that it can generate data for any reaction channel regardless of whether it has been measured or not. The quality of the final nuclear reaction simulation is then represented by the ability of TALYS to match, interpolate between, and extrapolate beyond the existing experimental data.

TALYS can be used for many purposes, not only nuclear data evaluation. As output, for a single projectile + target combination and a range of incident energies, TALYS produces cross sections (assuming incident neutrons here) for:

- total (n,tot), elastic (n,el) and non-elastic (n,non) reactions,
- capture channel (n,γ),
- single particle production channels (n,n'), (n,p), (n,d), (n,t), (n,h), (n,α),
- discrete level inelastic reactions, (n,n'$_1$), (n,n'$_2$), etc. and the continuum (n,n'$_{\text{cont}}$), and similarly
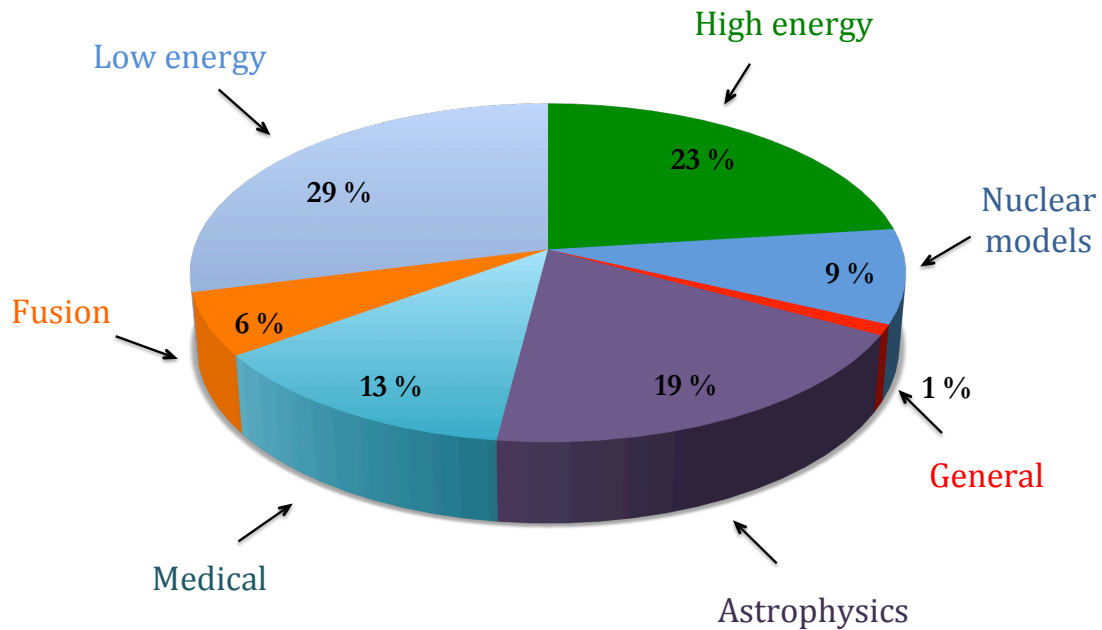
Figure 4.2: Segmentation of TALYS publications by application area, over the years 2001-2021.

for all other ejectiles,

- multi-particle production (n,2n), (n,np), etc,
- total fission (n,f) and its subdivision into first, second, etc. chance partial fission,
- residual production $(n,x)^Z_A$El, mostly relevant at higher energies, as far as not yet covered by the exclusive channels mentioned above,
- production of the ground state and isomers, if present, e.g. $(n,n')^g$, $(n,n')^m$, and similarly for all other channels and residual products,
- total particle production (n,xn), (n,xp), etc, mostly relevant at higher energies.

As for secondary distributions, TALYS produces:

- the elastic scattering angular distribution,
- the angular distributions for inelastic scattering per discrete level, and similarly for the other ejectiles,
- double-differential emission spectra for all outgoing particles,
- recoil distributions for the residual nuclides,
- particle production yields,
- discrete and continuum gamma-ray distributions for all open channels.

and other nuclear reaction information:

- Level density tables,
- Maxwellian-averaged cross sections,
- Astrophysical reaction rates,
- Integral (effective) cross sections (averaged over a spectrum),

- Fission yields.

The only output that TALYS does not (yet) provide is resonance information (provided by TARES), average number of neutrons from fission (provided by TAFIS), prompt fission neutron spectrum (provided by TANES), and output files in the ENDF format (provided by TEFAL). These codes are discussed in the next sections.

What is important for TASMAN is (1) all nuclear model parameters (TALYS keywords) have uncertainties and can be varied, and (2) all output quantities of TALYS (see the list above) will therefore vary as well.

4.2   The TEFAL code for ENDF-6 formatting

The ENDF-6 formatting code TEFAL processes the nuclear reaction results of TALYS, and data from other sources if TALYS is not used, into ENDF-6 nuclear data libraries. TEFAL was constructed to avoid any error-prone human interference in the creation of nuclear data files: the whole ENDF-6 file is created at once, on the basis of all nuclear reaction info that we feed it. Hence, the idea is to first run TALYS for a projectile-target combination and a range of incident energies, and to obtain a ready to use nuclear data library from the TEFAL code through processing of the TALYS results, possibly by merging it with experimental data or data from existing data libraries. All possible covariance data that the ENDF-6 format allows is included as well. For all this, a significant part of the ENDF-6 formats manual [8] was implemented in TEFAL.

4.3   TARES for resonance parameters and uncertainties

TARES is a program to obtain cross section information in the resonance range, solely used for the production of resonance parameters in the context of the TENDL library. Its primary goal was to be connected with TALYS to complete the low energy region. As TARES is used in the production of the TENDL library, it is part of the "T6" package. Since then, TARES is used in the production of the TENDL libraries and in the production of the so-called "random ENDF files", used to propagate nuclear data uncertainties following the "Total Monte Carlo" (TMC) approach [7]. It does not only perform original resonance analysis, but TARES is also using other codes such as CALENDF [9] for the generation of ladder of resonances and PREPRO [10] for the reconstruction of pointwise resonances. One of the particularity of TARES is to successfully linked all these codes for all isotopes of interest in TENDL.

It can be used from $Z = 3$ to $Z = 118$ to

- produce resonance parameters in the ENDF format (called MF2 or file 2). Mainly three formats are used: multi-level Breit Wigner (for actinides), Reich-Moore (for some non actinides) and LRF7 (for the majority of non actinides),
- produce resonance parameter uncertainties in the ENDF format (called MF32),
- produce cross section uncertainties in the resolved resonance range in the ENDF format (called MF33),
- produce random resonance parameters based on parameter uncertainties,
- reconstruct pointwise cross sections based on the resonance parameters in tabulated x-y format,
- reconstruct pointwise cross sections based on the resonance parameters in ENDF format (called MF3),
- produce grouped cross sections based on the resonance parameters in tabulated x-y format,

For every random sample performed by TASMAN, TARES can be called and produce its random parameters. However, often a pre-produced database is used.

4.4 TANES for prompt fission neutron spectra

TANES is a simple program to calculate the fission neutron spectrum based on the Los Alamos model [11]. The original Madland-Nix [12] or Los Alamos model for the calculation of prompt fission neutrons characteristics (spectra and multiplicity) has been implemented in a stand-alone module. The TANES code is using this stand-alone module, combined with parameter uncertainties (on the total kinetic energy, released energy and multi-chance fission probabilities) to reproduce and randomize the fission neutron spectrum. The program provides the central and random values for the fission neutron spectra at different incident energies (MF5) and their covariances (MF35). For covariance files, a simple Monte Carlo variation of the parameters is performed, using the Madland-Nix model. The random spectra (or more specifically their standard deviations) are then formatted into MF35.

For every random sample performed by TASMAN, TANES can be called and produce its random data. However, often a pre-produced database is used.

4.5 TAFIS for average number of fission neutrons

TAFIS calculates fission yields, prompt neutron emission from fission and other necessary fission quantities (kinetic energy of the fission products, kinetic energy of the prompt and delayed fission neutrons, total energy released by prompt and delayed gamma rays). For fission yields, it is using the systematics of fission-product yields from A.C. Wahl [13], combined with *ad hoc* uncertainties. It calculates the independent and cumulative fission yields at any incident energy up to 200 MeV and for different incident particles (spontaneous, neutrons, protons, deuterons, *etc*). Empirical equations representing systematics of fission-product yields are derived from experimental data. The systematics give some insight into nuclear-structure effects on yields, and the equations allow estimation of yields from fission of any nuclide in the range $Z = 90$ to 98 and $A = 230$ to 252. For neutron emission, different models are used depending on the energy range and are presented in Ref. [13]. The output of this program is a fission yield file with uncertainties, prompt and delayed neutron emission files for central and random values (MF1 MT452), a list of central and random fission quantities (MF1 MT458) and prompt neutron covariances (MF31). For the delayed neutron emission, as the correct physics is yet not included in TAFIS, a simple copy from existing libraries is done. For isotopes which are not included in other libraries, the delayed neutrons are simply copied from the nearest isotope. As for the MF35, a simple Monte Carlo variation of the model parameters is used to generate covariance files.

For every random sample performed by TASMAN, TAFIS can be called and produce its random data. However, often a pre-produced database is used.

5. Nuclear data uncertainties and covariances

The method for random parameter sampling, including all the resulting probability distributions and their moments have been described at length in various papers [5, 6]. Here we will only repeat the essential formula and focus more on the practical implementation in TASMAN.

5.1 Random uncertainty model

A short, simple mathematical formalism will suffice to explain the method of covariance generation from random model parameters. Often, a nuclear model calculation is performed with various model parameters $p_1, p_2, \ldots$, of which some are adjusted to obtain the optimal description of experimental data. An input file may for example contain the optical model parameters $p_1 = r_v$ (real central radius), $p_2 = a_v$ (real central diffuseness), for neutrons, protons, etc. followed by the level density parameters for all involved nuclides and so on. Let **p** be the vector of the L adjustable nuclear model parameters that are relevant to the problem under consideration, i.e.

$$\mathbf{p} = \{p_1, \ldots, p_l, \ldots, p_L\}. \tag{5.1}$$

In practice L may take on values from one up to several tens, or even hundreds. The case of many adjustable parameters occurs e.g. for complete optical model studies or when many residual nuclides, each characterized by their own set of parameters (level density, etc.), are involved in a calculation. Next, let σ be a vector of N calculated quantities,

$$\sigma = \{\sigma_1, \ldots, \sigma_i, \ldots, \sigma_N\}. \tag{5.2}$$

The obvious choice for the elements of σ are cross sections, and we will indeed generally use the term "cross sections" for σ. It should however be understood that σ may contain all other quantities of interest as well, such as angular distributions, differential energy spectra, etc. For

example, for a statistical study on parameters of the optical model and the associated observables, σ may contain total cross sections σ^{tot} and elastic angular distributions $d\sigma^{\text{el}}/d\Omega$ at various incident energies and outgoing angles. As another example, if we are interested in a covariance matrix for applied purposes (evaluated nuclear data file), σ may contain the total and partial cross sections for all open reaction channels at various incident energies. Whatever the objective, it is obvious that σ may contain a very large number of elements. The vector σ is a function T of the vector $\mathbf{p}$,

$$\sigma = T(\mathbf{p}), \tag{5.3}$$

where here T stands for the entire TALYS nuclear model code. The basis of the random sampling method is to let TALYS perform many calculations for the same projectile + target combination, whereby each time all L elements of the $\mathbf{p}$ vector are randomly sampled from a probability distribution. For example, this could be a uniform or normal distribution with a specific width Δp_l for each parameter p_l, i.e.

$$p_l^{(k)} = p_l^{(0)} \pm \Delta p_l, \quad l = 1, L, \tag{5.4}$$

where we have added a superscript (k) to denote the k-th TALYS run, i.e.

$$\sigma^{(k)} = T(\mathbf{p}^{(k)}). \tag{5.5}$$

Note that more general probability distributions can be used for the sampling, such as used for the TENDL library. The initial set of input parameters, the central values, is denoted as $\mathbf{p}^{(0)}$ and the corresponding initial calculated cross section vector as $\sigma^{(0)}$, obtained from the initial TALYS calculation. This initial set often represents one of the following two cases: (a) the best possible parameter set, adjusted to experimental data, obtained from a numerical and/or visual optimization procedure, or (b) a so-called global nuclear model calculation with all parameters equal to their default values. After the initial calculation, the first set of random parameters $\mathbf{p}^{(1)}$ will lead to a set of cross sections $\sigma^{(1)}$ that is different from $\sigma^{(0)}$. From the results, we can obtain the single covariance matrix element $(\sigma_i^{(1)} - \sigma_i^{(0)})(\sigma_j^{(1)} - \sigma_j^{(0)})$, which is a measure of the change of σ_j relative to the change of σ_i. In a Monte Carlo approach, such a single covariance matrix element alone is rather meaningless, since all parameters are simultaneously sampled from a random distribution, and no significant information can be available after one run. However, after performing many calculations, all statistical information like average values, uncertainties and covariance matrices will gradually become available.

As detailed in [kon15,kon19], the Bayesian Monte Carlo (BMC) method assigns a weight to each random sample, where the weight is a measure of the deviation of the random TALYS sample from experimental data. The probability that an experimental data set with uncertainties, represented by the vector σ is described by a nuclear model calculation driven by a parameter vector $\mathbf{p}$, is given by the likelihood function

$$\mathscr{L}(\mathbf{p}; \sigma) = \exp(-\chi^2/2), \tag{5.6}$$

where χ^2, an abbreviation of $\chi^2(\mathbf{p}; \sigma)$, is a measure for the deviation of model from experiment, weighted by the experimental covariance matrix. The expression for χ^2 is essential in the entire process and will be outlined in a separate section. With the random parameter samples the TALYS code is run to produce all cross sections, spectra, angular distributions etc. of interest. The $\chi^{2(k)}$ for the entire experimental data set under consideration is calculated so that the weight of each random

TALYS sample is given by

$$w^{(k)} = \frac{\mathscr{L}(\mathbf{p}^{(\mathbf{k})};\mathbf{x})}{\sum_{k=1}^{K} \mathscr{L}(\mathbf{p}^{(\mathbf{k})};\mathbf{x})}, \tag{5.7}$$

where K is the total number of TALYS runs needed for statistical convergence, and which automatically entails that $\sum_{k=1}^{K} w^{(k)} = 1$. The concept of BMC is thus rather straightforward: Run a model code K times and calculate $w^{(k)}$ for k=1,K. Each random sample k contains information about the specific combination of parameters that leads to a certain set of cross sections, and thus a value $\chi^{2(k)}$. Clearly, a large deviation between TALYS and experiment means a large χ^2 value and low $\mathscr{L}$ and $w^{(k)}$ for that particular combination of parameters. In this sense, knowledge on the parameter distribution is gradually built up. We obtain the full posterior probability distribution for each cross section and parameter, simply by applying the weights $w^{(k)}$ to each value of the sampled parameters and resulting cross sections.

The average covariance matrix for cross sections is given by

$$V_{ij} = \frac{1}{K} \sum_{k=1}^{K} w^{(k)} (\sigma_i^{(k)} - \bar{\sigma}_i)(\sigma_j^{(k)} - \bar{\sigma}_j), \quad i,j = 1,N, \tag{5.8}$$

and the average calculated cross sections are

$$\bar{\sigma}_i = \frac{1}{K} \sum_{k=1}^{K} w^{(k)} \sigma_i^{(k)}, \quad i = 1,N, \tag{5.9}$$

for which one would expect that $\bar{\sigma}_i \approx \sigma_i^{(0)}$ for all elements i, if the non-linearities are not too strong. Similarly, the average *relative* covariance matrix for cross sections can be obtained,

$$R_{ij} = V_{ij}/(\bar{\sigma}_i \bar{\sigma}_j), \quad i,j = 1,N. \tag{5.10}$$

The extent to which the calculated results co-vary is most clearly represented by the average *correlation* matrix for cross sections

$$C_{ij} = V_{ij}/\sqrt{V_{ii}V_{jj}}, \quad i,j = 1,N, \tag{5.11}$$

whose elements take on values between -1 (complete anti-correlation), 0 (no correlation) and 1 (complete correlation). As an example, if by changing model parameters the total neutron non-elastic cross section at a certain incident energy remains equal, while the (n,2n) cross section increases, we expect the (n,n') cross section to decrease to compensate for this, and this anti-correlation will be reflected by a negative value of the C_{ij} matrix element that relates $\sigma_{n,n'}$ and $\sigma_{n,2n}$ at that particular energy. The diagonal elements of the relative covariance matrix R represent the variance and its square root the uncertainty. Hence, the final calculated cross sections together with their uncertainties can be expressed as

$$\sigma_i^{\text{final}} = \bar{\sigma}_i(1 \pm \sqrt{R_{ii}}), \quad i = 1,N. \tag{5.12}$$

Actually, the values of Eqs. (5.8)-(5.12) can be updated, and inspected, after every TALYS run, and statistical convergence is reached when the running index k approaches K. For K, an appropriate value needs to be determined, and we usually continue until the average values for the statistical quantities (average, variance, skewness) no longer vary outside a small range.

We can construct similar statistical quantities for the L input parameters that we use in every run, such as the average covariance matrix for model parameters,

$$P_{lm} = \frac{1}{K}\sum_{k=1}^{K} w^{(k)}(p_l^{(k)} - \bar{p}_l)(p_m^{(k)} - \bar{p}_m), \quad l,m = 1,L, \tag{5.13}$$

where the average model parameters are,

$$\bar{p}_l = \frac{1}{K}\sum_{k=1}^{K} w^{(k)} p_l^{(k)}, \quad l = 1,L, \tag{5.14}$$

where we should obtain $\bar{p}_l \approx p_l^{(0)}$. Similarly, we have the relative covariance matrix for model parameters,

$$R_{lm}^P = P_{lm}/(p_l^{(0)} p_m^{(0)}), \quad l,m = 1,L, \tag{5.15}$$

and the correlation matrix for model parameters,

$$C_{lm}^P = P_{lm}/\sqrt{P_{ll}P_{mm}}, \quad l,m = 1,L. \tag{5.16}$$

The final model parameters with their uncertainties are

$$p_l^{\text{final}} = \bar{p}_l(1 \pm \sqrt{R_{ll}^P}), \quad l = 1,L. \tag{5.17}$$

Sample case 5.1 n + ^{93}Nb: Basic, unweighted uncertainty calculation

The first sample case is given below. It concerns a default calculation for neutrons on ^{93}Nb, with an energy grid specified in the file *energies*. As explained in this manual, the **#change** keyword is essential. After this line, TASMAN keywords, always starting with a '#', can be given while comment lines after **#change** need to start with '##'. Keyword **#mode 1** specifies that we are interested in an uncertainty calculation, i.e. a statistical spread of the final results. **#band y** means that the final cross section files contain, besides the central values, the lower and upper 1-sigma uncertainty band. In this sample case, only 2 parameters are varied, namely the radius and diffuseness of the real volume central OMP. Since only the central value, for both parameters 1., is specified, the default uncertainties for these parameters are used.

```
#
# General
#
projectile n
element nb
mass 93
energy energies
channels y
filechannels y
bins 20
#
# TASMAN keywords start here
#
#change
```

```
#mode 1
#band y
##
## Parameters to be varied
##
rvadjust n 1.
avadjust n 1.
```

This input file called *tasman.inp* can simply be run as follows:

tasman < **tasman.inp** > **tasman.out**

The *tasman.out* file starts with a display of the version of TASMAN you are using, the name of the author, and the Copyright statement. Also the physics dimensions used in the output are given. As can be seen in the output below, the first section of the output is a print of the keywords/input parameters. This is done in two steps: In the first block an exact copy of the input file as given by the user is returned. In the next block 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 we print the random parameters used at each run. At the end of this simple output, the total calculation time is printed, followed by a message that the calculation has been successfully completed.

```
   TASMAN-2.0 (Version: December 29, 2023)

 Copyright (C) 2023  A.J. Koning

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


 ########## USER INPUT ##########

 USER INPUT FILE

 #
 # general
 #
 projectile n
 element nb
 mass 93
 energy energies
 channels y
 filechannels y
```

```
bins 20
#
# tasman keywords start here
#
#change
#mode 1
#band y
##
## parameters to be varied
##
rvadjust n 1.
avadjust n 1.

USER INPUT FILE + DEFAULTS

Keyword          Value   Variable     Explanation


#
# Main TASMAN keywords
#
#mode               1    mode         mode for TASMAN: Monte Carlo covariances
#Ntalys            10    Ntalys       number of TALYS runs
#Nburn             10    Nburn        number of burn-in runs
#Nhigh             10    Nhigh        number of high energy runs
#Ehigh       1000.000    Ehigh        energy cutoff of high energy runs
#Liso               0    Liso         number of the isomeric state
#maxgam             4    maxgam       maximum number of discrete levels with g
#minbar             0    minbar       minimum number of fission barriers
#maxbar             3    maxbar       maximum number of fission barriers
#cross              y    flagcross    flag for covariances or optimization of
#angle              n    flagangle    flag for covariances or optimization of
#legendre           n    flagleg      flag for covariances or optimization of
#spectra            n    flagspectra  flag for covariances or optimization of
#gamma              n    flaggamma    flag for covariances or optimization of
#residual           n    flagresidual flag for covariances or optimization of
#prod               n    flagprod     flag for covariances or optimization of
#production         n    flagproduct  flag for covariances or optimization of
#discrete           n    flagdiscrete flag for covariances or optimization of
#isomer             n    flagisomer   flag for covariances or optimization of
#ompinc             y    flagompinc   flag to vary OMP parameters for incident
#omponly            n    flagomponly  flag for covariances or optimization for
#integral           n    flagintegral flag for covariances or optimization of
#macs               n    flagmacs     flag for covariances or optimization of
#gamgam             n    flaggamgam   flag for covariances or optimization of
#psf                n    flagpsf      flag for optimization of photon strength
```

```
#ld                  n       flagld        flag for optimization of level density p
#sample              n       flagsample    flag for sampling experimental data
#talysversion talys          talysversion  version of TALYS
#user    Arjan Koning         user          user for this calculation
#source TASMAN                source        source for this calculation
#format YANDF-0.1             oformat       format for output
#
# Keywords for mode 1
#
#offset               0       offset        offset for numbering or random files (TM
#tmcoffset            0       tmcoffset     offset for starting creation of ENDF-6 f
#seed        20032703         seed          seed for random number generator
#cwidth   1.00000E+00         cwidth        scaling constant for parameter variation
#fiscor   3.00000E+00         fiscor0       adjustment parameter for fission uncerta
#ecis                y       flagecis      flag to repeat ECIS calculations in a se
#block               y       flagblock     flag to block spectra, angle and gamma f
#gauss               y       flaggauss     flag to sample parameters from a Gaussia
#save                n       flagsave      flag to save all input and output files
#tmc                 n       flagtmc       flag for Total Monte Carlo (ENDF-6 data
#select              n       flagselect    flag for varying only parts of an ENDF-6
#covrand             n       flagcovrand   flag for covariance ENDF-6 file after ev
#gpf                 y       flaggpf       flag for creation of ENDF general purpos
#sdefault            y       flagsdefault  flag for creation of ENDF general purpos
#defonly             y       defonly       flag for creation of ENDF default file o
#s20                 n       flags20       flag for creation of ENDF general purpos
#s30                 n       flags30       flag for creation of ENDF general purpos
#s60                 n       flags60       flag for creation of ENDF general purpos
#mt                  n       flagmt        flag for creation of ENDF general purpos
#act                 n       flagacf       flag for creation of ENDF activation fil
#eaf                 n       flageaf       flag for creation of EAF activation file
#prepro              y       flagprepro    flag to run PREPRO codes
#njoy                y       flagnjoy      flag to run NJOY
#proconly            y       flagproconly  flag for processing of ENDF default file
#purr                n       flagpurr      flag to run PURR in NJOY
#resonance           y       flagres       flag for variation of resonance paramete
#runtares            n       flagruntares  flag to produce resonances in TASMAN wit
#nubar               n       flagnubar     flag for variation of nubar values for E
#runnubar            n       flagrunnubar  flag to produce nubar in TASMAN with TAF
#fns                 n       flagfns       flag or variation of fission neutron spe
#tefalversion tefal          tefalversion  version of TEFAL
#taresversion tares          taresversion  version of TARES
#tafisversion tafis          tafisversion  version of TAFIS
#tanesversion tanes          tanesversion  version of TANES
#tareslib default            tareslib      library name for adoption of resonance p
#background                   background    library name for adoption of background
```

```
#lrf7                n      flaglrf7    flag for resonance representation
#tafislib none              tafislib    library name for adoption of nubar value
#taneslib none              taneslib    library name for adoption of fission neu
#mcak                n      flagmcak    flag to use sampling from arbitrary dist
#readpar             n      flagreadpar flag to read in parameter distribution
#parameters          n      flagparvary flag to vary tabulated nuclear model par
#allpar              n      flagallpar  flag to vary all nuclear model parameter
#weight              y      flagweight  flag to use weights for random samples
#eweight             n      flageweight flag to use energy-dependent weights
#metropolis          n      flagmetro   flag for Metropolis sampling
#kdburn              y      flagkdburn  flag to use KD03 OMP function for burn-i
#magnet              n      flagmagnet  flag to push TALYS data to experimental
#global              n      flagglobal  flag to use global TALYS cross sections
#autoinclude         y      flagautoinc flag to automatically in- or exclude imp
#edependent          y      flagedep    flag for energy dependence of uncertaint
#equisample          y      flagequisam flag to use equal binning uniform sampli
#gofmode             1      gofmode     1: exp(-0.5*G) 2: exp(-0.5*G/G0) 3: exp(
#maxexpsets    1000000      maxexpsets  number of experimental data sets before
#average             2      average     average cross sections 1: average, 2: fi
#inponly             n      flaginponly flag to produce only TALYS input files,


 Parameters for input file:     0


rvadjust     n    1.00000
avadjust     n    1.00000


 Parameters for input file:     1


rvadjust     n    1.00596
avadjust     n    1.01529


 Parameters for input file:     2


rvadjust     n    0.99623
avadjust     n    0.99778


 Parameters for input file:     3


rvadjust     n    1.00478
avadjust     n    0.98480


 Parameters for input file:     4


rvadjust     n    0.99407
avadjust     n    1.00900
```

```
  Parameters for input file:     5


 rvadjust     n    0.98180
 avadjust     n    0.99771


  Parameters for input file:     6


 rvadjust     n    0.99850
 avadjust     n    0.99093


  Parameters for input file:     7


 rvadjust     n    1.01459
 avadjust     n    1.00360


  Parameters for input file:     8


 rvadjust     n    0.99788
 avadjust     n    1.00788


  Parameters for input file:     9


 rvadjust     n    1.00017
 avadjust     n    1.01027


  Parameters for input file:    10


 rvadjust     n    1.01037
 avadjust     n    1.01702


 Execution time:  0 hours   0 minutes   0.93 seconds


 This TASMAN run has finished
```

More interesting are the various output files that have been generated. Apart from the standard output files of TALYS, like *totalxs.tot*, *xs200000.tot*, etc. you will also find various files whose names end with *.ave*. These are the average cross section files, including an uncertainty band. These results are obtained after 10 random TALYS calculations.

A full covariance matrix, correlating all channels and energies with each other, is available. For example in *cov_intra.ave* we have:

```
# header:
#   title: Nb93(n,tot) cross section
#   source: TASMAN
```

```
#    user: Arjan Koning
#    date: 2023-12-16
#    format: YANDF-0.1
# target:
#    Z: 41
#    A: 93
#    nuclide: Nb93
# reaction:
#    type: (n,tot)
#    ENDF_MF: 3
#    ENDF_MT: 1
# covariance:
#    type: in-channel covariance
# datablock:
#    quantity: cross section
#    columns: 5
#    entries: 100
##      E_a            E_b            Rcov           Ccov           Vcov
##     [MeV]          [MeV]           []             []             []
     1.000000E-01   1.000000E-01   9.706488E-03   1.000000E+00   8.987610E+05
     1.000000E-01   2.000000E-01   9.188086E-03   9.894124E-01   9.013705E+05
     1.000000E-01   5.000000E-01   6.335274E-03   9.482473E-01   5.574611E+05
     1.000000E-01   1.000000E+00   5.746896E-03   9.581335E-01   3.820715E+05
     1.000000E-01   2.000000E+00   4.502018E-03   9.734957E-01   2.111240E+05
     1.000000E-01   3.000000E+00   1.400545E-03   9.506426E-01   5.472231E+04
     1.000000E-01   4.000000E+00  -5.533088E-04  -8.405757E-01  -1.970543E+04
.....................................
```

Sample case 5.2   **n + ^{93}Nb: Experiment-weighted uncertainty calculation**

In this sample case we add some more variation of the parameters and some more TASMAN features. The input file is as follows:

```
#
# General
#
projectile n
element nb
mass 93
energy energies
channels y
filechannels y
bins 20
outangle y
fileangle 0
fileangle 1
```

```
fileangle 2
fileangle 3
fileangle 4
#
# TASMAN keywords start here
#
#change
#mode 1
#ntalys 20
#seed 3042015
#band y
#save y
#angle y
#isomer y
#expinclude all
##
## Parameters to be varied
##
rvadjust n 1.
avadjust n 1.
v1adjust n 1. 0.05
v2adjust n 1.
rwdadjust n 1. 0.10
awdadjust n 1. 0.05
aadjust 41 94 1. 0.15
aadjust 41 93 1. 0.15
aadjust 40 93 1. 0.15
```

Note that we have added angular distributions for the elastic channel and first 4 inelastic levels. In the TASMAN part, we now perform 20 random calculations, instead of the default 10, and to avoid starting with the same results as sample case 1 we change the random seed. Moreover, we request to save the results of all individual TALYS runs, and output average + standard deviation files for angular distributions and isomeric production. Next, we have added variation of more OMP parameters and also a few level density parameters. For 3 OMP and 3 level density parameters we overrule the built-in variation by adding the relative standard deviation to the input lines.   ∎

5.1.1 Goodness-of-fit estimator

The issue of an appropriate goodness-of-fit estimator is discussed at length in [5] and [6]. A χ^2 estimator is used which takes approximately into account (1) model defects, (2) absence of experimental covariance data, (3) wrong experimental data, (4) unmeasured reactions channels.

We use the following pragmatic formula for χ^2 for each random sample k,

$$
\begin{aligned}
\chi^{2(k)} &= \frac{N}{\chi^{2(0)}} \sum_{c=1}^{C} \sum_{m=1}^{M_c} \frac{w_m}{N_{Mc}} \sum_{i=1}^{N_{Mc}} \left(\frac{\sigma_{Tci}^{(k)} - \sigma_{Ecmi}}{d\sigma_{Ecmi}} \right)^2 \\
&+ \frac{1}{N_{Tc}} \sum_{i=1}^{N_{Tc}} \left(\frac{\sigma_{Tci}^{(k)} - \sigma_{Tc0}}{d\sigma_{Tc0}} \right)^2,
\end{aligned}
\tag{5.18}
$$

where $\chi^{2(0)}$ serves as a damping value belonging to the TALYS run with central values for all parameters, as a remedy for the model defect, c runs over the C reaction channels with experimental data, m runs over M_c experimental data sets per reaction channel, where each data set has N_{Mc} data points, N is the total number of data points and w_m is a weight to discard or include particular experimental data sets. The second term can be considered as the prior since it covers pseudo-experimental data, with N_{Tc} the number of data points for each theoretical, or pseudo-experimental, channel.

5.2 Global predictive power and pseudo-experimental data

In Ref. [5] it was outlined how we established the predictive power of TALYS for neutron-induced reactions. We perform one 'blind' TALYS calculation for each target nuclide with all nuclear model parameters taking on their global, default values and compare the results with the *entire* EXFOR database for cross sections. By binning the deviations of model from experiment in histograms we establish the average deviation of TALYS from experiment, as a function of reaction channel and incident energy, averaged over all nuclides close to the stability line with experimental data. The predictive power of TALYS for neutron-induced reactions is then given by the following function for the standard deviation s, in %, per reaction channel

$$
\begin{aligned}
s(E) &= s_{min} + (a - s_{min}) \exp(-bE') \\
&+ \max(c - s_{min}, 0) \frac{E''^2}{(E''^2 + d^2)},
\end{aligned}
\tag{5.19}
$$

where s_{min} is the minimal standard deviation for the global prediction, often obtained near the peak of the excitation function, and generally somewhat lower than the average value for the reaction channel s_{ave}. The second term depends on $E' = E - E_{thr}$, with E_{thr} the threshold incident energy, in practice taken as the energy where the TALYS cross section crosses the value of 1 mb. (For reactions with positive Q-values, like (n, p) and (n, α) channels, we obviously apply this only to the rising part of the excitation function, and not in the thermal range). The exponent, scaled by parameter b, mimics the sharp decrease in relative deviation between TALYS and experimental data when the energy exceeds threshold. Parameter a is a measure for the uncertainty at threshold. The last term mimics the change in uncertainty at higher energies, depending on parameters c and d, and $E'' = E' - E_c$, where E_c is a measure for the energy at which the relative uncertainty is minimal, i.e. generally at the highest cross section value. The parameters for the various reaction channels are given in Table 5.1.

Note that almost *all* EXFOR data, i.e. neutron reactions on all target nuclides between F and Fm, determine the parameters of this global formula. Eq.5.19 can then be applied to one nuclide in particular to create pseudo-experimental data for that target nucleus, i.e. we fill every reaction channel with central values provided by the global TALYS calculation for that nucleus and assign uncertainties to each point using formula (5.19). In some sense, we create an entire "EXFOR"

Table 5.1: Global cross section uncertainties per reaction channel from default TALYS calculations; average deviation and parameters for energy-dependent variation, see Eq. (5.19). The relative deviations s_{ave}, s_{min}, a and c are given in %, b is a dimensionless factor, while the energies d and E_c are given in MeV.

Reaction	s_{ave}	s_{min}	a	b	c	d	E_c
(n,tot)	6	8	60	0.3	0	12	6
(n,el)	10	12	60	0.3	0	12	6
(n,non)	10	12	60	0.3	0	12	6
(n,inl)	50	12	100	1	100	12	5
(n,γ)	62	40	60	0.3	0	20	20
(n,2n)	25	24	100	1	100	15	6
(n,3n)	150	40	100	1	100	12	6
(n,f)	110	50	100	1	100	12	6
(n,p)	53	34	100	1	100	12	6
(n,α)	120	45	100	1	100	12	6

database with pseudo-experimental data, and this is built into TASMAN as an essential part of BMC.

5.2.1 Exotic nuclides

For completeness, we also mention the issue of uncertainties for exotic nuclides. The estimated global predictive power of TALYS outlined above would apply to nuclides for which experimental data exist, i.e. for stable nuclides or nuclides close to them. It can not be expected that the same predictive power applies if we move away from the line of stability, if only because the validity of the nuclear models becomes dubious. A previous attempt [14] to take this increase in uncertainty into account consisted in an ad-hoc empirical increase of model parameter uncertainty, which was assumed to be a function of the distance of the nuclide under consideration to the stability line. The following expression for a target nucleus (Z,A) is applied

$$
\begin{aligned}
s_{c,ave}^{glob}(Z,A) &= s_{c,ave}^{glob}(Z,A_L).D(A), \quad \text{if } A < A_L \\
&= s_{c,ave}^{glob}(Z,A_H).D(A), \quad \text{if } A > A_H
\end{aligned}
$$

(5.20)

where A_L and A_H are the lightest and heaviest stable isotope, respectively. The function $D(A)$ has a specific numerical form which increases gradually from 1 at $A_{L,H}$ to 5 at the driplines. This factor of 5 is completely arbitrary. We note however that a more proper approach would be to sample the *models* together with the parameters, instead of sampling the parameters only. Various theoretical models for the OMP, level densities, gamma ray strength functions, etc. which are available in TALYS could be randomly used to obtain cross section distributions, again with their mean and variance. The resulting standard deviation is then assigned as the cross section uncertainty for that particular nuclide. Next, we may choose our standard set of nuclear models again and with the usual Monte Carlo sampling technique we obtain the associated parameter distributions. Sampling models is possible with TASMAN but has not yet been used to accomplish the above.

5.3 Sensitivity analysis

We can relate the variation of the cross sections to the variation of the model parameters. This can be done through the relative sensitivity matrix S. There are two ways to calculate the sensitivity matrix with TASMAN. The first is by means of the classical relation

$$S_{il} = \frac{w^{(k)}(\sigma_i^{(1)} - \sigma_i^{(0)})}{(p_l^{(1)} - p_l^{(0)})} \frac{p_l^{(0)}}{\sigma_i^{(0)}}, \quad i = 1, N, \quad l = 1, L. \tag{5.21}$$

where $\sigma_i^{(1)}$ is the cross section obtained using parameter $p_l^{(1)}$. In practice, for this option we keep all TALYS input parameters at their default values and move the parameters one by one to obtain the S_{il} elements. This quantity sheds light on the relative importance of the various model parameters. S is dimensionless and directly reflects the dependence of a cross section σ_i on the model parameter p_l. For example, a value of -3.5 for the element S_{il} means that if the model parameter p_l is increased by 10% , the cross section σ_i decreases by 35 % . A different way is to build up the sensitivity matrix during the random runs, i.e. similar to obtaining the average covariance matrix of Eq. (5.8).

$$S_{il} = \frac{\sum_{k=1}^{K}(p_l^{(k)} - \bar{p}_l)(\sigma_i^{(k)} - \bar{\sigma}_i)}{\sum_{k=1}^{K}(p_l^{(k)} - \bar{p}_l)^2} \frac{\bar{p}_l}{\bar{\sigma}_i}, \quad i = 1, N, \quad l = 1, L. \tag{5.22}$$

Again, S will converge to its final value after many calculations. Actually, it is the statistical convergence of S that determines the appropriate value for K. Note however that this approach will only work for the most sensitive parameters, the S values for insensitive parameters are unreliable since they drown in the Monte Carlo noise of the sensitive parameters. One advantage of the Monte Carlo approach is that each σ_i generally does not depend on all parameters. This means that in one and the same run we contribute to the statistical convergence of uncorrelated σ_i simultaneously. If a particular cross section depends on only 1 or 2 parameters, then the average of the associated covariance matrix elements will be obtained quicker than for a cross section that depends on many parameters. This information is contained in S. We thus have to wait for the convergence of the cross section that depends on the most parameters, until we can adopt Eqs. (5.8)-(5.17) as the final results.

Sample case 5.3 n + ^{120}Sn: Linear sensitivity analysis

In this sample case we create the sensitivity matrix of cross sections to model parameters. The input file is as follows:

```
#
# TALYS input file generated by Autotalys
#
projectile n
element    Sn
mass       120
energy     energies
partable y
bins 20
#
# Output of extra channels
```

```
#
channels y
filechannels y
#
# Reduce output for activation-only calculation
#
outspectra n
outangle n
ddxmode 0
outdiscrete n
#
# TASMAN
#
#change
#mode 2
##
## Parameters to be varied
##
v1adjust       n   1.00000
v2adjust       n   1.00000
rvadjust       n   1.00000
avadjust       n   1.00000
gnadjust          50 121    1.00000
gpadjust          50 121    1.00000
M2constant         1.00000
gamgamadjust      50 121    1.00000
aadjust           50 121    1.00000
aadjust           50 120    1.00000
```

Note that we have only include 10 parameters. If you want, you can paste parts of the TALYS output file *parameters.dat* to the end of this input file, at the expense of waiting longer for the result of course.

Besides the standard TALYS output file, various files starting with 'sens' have been created:
- sensitivity files per reaction channel, e.g. *sens_totalxs.tot, sens_xs000000.tot*, etc. gives the sensitivity per incident energy for each model parameter
- sensitivity file per incident energy and tabulated per reaction channel *sensitivity.E* and averaged over all reaction channels *sensitivity.Eave*
- sensitivity file per reaction channel and incident energy and tabulated per parameter *sensitivity.xs* and averaged over all incident energies *sensitivity.xsave*.
- sensitivity file per parameter and reaction channel and tabulated per incident energy *sensitivity.par* and averaged over all incident energies *sensitivity.parave*.
- sensitivity per parameter averaged over all reaction channels and incident energies *sensitivity.all*.

To learn about a particular parameter sensitivity for a reaction channel at one incident energy I usually consult *sensitivity.xs*. ∎

5.4 Uncertainties of nuclear model parameters

All parameters of the models built in TALYS have what we call "default uncertainties". It means that when these uncertainties are applied to the parameters in Eq. (5.4), the resulting cross sections, angular distributions, etc. display uncertainty bands which on average overlap different experimental data sets. In other words, the default uncertainties have been set to simulate the current discrepancy between measurements and between models and measurements. Two comments are directly in order: (1) of course, this can never be full-proof, since certain reaction channels are measured better than others, (2) it is clear that when zooming in on a particular nucleus, the various parameter uncertainties need to be altered. The number of parameters available in TALYS is quite large, and it is expected that the global parameter uncertainties given here will often be revised. We will discuss the parameter uncertainties model by model.

5.4.1 Optical Model

The most important quantities calculated by the optical model are
- the shape elastic angular distribution,
- the reaction cross section,
- the total cross section,
- inelastic cross sections and angular distributions using either a coupled-channels (deformed nuclei) or DWBA (spherical nuclei) approach,
- transmission coefficients, for compound nucleus decay and pre-equilibrium emission,

and a good optical model is expected to reliably predict these quantities for energies and nuclides for which no measurements exist. All optical model calculations in TALYS are performed by ECIS-06 [15], which is implemented as a subroutine. Unless otherwise specified, the optical model potential is that of Koning-Delaroche [16] for non-actinides and Soukhovitskii et al. [17] for actinides. As detailed in [16], local parameterizations, i.e. per target nuclide, are used as preference, and if they do not exist due to absence of experimental data the global parameterization is used. In addition, by default the dispersive (unpublished) variant of the potential is taken with the parameters as stored in the optical model database of TALYS.

For every nuclide at or near the valley of stability, a coupling scheme has been stored in the TALYS database, so that upon request automatically a coupled-channels calculation can be invoked. If a specific deformed potential is not available in the database, then for deformed non-fissile nuclides, e.g. in the rare earth region, we take the local or global spherical potential of Koning-Delaroche and subtract 15% from the imaginary surface potential parameter.

The other direct reaction methods used, depending on the target nucleus, such as DWBA, weak-coupling and various rotational and vibrational coupled-channels schemes are discussed at length in [14].

For deuterons, tritons, and Helium-3 particles, we use a simplification of the folding approach of Watanabe [18], see also Ref. [19]. We take the nucleon OMP's described above, either local or global, as the basis for these complex particle potentials. A change compared to some of the recent TENDL versions is that for alpha particles we now use the potential of Avrigeanu et al. [20], which provides a globally good description of alpha particles for both projectiles as ejectiles.

The uncertainties of the OMP parameters were estimated for a task of the RIPL project, and the results were published in [21]. The global uncertainties were basically obtained using trial and error: they were set such that for a range of total cross sections and angular distributions, the collection of random curves overlaps as much as possible the data and their uncertainties or discrepant data sets. The parameter uncertainties given in Table 5.2 correspond to those defined in KD03. Of course,

the global uncertainty of every parameter can be overruled in a TASMAN input file, leading to a different spread of the results.

Table 5.2: Uncertainties of neutron optical model parameters, given as fraction (%) of the absolute value.

Parameter	NDS-2012 uncertainty (%)	Prior uncertainty (%)
r_V^n	2	10
a_V^n	2	10
v_1^n	2	10
v_2^n	3	15
v_3^n	3	15
v_4^n	5	25
w_1^n	10	50
w_2^n	10	50
r_D^n	3	15
a_D^n	4	20
d_1^n	10	50
d_2^n	10	50
d_3^n	10	50
r_{SO}^n	10	50
a_{SO}^n	10	50
v_{so1}^n	5	25
v_{so2}^n	10	50
w_{so1}^n	20	100
w_{so2}^n	20	100

5.4.2 Level Density

For most nuclides, the level density is taken to be of the Backshifted Fermi Gas form, although for some nuclides we use a combined Constant Temperature + Fermi gas form, as detailed in Ref. [22]. There is a level density prescription for the target nucleus, compound nucleus and all residual nuclides that can be reached in the reaction, and for each of these nuclides this involves typically 3 or 4 parameters: the level density parameter a, the spin cutoff parameter σ^2, and the pairing shift P (or E_0 and T in the constant temperature model). For deformed nuclides, extra parameters for collective enhancement are involved. For nuclides that may fission, the above number of parameters is multiplied by the number of fission barriers, since on top of each barrier a discrete and continuum level scheme is built. Although models for using microscopic level density tables are available, most TALYS analyses are still based on phenomenological models, since there is more experience with parameter adjustment for these models, as changes from nucleus to nucleus are smoother for global phenomenological models. Since these microscopical level densities, which we will call ρ_{micro}, have not been adjusted to experimental data, we add adjustment flexibility through a scaling function, i.e.

$$\rho(E_x, J, \pi) = \exp(c\sqrt{E_x - \delta})\rho_{HFM}(E_x - \delta, J, \pi) \tag{5.23}$$

where by default $c = 0$ and $\delta = 0$ (i.e. unaltered values from the tables). The "pairing shift" δ 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 δ together gives adjustment flexibility at both low and higher energies.

Table 5.3 gives the uncertainties of the various level density parameters.

Table 5.3: Uncertainties of level density parameters, given as fraction (%) of the absolute value.

Parameter	NDS-2012 uncertainty (%)	Prior uncertainty (%)
a	11.25-0.03125.A	5.(11.25-0.03125.A)
σ^2	30	120
α	30	120
β	30	120
γ	30	120
δ_W	± 1 MeV	± 5 MeV
R_σ	30	120
E_0	20	100
E_M	20	100
T	10	50
δ	± 2 MeV	± 10 MeV
K_{rot}	80	400

5.4.3 Gamma-ray Strength Function

The gamma-ray strength function is often taken as that of Kopecky-Uhl [23], of which the most important parameters are the height, width, and energy of the Giant Dipole Resonance. Like the particle transmission coefficients that emerge from the optical model, gamma-ray transmission coefficients enter the Hauser-Feshbach model for the calculation of the competition of photons with other particles. For $E1$-transitions, GDR parameters for various individual nuclides exist in the RIPL database, and these are stored in the nuclear structure database of TALYS. Some 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 transitions other than $E1$, systematic formulae compiled by Kopecky [24], for the resonance parameters are used.

At sufficiently low incident neutron energies, the average radiative capture width Γ_γ is due entirely to the s-wave interaction, and this width at the neutron separation energy is used to normalize the gamma-ray transmission coefficients. These Γ_γ values are, when available, read from our nuclear structure database. For nuclides for which no experimental value is available, values come from interpolation, in mass, between available experimental values.

Similar to level densities, TALYS also provides microscopic options for $E1$ radiation, with gamma-ray strength functions calculated according to the QRPA model, see Ref. [25]. Here, the adjustment is as follows

$$f_{E1}(E_\gamma) = f^{nor} f_{\text{micro}}(E_\gamma + E_{\text{shift}}) \tag{5.24}$$

where by default $f^{nor} = 1$ and $E_{\text{shift}} = 0$ (i.e. unaltered values from the tables). Since these

microscopical strength functions have not yet been adjusted to experimental data, they are not yet used as default, but in the near future they will be.

Table 5.4 gives the uncertainties of the various gamma-ray strength function parameters.

Table 5.4: Uncertainties of gamma-ray strength function parameters, given as fraction (%) of the absolute value.

Parameter	NDS-2012 uncertainty (%)	Prior uncertainty (%)
Γ_γ	20	100
$\sigma_{E\ell}$	20	100
$\Gamma_{E\ell}$	20	100
$E_{E\ell}$	10	50
E_{nor}	20	100
E_{shift}	$\pm\, 0.8$ MeV	$\pm\, 4$ MeV

5.4.4 Compound Nucleus Model

For low incident energies, the Hauser-feshbach model with width fluctuation correction has been implemented. We still stick to the conclusion of [26] that Moldauer's model is appropriate for this. For compound nucleus angular distributions, the Blatt-Biedenharn model is included. TALYS adds the resulting Legendre coefficients to those of the shape elastic scattering and direct inelastic scattering to yield the total Legendre coefficients at each incident energy. At higher incident energies, the binary compound nucleus cross section becomes small, and compound nucleus evaporation proceeds mainly via the multiple emission Hauser-Feshbach model, which is a bit simpler than that of the binary reaction since width fluctuation corrections and angular distributions are not included. A lot of effort has been invested in the early years of TALYS development to make these calculations as efficient as possible, since often more than 95% of the calculation time is spent inside the Hauser-Feshbach loops. Essential is to store intermediate data, containing a combination of level densities and transmission coefficients, in arrays outside the various loops, and to use integer, instead of real, manipulation for angular momentum variables.

5.4.5 Fission

For fission, the default model implemented in TALYS is based on the transition state hypothesis of Bohr and the Hill-Wheeler expression. As described in e.g. Ref. [14], this represents the probability of tunneling through a single barrier with height B_f and width $\hbar\omega_f$ for a compound nucleus with excitation energy E_x. It reads

$$T_f(E_x) = \frac{1}{1 + \exp\left[-2\pi\frac{(E_x - B_f)}{\hbar\omega_f}\right]}, \tag{5.25}$$

For a transition state with excitation energy ε_i above the top of the same barrier, one has

$$T_f(E_x, \varepsilon_i) = \frac{1}{1 + \exp\left[-2\pi\frac{(E_x - B_f - \varepsilon_i)}{\hbar\omega_f}\right]}, \tag{5.26}$$

which means that the barrier is simply shifted up by ε_i.

This provides the input for formulae for double and triple humped barriers. These transmission coefficients enter the Hauser-Feshbach model to compete with the particle and photon transmission coefficients. The fission calculation in TALYS is steered by many parameters, such as for each fission barrier its height, width and level density parameters for each nuclide in the chain. In addition class II/III states for resonance structure in between the wells may alter the prediction. Experimental parameters are taken from the RIPL database [24], which contains a large collection of actinide fission barrier heights and curvatures for both the inner and outer barrier based on a fit to experimental data. It has to be noted here that even with a seemingly robust set of parameters for all these fission and level density parameters it is very hard to describe the fission channel, and we consider this as one of the weak points of TALYS. Often, we have to normalize the TALYS calculations to other well-evaluated nuclear data libraries to get a satisfactory answer.

Table 5.5 gives the uncertainties of the various fission parameters.

Table 5.5: Uncertainties of fission parameters, given as fraction (%) of the absolute value.

Parameter	NDS-2012 uncertainty (%)	Prior uncertainty (%)
B_f	10	50
$\hbar\omega_f$	10	50

5.4.6 Pre-equilibrium Model

For pre-equilibrium reactions, TALYS makes use of the two-component exciton model model[27], involving one general matrix element M^2 steering the damping from simple to more complex exciton states, and the single-particle state density parameters g_ν and g_π which determine the particle-hole state density for each nuclide considered in the reaction chain. The latter two parameters are often used to adjust (n,2n) and (n,p) cross sections. The emission rates that appear in the exciton model are based on the same OMP transmission coefficients as those used in the other reaction mechanisms, such as compound nucleus decay. The pre-equilibrium stage 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. The two-component exciton model has been proven to be a powerful method to describe this process. As described in [14], for deuteron up to alpha emission also stripping, pick-up, break-up and knock-out parameters are required as additions to the aforementioned exciton model. Also an exciton model for photon emission and multiple pre-equilibrium reactions are included.

Table 5.6 gives the uncertainties of the various pre-equilibrium parameters.

5.5 Exact uncertainty propagation: Total Monte Carlo

The propagation of probability distributions of nuclear model parameters to probability distributions of the output of TALYS, such as cross sections, angular distributions etc. is exact. As explained above, we can then take weighted averages after the Monte Carlo sampling so that we get average cross sections, their standard deviations and covariance matrices, as well as for the TALYS model parameters. Total Monte Carlo, first introduced in [7] takes the exact uncertainty propagation once step further: for each random TALYS input file, the corresponding TALYS output file is processed

Table 5.6: Uncertainties of pre-equilibrium parameters, given as fraction (%) of the absolute value.

Parameter	NDS-2012 uncertainty (%)	Prior uncertainty (%)
M^2	30	150
$R_{\pi\pi}$	30	150
$R_{\pi\nu}$	30	150
$R_{\nu\pi}$	30	150
$R_{\nu\nu}$	30	150
R_γ	50	250
g_ν	11.25-0.03125.A	5.(11.25-0.03125.A)
g_π	11.25-0.03125.A	5.(11.25-0.03125.A)
E_{surf}	20	100
C_{break}	80	400
C_{strip}	80	400
C_{knock}	80	400

into a random ENDF-6 formatted data library which then provides one random sample of an applied calculation with e.g. MCNP. In other words, the Monte Carlo loop is put around both fundamental and applied nuclear calculations. Since its initial application on criticality benchmarks, TMC has been used for many applications for which uncertainty quantification was deemed important.

In practical terms, it means that for TMC, TASMAN does not only need to drive TALYS but also the ENDF-6 formatting code TEFAL. For each random TALYS input file TASMAN lets TALYS calculate its output and subsequently calls TEFAL to produce a random ENDF-6 file. In the same process, TARES may be called for the resonance range and TAFIS and TANES for fission quantities, in the case of actinides.

Sample case 5.4 n + Sn120: Total Monte Carlo

This is a sample case which demonstrates the principle of Total Monte Carlo. It also contains all particular features we use to create covariance files for the TENDL library. With every random sample of parameters, the random ENDF-6 file is retained for TMC uncertainty propagation. Below is the input file for ^{120}Sn.

```
#
# TASMAN input file generated by Autotalys
#
#
# TALYS input file generated by Autotalys
#
projectile n
element    Sn
mass       120
Ltarget    0
```

```
energy      energies
maxlevelstar    30
partable y
bins 40
#
# Use best parameters from database
#
best y
#
# Produce files for processing into ENDF and increase required precision
#
endf y
endfdetail y
popeps 1.e-12
transeps 1.e-20
transpower 15
xseps 1.e-20
#
# Output of extra channels
#
channels y
filechannels y
#
# Recoils
#
recoil y
recoilaverage y
#
# TASMAN
#
bins 40
#change
#liso 0
#seed 120
##
## Covariance keywords
##
#mode 1
#band y
#save y
#offset 0
#gauss n
#discrete y
#average 2
#ntalys 3
```

```
##
## Do not use weights
##
#weight n
##
## Covariance data for isomers
##
#isomer y
##
## Covariance data for residual and production cross sections
##
#residual y
#prod y
##
## Total Monte Carlo keywords
##
#tmc y
#angle y
#legendre y
#resonance y
##tareslib default
#runtares n
##
## Parameters to be varied
##
v1adjust      n   1.00000
v2adjust      n   1.00000
v3adjust      n   1.00000
v4adjust      n   1.00000
rvadjust      n   1.00000
avadjust      n   1.00000
w1adjust      n   1.00000
w2adjust      n   1.00000
w3adjust      n   1.00000
w4adjust      n   1.00000
rwadjust      n   1.00000
awadjust      n   1.00000
rvdadjust     n   1.00000
avdadjust     n   1.00000
d1adjust      n   1.00000
d2adjust      n   1.00000
d3adjust      n   1.00000
rwdadjust     n   1.00000
awdadjust     n   1.00000
rvsoadjust    n   1.00000
```

```
avsoadjust      n   1.00000
vso1adjust      n   1.00000
vso2adjust      n   1.00000
wso1adjust      n   1.00000
wso2adjust      n   1.00000
rcadjust        n   1.00000
Cstrip          n   1.00000
Cknock          n   1.00000
Cbreak          n   1.00000
M2constant          1.00000
Rgamma              2.00000
gamgamadjust        50 121   0.70000
sgradjust           50 121   1.000 E1
egradjust           50 121   1.000 E1
ggradjust           50 121   1.000 E1
s2adjust            50 121   1.00000   0
s2adjust            50 120   1.00000   0
s2adjust            49 120   1.00000   0
s2adjust            49 119   1.00000   0
s2adjust            49 118   1.00000   0
s2adjust            48 118   1.00000   0
s2adjust            48 117   1.00000   0
s2adjust            50 119   1.80000   0
s2adjust            49 117   1.00000   0
s2adjust            48 116   1.00000   0
s2adjust            47 116   1.00000   0
s2adjust            47 115   1.00000   0
s2adjust            47 114   1.00000   0
s2adjust            46 114   1.00000   0
s2adjust            46 113   1.00000   0
s2adjust            48 119   1.00000   0
s2adjust            47 117   1.00000   0
s2adjust            50 118   1.00000   0
s2adjust            49 116   1.00000   0
s2adjust            48 115   1.00000   0
s2adjust            47 113   1.00000   0
s2adjust            46 112   1.00000   0
s2adjust            50 117   1.00000   0
s2adjust            49 115   1.00000   0
s2adjust            48 114   1.00000   0
s2adjust            46 115   1.00000   0
s2adjust            47 118   1.00000   0
s2adjust            46 116   1.00000   0
s2adjust            47 112   1.00000   0
s2adjust            46 111   1.00000   0
```

```
gnadjust          50 121   0.90000
gnadjust          50 120   1.00000
gnadjust          49 120   1.00000
gnadjust          49 119   1.00000
gnadjust          49 118   1.00000
gnadjust          48 118   1.00000
gnadjust          48 117   1.00000
gnadjust          50 119   1.00000
gnadjust          49 117   1.00000
gnadjust          48 116   1.00000
gnadjust          47 116   1.00000
gnadjust          47 115   1.00000
gnadjust          47 114   1.00000
gnadjust          46 114   1.00000
gnadjust          46 113   1.00000
gnadjust          48 119   1.00000
gnadjust          47 117   1.00000
gnadjust          50 118   1.00000
gnadjust          49 116   1.00000
gnadjust          48 115   1.00000
gnadjust          47 113   1.00000
gnadjust          46 112   1.00000
gnadjust          50 117   1.00000
gnadjust          49 115   1.00000
gnadjust          48 114   1.00000
gnadjust          46 115   1.00000
gnadjust          47 118   1.00000
gnadjust          46 116   1.00000
gnadjust          47 112   1.00000
gnadjust          46 111   1.00000
gpadjust          50 121   0.90000
gpadjust          50 120   1.00000
gpadjust          49 120   1.00000
gpadjust          49 119   1.00000
gpadjust          49 118   1.00000
gpadjust          48 118   1.00000
gpadjust          48 117   1.00000
gpadjust          50 119   1.00000
gpadjust          49 117   1.00000
gpadjust          48 116   1.00000
gpadjust          47 116   1.00000
gpadjust          47 115   1.00000
gpadjust          47 114   1.00000
gpadjust          46 114   1.00000
gpadjust          46 113   1.00000
```

```
gpadjust         48 119   1.00000
gpadjust         47 117   1.00000
gpadjust         50 118   1.00000
gpadjust         49 116   1.00000
gpadjust         48 115   1.00000
gpadjust         47 113   1.00000
gpadjust         46 112   1.00000
gpadjust         50 117   1.00000
gpadjust         49 115   1.00000
gpadjust         48 114   1.00000
gpadjust         46 115   1.00000
gpadjust         47 118   1.00000
gpadjust         46 116   1.00000
gpadjust         47 112   1.00000
gpadjust         46 111   1.00000
aadjust          50 120   1.00000
aadjust          49 120   1.00000
aadjust          49 119   1.00000
aadjust          49 118   1.00000
aadjust          48 118   1.00000
aadjust          48 117   1.00000
aadjust          50 119   1.00000
aadjust          49 117   1.00000
aadjust          48 116   1.00000
aadjust          47 116   1.00000
aadjust          47 115   1.00000
aadjust          47 114   1.00000
aadjust          46 114   1.00000
aadjust          46 113   1.00000
aadjust          48 119   1.00000
aadjust          47 117   1.00000
aadjust          50 118   1.00000
aadjust          49 116   1.00000
aadjust          48 115   1.00000
aadjust          47 113   1.00000
aadjust          46 112   1.00000
aadjust          50 117   1.00000
aadjust          49 115   1.00000
aadjust          48 114   1.00000
aadjust          46 115   1.00000
aadjust          47 118   1.00000
aadjust          46 116   1.00000
aadjust          47 112   1.00000
aadjust          46 111   1.00000
v1adjust      p    1.00000
```

```
v2adjust      p    1.00000
v3adjust      p    1.00000
v4adjust      p    1.00000
rvadjust      p    1.06000
avadjust      p    1.06000
w1adjust      p    1.00000
w2adjust      p    1.00000
w3adjust      p    1.00000
w4adjust      p    1.00000
rwadjust      p    1.06000
awadjust      p    1.06000
rvdadjust     p    1.00000
avdadjust     p    1.00000
d1adjust      p    1.00000
d2adjust      p    1.00000
d3adjust      p    1.00000
rwdadjust     p    1.00000
awdadjust     p    1.00000
rvsoadjust    p    1.00000
avsoadjust    p    1.00000
vso1adjust    p    1.00000
vso2adjust    p    1.00000
wso1adjust    p    1.00000
wso2adjust    p    1.00000
rcadjust      p    1.00000
Cstrip        p    1.00000
Cknock        p    1.00000
Cbreak        p    1.00000
v1adjust      a    1.00000
v2adjust      a    1.00000
v3adjust      a    1.00000
v4adjust      a    1.00000
rvadjust      a    1.00000
avadjust      a    1.00000
w1adjust      a    1.00000
w2adjust      a    1.00000
w3adjust      a    1.00000
w4adjust      a    1.00000
rwadjust      a    1.00000
awadjust      a    1.00000
rvdadjust     a    1.00000
avdadjust     a    1.00000
d1adjust      a    1.00000
d2adjust      a    1.00000
d3adjust      a    1.00000
```

```
rwdadjust      a   1.00000
awdadjust      a   1.00000
rvsoadjust     a   1.00000
avsoadjust     a   1.00000
vso1adjust     a   1.00000
vso2adjust     a   1.00000
wso1adjust     a   1.00000
wso2adjust     a   1.00000
rcadjust       a   1.00000
Cstrip         a   1.00000
Cknock         a   1.00000
Cbreak         a   1.00000
v1adjust       d   1.00000
v2adjust       d   1.00000
v3adjust       d   1.00000
v4adjust       d   1.00000
rvadjust       d   1.00000
avadjust       d   1.00000
w1adjust       d   1.00000
w2adjust       d   1.00000
w3adjust       d   1.00000
w4adjust       d   1.00000
rwadjust       d   1.00000
awadjust       d   1.00000
rvdadjust      d   1.00000
avdadjust      d   1.00000
d1adjust       d   1.00000
d2adjust       d   1.00000
d3adjust       d   1.00000
rwdadjust      d   1.00000
awdadjust      d   1.00000
rvsoadjust     d   1.00000
avsoadjust     d   1.00000
vso1adjust     d   1.00000
vso2adjust     d   1.00000
wso1adjust     d   1.00000
wso2adjust     d   1.00000
rcadjust       d   1.00000
Cstrip         d   1.00000
Cknock         d   1.00000
Cbreak         d   1.00000
v1adjust       t   1.00000
v2adjust       t   1.00000
v3adjust       t   1.00000
v4adjust       t   1.00000
```

```
rvadjust       t   1.00000
avadjust       t   1.00000
w1adjust       t   1.00000
w2adjust       t   1.00000
w3adjust       t   1.00000
w4adjust       t   1.00000
rwadjust       t   1.00000
awadjust       t   1.00000
rvdadjust      t   1.00000
avdadjust      t   1.00000
d1adjust       t   1.00000
d2adjust       t   1.00000
d3adjust       t   1.00000
rwdadjust      t   1.00000
awdadjust      t   1.00000
rvsoadjust     t   1.00000
avsoadjust     t   1.00000
vso1adjust     t   1.00000
vso2adjust     t   1.00000
wso1adjust     t   1.00000
wso2adjust     t   1.00000
rcadjust       t   1.00000
Cstrip         t   1.00000
Cknock         t   1.00000
Cbreak         t   1.00000
v1adjust       h   1.00000
v2adjust       h   1.00000
v3adjust       h   1.00000
v4adjust       h   1.00000
rvadjust       h   1.00000
avadjust       h   1.00000
w1adjust       h   1.00000
w2adjust       h   1.00000
w3adjust       h   1.00000
w4adjust       h   1.00000
rwadjust       h   1.00000
awadjust       h   1.00000
rvdadjust      h   1.00000
avdadjust      h   1.00000
d1adjust       h   1.00000
d2adjust       h   1.00000
d3adjust       h   1.00000
rwdadjust      h   1.00000
awdadjust      h   1.00000
rvsoadjust     h   1.00000
```

```
avsoadjust     h    1.00000
vso1adjust     h    1.00000
vso2adjust     h    1.00000
wso1adjust     h    1.00000
wso2adjust     h    1.00000
rcadjust       h    1.00000
Cstrip         h    1.00000
Cknock         h    1.00000
Cbreak         h    1.00000
#weight y
#mcak y
#global y
#deltaE n
#gauss n
#kdburn n
#libinclude irdff2.0 5.
#expinclude all 1.
#talinclude all
#chi2 1
#Fmax 1.2 1
#Fmax 1.4 4
#Fmax 1.4 16
#Fmax 1.4 51
#Fmax 1.8 102
#Fmax 1.8 103
#Fmax 2. 107
#Fmax 2.
#fracmax 0.1
#readpar y
#Ehigh 30.
#nhigh 0
#prod y
#gamma y
#spectra y
```

Besides the uncertainty files mentioned in the previous sample cases, the important covariance information is in the files starting with *var* (variance) and *cov* (covariance). For example, *cov_intra.ave* contains all covariance matrix elements of cross sections correlated within the same reaction channel. These files are read by another software package, TEFAL, for the production of covariance files in nuclear data libraries. ∎

6. Optimization ("Search")

The basic feature of TASMAN is to vary the input parameters of TALYS and to analyze the calculated results. This makes it obvious to include the option of automatic fitting to experimental data and to other evaluated data sets. Note that the latter is directly an important point: if other evaluators have already performed a proper evaluation of a reaction channel, we can regard their result as pseudo-experimental data that we want to optimize the TALYS parameters to. At least the experimental outliers are already removed from the scheme, which is usually a problem for 'blind' optimization to EXFOR.

It is possible to read in experimental and evaluated data, ranging from a single data point to multi-nuclide multi-channel collections of experimental data, and to search for the optimal TALYS parameter set using one of the built-in optimization methods. The notion of the "best fit" depends on the choice of the goodness-of-fit estimator and we will outline those first. Next, we will briefly mention TASMAN's optimization methods, and finally we describe how to perform a search.

6.1 Goodness-of-fit estimators

Goodness-of-fit (GOF) estimators can be used to describe the deviation from one model to another model, from a model to experiment, or for a loss function to be minimized in an optimization problem. Often, one of two GOF estimators, χ^2 or the C/E (calculated/experimental) value is used. Let σ_T^i, σ_E^i, $d\sigma_E^i$ be the theoretical and experimental cross section and experimental uncertainty, respectively for point i out of a total of N data points, then the χ^2 per point for a data set is

$$\chi^2 = \frac{1}{N} \sum_i^N \left(\frac{\sigma_T^i - \sigma_E^i}{d\sigma_E^i} \right)^2, \tag{6.1}$$

which should be generalized, if a covariance matrix is available, to the generalized χ^2 formula. The problem of χ^2 is that different experimental data sets often have to be closely inspected

regarding their uncertainties, otherwise any evaluation or optimization procedure will automatically strongly favor the set with lowest uncertainty. Therefore, in a detailed evaluation procedure, the reported uncertainties often need to be adjusted to come to a reliable generalized χ^2 procedure. In addition, nuclear model calculations may be several standard deviations away from the experimental central values, making the consequences of differences in experimental uncertainties very strong, and resulting in very large χ^2 values, often dominated by the experimental data set with small uncertainties.

The other often used GOF estimator, especially for model calculations, is the C/E value which for a for a single point is simply

$$C/E = \frac{\sigma_T^i}{\sigma_E^i}. \tag{6.2}$$

For a set of data points this can be generalized to the F_{rms}-factor (root-mean-square)

$$F_{\text{rms}} = \exp\left[\frac{1}{N}\sum_i^N \left(\ln \frac{\sigma_T^i}{\sigma_E^i}\right)^2\right]^{1/2}. \tag{6.3}$$

The advantage over χ^2 is that the resulting F_{rms}-value is a direct measure of the relative deviation. A value of F_{rms}=1.2 means that for the entire data set we are roughly 20% off on average from the central values of the experimental data set. The disadvantage is that the experimental uncertainty is not taken into account.

Ideally, one would have an estimator which integrates both the C/E deviation from 1 and the experimental uncertainty in one and the same estimator. To get to our preferred estimator we follow an idea of Goriely et al [28]. Analogous to Eq. (6.3), the relative deviation is defined as

$$F_{rms} = \exp\left[\frac{1}{N}\sum_i^N (\ln r_i)^2\right]^{1/2}, \tag{6.4}$$

and the asymmetry as

$$\varepsilon_{rms} = \exp\left[\frac{1}{N}\sum_i^N \ln r_i\right]. \tag{6.5}$$

Instead of taking $r_i = C/E$ which would return Eq. (6.3), in Ref. [28] they take

$$
\begin{aligned}
r_i &= \frac{\sigma_T^i}{\sigma_E^i - d\sigma_E^i} \quad \text{if} \quad \sigma_T^i < \sigma_E^i - d\sigma_E^i, \\
&= \frac{\sigma_T^i}{\sigma_E^i + d\sigma_E^i} \quad \text{if} \quad \sigma_T^i > \sigma_E^i + d\sigma_E^i, \\
&= 1 \quad \text{otherwise}, \tag{6.6}
\end{aligned}
$$

which basically states that as soon as the theoretical value is inside the 1-sigma experimental uncertainty band, the GOF estimator is 1, and therefore all theoretical values inside $\sigma_E^i \pm d\sigma_E^i$ represent an equally good fit. For practical purposes, this is a powerful recipe: a small experimental uncertainty has a stronger weight than a large uncertainty, though not as strong as in the case of χ^2. The behaviour of this GOF estimator is depicted in Fig. 6.1, where we plot the various deviation estimators for an experimental data point of $\sigma_E^i = 10 \pm 1$ mb, as a function of the theoretical value C (σ_T^i). Note that the asymmetry estimator of Eq. (6.5) also plays an important role in GOF estimates:

F_{rms} may have a value significantly different from 1, but as long as ε_{rms} is close to 1, we can at least expect that there is no bias towards persistent under- or overestimation of the data by our model.

Here, we refine Eq. (6.6) to allow optimization procedures to aim for the real minimum, since for Eq. (6.6) the optimization would stop as soon as the value of $\sigma_E^i \pm d\sigma_E^i$ is reached. For this we look at the probability density function of an experimental data point, which is usually assumed to be a Normal distribution

$$\text{pdf}(\sigma_T^i) = \frac{1}{\sigma_E^i \sqrt{2\pi}} \exp\left[-\frac{1}{2}\left(\frac{\sigma_T^i - \sigma_E^i}{d\sigma_E^i}\right)^2\right]. \tag{6.7}$$

The aim is to reward theoretical values closer to the experimental central value also inside the uncertainty band. Therefore, we use the cumulative density function as a weight for the deviation of the theoretical from the experimental point,

$$\text{cdf}(\sigma_T^i) = \frac{1}{2}\left[1 + \text{erf}(\frac{\sigma_T^i}{\sqrt{2}})\right], \tag{6.8}$$

where *erf* is the error function. The *cdf* represents the probability obtained by integrating the *pdf* over a certain region. When theoretical values are close to the central experimental value, the *cdf* increases relatively quick, while when we are in the tail of the pdf, the *cdf* slowly reaches convergence. If we define

$$x = \frac{\sigma_T^i - \sigma_E^i}{d\sigma_E^i \sqrt{2}}, \tag{6.9}$$

then we postulate that the deviation of theory versus experiment can be expressed as follows

$$
\begin{aligned}
r_i &= 1 - (\frac{C}{E} - 1)\text{erf}(x) \quad \text{if} \quad \sigma_T^i < \sigma_E^i, \\
&= 1 + (\frac{C}{E} - 1)\text{erf}(x) \quad \text{if} \quad \sigma_T^i > \sigma_E^i, \\
&= 1 \quad \text{if} \quad \sigma_T^i = \sigma_E^i.
\end{aligned}
\tag{6.10}
$$

By using the *cdf* as a measure for the deviation, we obtain r_i values in between C/E and the values of Eq. (6.6), see Fig. 6.1. It has the required boundary conditions of $r_i = 1$ when $C/E = 1$ (the fit is 'perfect' regardless of the experimental uncertainty) and $r_i = C/E$ at large deviations between theory and experiment (the deviation is so large that the experimental uncertainty becomes irrelevant).

6.2 Optimization methods

Although visual inspection and common sense of the physicist will remain indispensable to obtain the best evaluated data, part of our quest for the best nuclear model parameters is based on automatic optimization of the goodness-of-fit estimator. We are using an optimization subroutine inside TASMAN, called KOEKEL, which contains 4 different search algorithms, which can be invoked with the **mode** keyword. The current default is **mode 4** (Csendes optimization). This can be changed by creating a file *koekel.inp* in your working directory with e.g. the line **mode 1**.

For search methods which are not completely stochastic (like simulated annealing) it is important to be economical with the number of parameters to be searched, also since the search routines may have an upper limit for the number of parameters to be varied. If a search method is completely stochastic, this number is not relevant: insensitive parameters will also be varied but no time is wasted since each random TALYS run is done with all (sensitive and insensitive) parameters varied, which makes no difference in the calculation time.
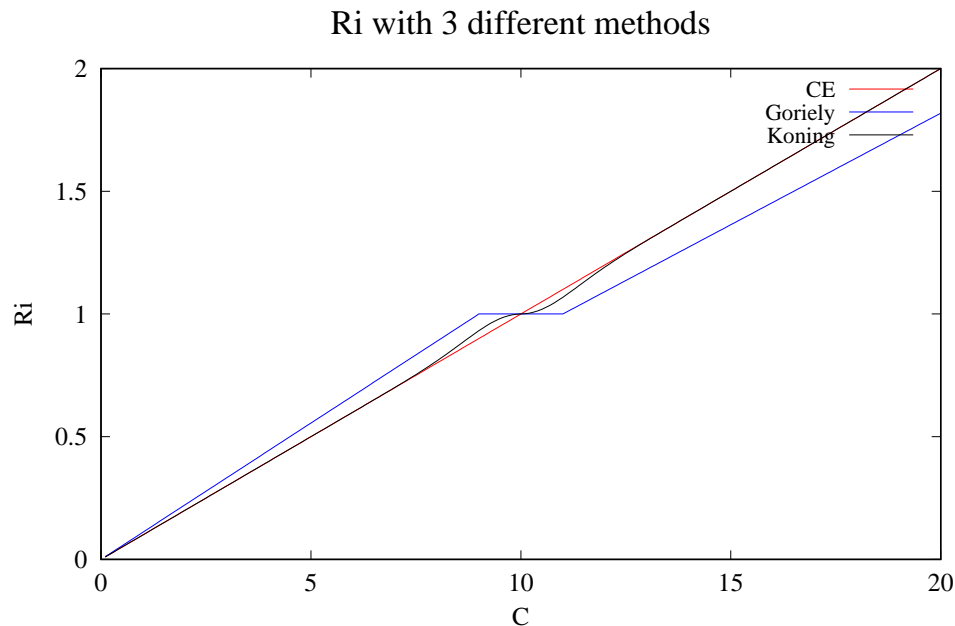
Figure 6.1: Different Ri estimators for an experimental data point of 10 ± 1.

6.2.1 Simulated annealing

One option for optimization utilizes simulated annealing; a global optimization method that distinguishes between different local optima.

In simulated annealing, starting from an initial point (i.e. multi-dimensional parameter guess) and evaluated function, the algorithm makes a change in variable and re-evaluates the function. The algorithm then accepts any downhill step and the process is repeated. An uphill step may be accepted as well, an aspect of the process that enables it to escape from local optima. This uphill decision is made using a Monte Carlo criteria with temperature and the size of the downhill move assessed in a probabilistic manner. The smaller the "temperature" and the size of the "downhill" move, the more likely that move will be accepted. If the trial is accepted, the algorithm moves on from that point. If it is rejected, another point is chosen instead for a trial evaluation. As the optimization process proceeds, the algorithm closes in on the global optimum. Since the algorithm makes very few assumptions regarding the function to be optimized, it is quite robust with respect to non-quadratic surfaces. For the simulated annealing optimization, I have written a stand-alone program called Koekel, which has several possibilities regarding optimization speed, biases, etc. It may be competitive, if not superior, to multiple restarts of conventional optimization routines. For a certain initial temperature, a user-defined number of function evaluations is tried. Then, the temperature is reduced, and starting from the previously found minimum, the search continues. The degree of robustness of the search can be adjusted by the user. The temperature, the number of function trials for each temperature and the temperature reduction factor are the

crucial parameters in a simulated annealing search. It determines the probability that the function to be minimized escapes from a local minimum, and also how quickly a final optimum is reached. We have minimized the total number of trial evaluations by investigating the sensitivity of several nuclear model problems to the parameters of the simulated annealing program. Nevertheless, the part of the optimization that is done by χ^2 typically takes hundreds to thousands TALYS calculations for a typical nucleus. The number of parameters to be varied is unlimited.

6.2.2 Other optimization methods

We have included a few other optimization methods in TASMAN, which sometimes are more efficient than simulated annealing. Since these are rather old Fortran77 programs they are often restricted in the number of parameters that can be optimized simultaneously and hardwired dimensions make it difficult to change that. **mode 2** concerns an implementation of the Simplex function minimisation procedure by Melder-Nead (1965). This method is non-stochastic. Another option **mode 3** concerns an implementation of the Rosenbrock minimisation procedure. This method is also non-stochastic. Finally, there is the method of Csendes which usually gives the best and quickest result . This is an implementation of the Boender-Kan-Timmer-Stougie method. It is however restricted to the optimization of 100 parameters.

Unfortunately, we have not yet found the consistent 'best' answer for which optimization method to use for which problem. We tend to switch between **mode 4** (the default) and **mode 2**.

6.3 Optimization in practice

In principle, the variation methods for uncertainty propagation are also suitable to obtain the optimal set of parameters, i.e. the parameters which lead to calculated results that most closely matches experimental data. TASMAN also contains several search methods to accomplish this, and the main TASMAN keyword for this is **#mode 3**. Various experimental data sets can be included, or excluded. In addition, one can optimize TALYS calculations to evaluated data libraries. I advise to read the first sample case in some detail since it contains a description of where the important output files for the optimization process can be found. Obviously, this output is similar for the other sample cases.

6.3.1 Some sample cases for optimization to experimental data only

Sample case 6.1 **n-Nb093-omp-opt: Optimization of optical model parameters for total cross section**

The sample case below does a simple optical model search, on total cross sections only. The *tasman.inp* file looks as follows

```
#
# General
#
projectile n
element nb
mass 93
energy energies
bins 20
omponly y
```

```
#
# TASMAN keywords start here
#
#change
#mode 3
#ntalys 100
#expinclude 1
##
## Parameters to be varied
##
rvadjust n 1.
avadjust n 1.
v1adjust n 1.
v2adjust n 1.
d1adjust n 1.
d2adjust n 1.
d3adjust n 1.
rwdadjust n 1.
awdadjust n 1.
```

and you can run this after going to */samples/n-Nb093-omp-opt/new/* with *tasman < tasman.inp > tasman.out &*. TASMAN does not have the same flexibilty for the **energy** keyword as TALYS, so you always have to give either 1 incident energy or your own list of incident energies, like in TALYS. (for example, **energy 0.5 20 0.5** does not work for TASMAN.) Note that for this sample case, we stop at a maximum of 100 TALYS runs, and you may need a lot more to reach convergence. With the **#expinclude** keyword we automatically include all experimental data from the EXFOR database for MT1, i.e. total cross sections. We limit the search to 9 different parameters and for this particular sample case we give them explicitly including their starting values, so TASMAN knows how to change them. Each time the search method finds a new optimum, it is mentioned in the main output file *tasman.out*.

The main output file now contains more information. Below the used parameters for the particular random input file there is now a block with a goodness-of-fit summary of the TALYS run, when compared with experimental data,

```
 Comparison of TALYS and experimental data for  Z:  41 A:  93  Run:     0

 Goodness-of-fit summary for Z:  41 A:  93

 Channel          Sets  Points     GOF          Total optimum        Channel optimum
 totalxs.tot        25   866       1.115        1.0000E+38 (   0)    1.0000E+38 (   0)

GOF no.               0   1.115
Number of nuclides     1
Number of channels     1
Number of data sets   25
Number of points      866                  Optimum
Total chi2             271.0
chi2 per point         10.84               1.0000E+38
F per point            1.0239E+06          1.0000E+38
```

```
Total D               2.5596E+07
D per point           1.0239E+06        1.0000E+38
Av. distance           1012.    mb     1.0000E+19 mb
K per point            1.271            1.0000E+38


 New optimum G=    1.11451721
```

In our case, the optimum is found for sample run number 95 which gives
```
 New optimum G=  1.10180080
```

The block above is just a summary, giving the number of sets, and various goodness-of-fit estimators, per channel, and as total. This is given for every random run. Run number 0 is always the one with default parameter values. Note that the default is to use the F-value as the leading goodness-of-fit estimator. This can be changed with the **#chi2** keyword. More extensive information about the goodness-of-fit is given in the *gof.NNNN* files where NNNN denotes the random run. You may want to study e.g. *gof.0001* which gives details about every included experimental data set, and point, and its deviation from the TALYS calculation. At the bottom of the file, the summary for each particular random run is given, just like in the main output file. Note also that inside *gof.NNNN* averages per reaction channel are given. Moreover, when a new optimum is found, the current optimal set of TALYS output files and parameters are copied to a directory *optimum/*. In the process, some linux 'cp' warnings may occur for non-existing files, as I was not able to get rid of them.   ■

   In our quest for more efficiency and automation, we recently added the **#keyvary** keyword to TASMAN, so that the default parameters no longer have to be given in the input as in the sample case above. Instead, we specify *which* parameters need to be varied, to which depth in Z and A away from the initial compound nucleus and for which particles. Several of the following sample cases make use of that feature.

Sample case 6.2 **p + Ag107: Optimization of model parameters for (p,n) cross section**
The sample case includes more nuclear model parameter than just the optical model and is meant for fitting an experimental (p,n) cross section only. The *tasman.inp* file looks as follows

```
#
# TASMAN input file generated by Autotalys
#
#
# TALYS input file generated by Autotalys
#
projectile p
element    Ag
mass       107
Ltarget    0
energy     energies
maxlevelstar    30
partable y
bins 40
equidistant y
```

```
#
# Use best parameters from database
#
best y
#
# Output of extra channels
#
channels y
filechannels y
spherical y
#
# Reduce output for activation-only calculation
#
outspectra n
outangle n
ddxmode 0
outdiscrete n
#
# TASMAN
#
bins 20
#change
#liso 0
#ntalys 10
#seed 107
##
## Search keywords
##
#mode 3
#chi2 2
#save y
#deltaE n
##
## Parameters to be varied
##
#deltaE n
#parameters y
#expinclude 4
#Zdeep 1
#Adeep 1
#partvary p
#partvary n
#fracmax 0.03 4
#keyvary rvadjust
#keyvary avadjust
```

```
#keyvary rwdadjust
#keyvary awdadjust
#keyvary d2adjust
#keyvary w1adjust
#keyvary gpadjust
#keyvary gnadjust
#keyvary M2constant
```

Besides the TALYS input parameters this contains various TASMAN keywords which can be found in this manual. The most essential for this sample case are **expinclude 4** which means optimization to experimental data for MT4, i.e. (p,n) cross sections, An important line is **#parameters y** which means that TASMAN will first do one TALYS run with **partable y**, meaning that when TALYS is finished a file *parameters.dat* has been created with all the nuclear model parameters used in the run. TASMAN then reads this file and uses it in combination with the **#keyvary** keywords which sets the TALYS parametrs to be varied. For this sample case this is only done for nuclides which are Z=1, A=1 deep from the initial compound nucleus (this is for **gnadjust** and **gpadjust**) and for neutrons and protons only (this is for the OMP parameters). The *talys.inp* file this TASMAN input file generates shows all the parameters which are then varied. Also note that this sample cases does not contain **#ntalys** keyword for the number of trials, and it will thus continue until the optimization routine declares convergence. In our sample directory, we interrupted the TASMAN run after a few samples.   ■

Optimization can be done at several levels of efficiency, which at the moment we have not yet automated. As you can see in the sample case above, the number of TALYS **bins** is only 20, which means that this is not the most precise TALYS calculation one can imagine. One may first try to find an optimum as fast as possible, and then fine tune this later using more bins. The same efficiency measures can be reached by varying the number of incident energies for example.

6.3.2 Some sample cases for optimization to evaluated libraries

If important reaction channels for certain nuclides have already been evaluated by others, we can choose to optimize our calculations to that instead of to experimental data. The following cases are examples of that

Sample case 6.3 n + U238: Optimization of fission cross section to ENDF/B-VIII
The *tasman.inp* file looks as follows

```
projectile n
element    u
mass       238
energy    energies
maxrot 4
dispersion y
strength 8
strengthm1 8
upbend y
ldmodel 5
fismodel 5
```

```
fispartdamp y
hbstate n
class2 n
filechannels y
channels y
ftable        92 239     0.63082 e1
etable        92 239    -1.06343 e1
ecissave y
eciscalc y
inccalc y
outdiscrete y
filediscrete 1
filediscrete 2
filediscrete 3
filediscrete 4
v1adjust      n     1.00686
v2adjust      n     0.96942
v3adjust      n     1.05701
rvadjust      n     1.00987
avadjust      n     1.03252
w1adjust      n     0.97443
w2adjust      n     1.17850
d1adjust      n     1.06325
d2adjust      n     1.22364
d3adjust      n     1.39542
rwdadjust     n     1.00953
awdadjust     n     1.01263
ctable        92 239    -0.39753 1
ptable        92 239    -0.03277 1
ctable        92 239     0.66973 2
ptable        92 239    -0.45975 2
ctable        92 239     0.44307 3
ptable        92 239    -0.14079 3
betafiscor    92 239     0.80811
vfiscor       92 239     0.95026
#change
#mode 3
#chi2 2
#ecis n
#save y
#deltaE y
#parameters y
#ntalys 20
##expinclude 18
#libinclude 18 endfb8.0
```

```
#fracmax 0.02
#Zdeep 0
#Adeep 0
#minbar 1
##keyvary bdampadjust
#keyvary ctable
#keyvary ptable
#keyvary vfiscor
#keyvary betafiscor
```

7. Reference Guide

In this part, all TASMAN 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 section, 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.

7.1 General keywords

As explained above, the minimum input file is to take a TALYS input file and to write **#change** at the appropriate place, making the parameter values of the TALYS keywords that follow *below* this line variable for each random run. A minimal specification leaves all choices for optimization or covariance procedures to be made by the author. In general, you probably want to be more specific. In this section, 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.

The highest level of input is formed by the keywords that determine the type of analysis that TASMAN will perform, such as either optimization (search) or uncertainty quantification.

change

Keyword which specifies that the *rest* of the input file concerns TASMAN keywords and is subject to parameter variation by TASMAN. The lines following **#change** do not only concern other specific TASMAN keywords, but also TALYS keywords. If e.g. a numerical value for a TALYS keyword is given below the input line with **#change**, TASMAN knows that the associated parameter should be varied in an uncertainty or optimization procedure. Any keywords given before the line **#change** will be disregarded by TASMAN (i.e. will not be varied), but of course will be accepted by TALYS. In other words, it is also possible to give parameter values which will *not* be varied. Comments inserted after the line with **#change** should start with **##**.

Examples

> **#change**

Range

None

Default

None, without **#change** TASMAN will not work.

mode

Type of analysis that TASMAN will perform. There are 4 options:

Examples

mode 1: Monte Carlo uncertainty, covariance and sensitivity method for BMC, TMC, etc.
mode 2: Sensitivity matrix using linear approximation
mode 3: Optimization ("search") to experimental and evaluated data for one nucleus
mode 4: Optimization ("search") to experimental and evaluated data for many nuclides simultaneously

Range

$1 \leq \textbf{mode} \leq 4$

Default

mode 1

7.2   General keywords for uncertainty and covariance generation

Once **#mode 1** has been chosen, there are various general keywords to steer the Monte Carlo uncertainty quantification. These are outlined below.

Ntalys

Number of random TALYS calculations in the Monte Carlo approach. Generally, a value of around 300 is used, although this number may depend on the number of parameters, the specific problem, and your computer power.

Examples

#Ntalys 1000

Range

#Ntalys $\geq$ 0

Default

#Ntalys 10, which is probably too small for statistically meaningful results, but which is enough to verify that the first samples indeed provide the expected variation.

Nburn

Number of random "burn-in" calculations in the Monte Carlo approach. This will let TASMAN first generate **#Nburn** random input files with parameters drawn from a uniform (i.e. non-informative) or other initial (e.g. Gaussian) probability distribution. After **#Nburn** runs, parameters can then be drawn from the probability distribution built up so far.

Examples

   #Nburn 500

Range

#Ntalys $\geq$ 0

Default

#Nburn = #Ntalys, i.e. all random samples are drawn from the initial probability distribution.

Comments

This keyword is not needed if one draws from an initial numerical probability distribution, as stored in e.g. the *talys/structure/best/Ni058/parameters/* directory.

save

Flag to save all input and output files of each individual TALYS run. This may be helpful for extra statistical analysis, or to test which TALYS input file caused a certain result. Obviously, this requires a lot of memory storage.

Examples

> **#save n**
> **#save y**

Range

y or **n**

Default

#save n

band

Flag to present the final uncertainties in cross section files as absolute error bands instead of relative errors. Hence, if the final cross section at 1 MeV is 100 mb $\pm$ 10 mb, **#band n** will produce a file with 1. 100. 10., while for **#band y** this will be 1. 100. 90. 110., enabling to plot the uncertainty band.

Examples

> **#band n**
> **#band y**

Range

y or **n**

Default

#band n

gauss

Flag to sample parameters from a Gaussian distribution instead of a uniform distribution.

Examples

#gauss n
#gauss y

Range

y or **n**

Default

#gauss y

metropolis

Flag for Metropolis sampling.

Examples

> **#metropolis n**
> **#metropolis y**

Range

y or **n**

Default

#metropolis n

mcak

Flag to use sampling from arbitrary distribution.

Examples

#mcak n
#mcak y

Range

y or **n**

Default

#mcak n

Comments

In a subdirectory of TASMAN, numerical parameter distributions are stored in e.g. *tasman/parameters/Zr090*. These have been produced in 2013 with TALYS-1.4 and the version of EXFOR which was available at the time. I guess mcak stands for Monte Carlo Arjan Koning.

kdburn

Flag to use KD03 OMP function for burn-in.

Examples

#kdburn n
#kdburn y

Range

y or **n**

Default

#kdburn y

dev

Global deviation, as a fraction of 1, of random TALYS results from the central value cross section. On the input line, we read **#dev**, MT number and value. See table 7.1 for the correspondence of MT numbers with reaction channels.

Examples

#dev 18 0.05 (MT18 = fission cross section)

Range

#dev > 0.

Default

#dev 1 0.05
#dev 2 0.15
#dev 3 0.15
#dev 4 0.20
#dev 16 0.15
#dev 17 0.30
#dev 18 0.10
#dev 22 0.30
#dev 28 0.30
#dev 37 0.40
#dev 102 0.50
#dev 103 0.40
#dev 104 0.70
#dev 107 0.60
#dev [all other MT's] 1000.

Table 7.1: The ENDF-6 MT numbers that can be considered in TASMAN and the corresponding TALYS output files.

MT	Reaction	MF3 (TALYS)	MF4,6 (TALYS)
1=2+3	total	totalxs.tot	
2	elastic	elastic.tot	nnxxx.xxxleg.L00
3=4+5+[11-18]+[22-37] +[41-45]+[102-117]	non-elastic	nonelastic.tot	
4=[51-90]+91	total (n,n')	xs100000.tot	
5	(n,x)		
11	(n,2nd)	xs201000.tot	sp201000Exxxx.xxx.tot
16	(n,2n)	xs200000.tot	sp200000Exxxx.xxx.tot
17	(n,3n)	xs300000.tot	sp300000Exxxx.xxx.tot
18=[19-21]+38	total (n,f)	fission.tot	
19	1st-chance (n,f)	xs000000.fis	
20	2nd-chance (n,f)	xs100000.fis	
21	3rd-chance (n,f)	xs200000.fis	
22	(n,nα)	xs100001.tot	sp100001Exxxx.xxx.tot
23	(n,n3α)	xs100003.tot	sp100003Exxxx.xxx.tot
24	(n,2nα)	xs200001.tot	sp200001Exxxx.xxx.tot
25	(n,3nα)	xs300001.tot	sp300001Exxxx.xxx.tot
28	(n,np)	xs110000.tot	sp110000Exxxx.xxx.tot
29	(n,n2α)	xs100002.tot	sp100002Exxxx.xxx.tot
30	(n,2n2α)	xs200002.tot	sp200002Exxxx.xxx.tot
32	(n,nd)	xs101000.tot	sp101000Exxxx.xxx.tot
33	(n,nt)	xs100100.tot	sp100100Exxxx.xxx.to
34	(n,nh)	xs100010.tot	sp100010Exxxx.xxx.tot
35	(n,nd2α)	xs101002.tot	sp101002Exxxx.xxx.tot
36	(n,nt2α)	xs100102.tot	sp100102Exxxx.xxx.tot
37	(n,4n)	xs400000.tot	sp400000Exxxx.xxx.tot
38	4th-chance (n,f)	xs300000.fis	
41	(n,2np)	xs210000.tot	sp210000Exxxx.xxx.tot
42	(n,3np)	xs310000.tot	sp310000Exxxx.xxx.tot
44	(n,n2p)	xs120000.tot	sp120000Exxxx.xxx.tot
45	(n,npα)	xs110001.tot	sp110001Exxxx.xxx.tot
51-90	(n,n'$_1$) - (n,n'$_{40}$)	nn.L01-nn.L40	nnxxxx.xxxleg[ang].L01-L40
91	continuum (n,n')	nn.con	sp100000Exxxx.xxx.tot
102	(n,γ)	xs000000.tot	sp000000Exxxx.xxx.tot
103=[600-649]	(n,p)	xs010000.tot	sp010000Exxxx.xxx.tot
104=[650-699]	(n,d)	xs001000.tot	sp001000Exxxx.xxx.tot

Table 7.2: The ENDF-6 MT numbers that can be considered in TASMAN and the corresponding TALYS output files (continued).

MT	Reaction	MF3 (TALYS)	MF4,6 (TALYS)
105=[700-749]	(n,t)	xs000100.tot	sp000100Exxxx.xxx.tot
106=[750-799]	(n,h)	xs000010.tot	sp000010Exxxx.xxx.tot
107=[800-849]	(n,α)	xs000001.tot	sp000001Exxxx.xxx.tot
108	(n,2α)	xs000002.tot	sp000002Exxxx.xxx.tot
109	(n,3α)	xs000003.tot	sp000003Exxxx.xxx.tot
111	(n,2p)	xs020000.tot	sp020000Exxxx.xxx.tot
112	(n,pα)	xs010001.tot	sp010001Exxxx.xxx.tot
113	(n,t2α)	xs000102.tot	sp000102Exxxx.xxx.tot
114	(n,d2α)	xs001002.tot	sp001002Exxxx.xxx.tot
115	(n,pd)	xs011000.tot	sp011000Exxxx.xxx.tot
116	(n,pt)	xs010100.tot	sp010100Exxxx.xxx.tot
117	(n,dα)	xs001001.tot	sp001001Exxxx.xxx.tot
152	(n,5n)	xs500000.tot	sp500000Exxxx.xxx.tot
153	(n,6n)	xs600000.tot	sp600000Exxxx.xxx.tot
154	(n,2nt)	xs200100.tot	sp200100Exxxx.xxx.tot
155	(n,tα)	xs000101.tot	sp000101Exxxx.xxx.tot
156	(n,4np)	xs410000.tot	sp410000Exxxx.xxx.tot
157	(n,3nd)	xs301000.tot	sp301000Exxxx.xxx.tot
158	(n,ndα)	xs101001.tot	sp101001Exxxx.xxx.tot
159	(n,2npα)	xs210001.tot	sp210001Exxxx.xxx.tot
160	(n,7n)	xs700000.tot	sp700000Exxxx.xxx.tot
161	(n,8n)	xs800000.tot	sp800000Exxxx.xxx.tot
162	(n,5np)	xs510000.tot	sp510000Exxxx.xxx.tot
163	(n,6np)	xs610000.tot	sp610000Exxxx.xxx.tot
164	(n,7np)	xs710000.tot	sp710000Exxxx.xxx.tot
165	(n,4nα)	xs400001.tot	sp400001Exxxx.xxx.tot
166	(n,5nα)	xs500001.tot	sp500001Exxxx.xxx.tot
167	(n,6nα)	xs600001.tot	sp600001Exxxx.xxx.tot
168	(n,7nα)	xs700001.tot	sp700001Exxxx.xxx.tot
169	(n,4nd)	xs401000.tot	sp401000Exxxx.xxx.tot
170	(n,5nd)	xs501000.tot	sp501000Exxxx.xxx.tot
171	(n,6nd)	xs601000.tot	sp601000Exxxx.xxx.tot
172	(n,3nt)	xs300100.tot	sp300100Exxxx.xxx.tot
173	(n,4nt)	xs400100.tot	sp400100Exxxx.xxx.tot
174	(n,5nt)	xs500100.tot	sp500100Exxxx.xxx.tot
175	(n,6nt)	xs600100.tot	sp600100Exxxx.xxx.tot

Table 7.3: The ENDF-6 MT numbers that can be considered in TASMAN and the corresponding TALYS output files (continued).

MT	Reaction	MF3 (TALYS)	MF4,6 (TALYS)
176	(n,2nh)	xs200010.tot	sp200010Exxxx.xxx.tot
177	(n,3nh)	xs300010.tot	sp300010Exxxx.xxx.tot
178	(n,4nh)	xs400010.tot	sp400010Exxxx.xxx.tot
179	(n,3n2p)	xs320000.tot	sp320000Exxxx.xxx.tot
180	$(n,3n2\alpha)$	xs300002.tot	sp300002Exxxx.xxx.tot
181	$(n,3np\alpha)$	xs310001.tot	sp310001Exxxx.xxx.tot
182	(n,dt)	xs001100.tot	sp001100Exxxx.xxx.tot
183	(n,npd)	xs111000.tot	sp111000Exxxx.xxx.tot
184	(n,npt)	xs110100.tot	sp110100Exxxx.xxx.tot
185	(n,ndt)	xs101100.tot	sp101100Exxxx.xxx.tot
186	(n,nph)	xs110010.tot	sp110010Exxxx.xxx.tot
187	(n,ndh)	xs101010.tot	sp101010Exxxx.xxx.tot
188	(n,nth)	xs100110.tot	sp100110Exxxx.xxx.tot
189	$(n,nh\alpha)$	xs100011.tot	sp100011Exxxx.xxx.tot
190	(n,2n2p)	xs220000.tot	sp220000Exxxx.xxx.tot
191	(n,ph)	xs010010.tot	sp010010Exxxx.xxx.tot
192	(n,dh)	xs001010.tot	sp001010Exxxx.xxx.tot
193	$(n,h\alpha)$	xs000011.tot	sp000011Exxxx.xxx.tot
194	(n,4n2p)	xs420000.tot	sp420000Exxxx.xxx.tot
195	$(n,4n2\alpha)$	xs400002.tot	sp400002Exxxx.xxx.tot
196	$(n,4np\alpha)$	xs410001.tot	sp410001Exxxx.xxx.tot
197	(n,3p)	xs030000.tot	sp030000Exxxx.xxx.tot
198	(n,3p)	xs130000.tot	sp130000Exxxx.xxx.tot
199	$(n,3n2p\alpha)$	xs320001.tot	sp320001Exxxx.xxx.tot
200	(n,5n2p)	xs520000.tot	sp520000Exxxx.xxx.tot
600-640	(n,p_0) - (n,p_{40})	np.L00-np.L40	npxxxx.xxxleg[ang].L01-L40
649	continuum (n,p)	np.con	sp010000Exxxx.xxx.tot
650-690	(n,d_0) - (n,d_{40})	nd.L00-nd.L40	ndxxxx.xxxleg[ang].L01-L40
699	continuum (n,d)	nd.con	sp001000Exxxx.xxx.tot
700-740	(n,t_0) - (n,t_{40})	nt.L00-nt.L40	ntxxxx.xxxleg[ang].L01-L40
749	continuum (n,t)	nt.con	sp000100Exxxx.xxx.tot
750-790	(n,h_0) - (n,h_{40})	nh.L00-nh.L40	nhxxxx.xxxleg[ang].L01-L40
799	continuum (n,h)	nh.con	sp000010Exxxx.xxx.tot
800-840	(n,α_0) - (n,α_{40})	na.L00-na.L40	naxxxx.xxxleg[ang].L01-L40
849	continuum (n,α)	na.con	sp000001Exxxx.xxx.tot

source

The source of the calculations. obviously this is TALYS, but it may be helpful to specify the version such that it appears in the metadata of the many output files. You may edit the 'source' variable variable in input2.f90 and put your own version there. However, you can also use the input keyword.

Examples

#source TALYS-2.01beta3v20240223

Range

Any string

Default

TALYS-2.0

user

The current user who produces the result. You may edit the 'user' variable in input2.f90 and put your own name there. It will appear in the metadata of the various output files. However, you can also use the input keyword.

Examples

> **#user Vlad Avrigeanu**

Range

Any string

Default

Arjan Koning

format

The format of the metadata in the output files. It may be helpful to specify the version. You may edit the 'format' variable variable in input2.f90 and put your own version there. However, you can also use the input keyword.

Examples

#format YANDF-0.1beta2

Range

Any string

Default

YANDF-0.1

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*.

7.3 Uncertainty keywords related to ENDF libraries

All keywords in this subsection are only active if **#tmc y** and thus only relevant for Total Monte Carlo, i.e. to produce an ENDF-6 file at each random run.

tmc

Flag for the creation of a random ENDF-6 data file after each random TALYS run (Total Monte Carlo).

Examples

#tmc n
#tmc y

Range

y or **n**

Default

#tmc n

gpf

Flag for the creation of a complete, random ENDF-6 general purpose file.

Examples

#gpf n
#gpf y

Range

y or **n**

Default

#gpf y

Comments

This keyword is only relevant if **#tmc y**.

s20

Flag for the creation of a random ENDF-6 general purpose file with a switch to MT5 representation at 20 MeV.

Examples

#s20 n
#s20 y

Range

y or **n**

Default

#s20 n

Comments

This keyword is only relevant if **#tmc y**.

s30

Flag for the creation of a random ENDF-6 general purpose file with a switch to MT5 representation at 30 MeV.

Examples

> **#s30 n**
> **#s30 y**

Range

y or **n**

Default

#s30 n

Comments

This keyword is only relevant if **#tmc y**.

s60

Flag for the creation of a random ENDF-6 general purpose file with a switch to MT5 representation at 60 MeV.

Examples

#s60 n
#s60 y

Range

y or **n**

Default

#s60 n

Comments

This keyword is only relevant if **#tmc y**.

defonly

Flag for the creation of only a default ENDF-6 general purpose file.

Examples

#defonly n
#defonly y

Range

y or **n**

Default

#defonly y

Comments

This keyword is only relevant if **#tmc y**. Enabling **#defonly** means that s0, s60 etc. files are not created.

mt

Flag for the creation of a random ENDF-6 general purpose file without a sudden switch to MT5 representation.

Examples

#mt n
#mt y

Range

y or **n**

Default

#mt n

Comments

This keyword is only relevant if **#tmc y**.

acf

Flag for the creation of a random ENDF-6 activation file.

Examples

#acf n
#acf y

Range

y or **n**

Default

#acf n

Comments

This keyword is only relevant if **#tmc y**.

eaf

Flag for the creation of a random EAF activation file.

Examples

> **#eaf n**
> **#eaf y**

Range

y or **n**

Default

#eaf n

Comments

This keyword is only relevant if **#tmc y**.

liso

Number of the isomeric state. This makes sure the proper filename is given to an ENDF-6 file.

Examples

#liso 1 ("m")
#liso 2 ("n")

Range

$0 \leq$ **liso** ≤ 2

Default

liso 0

Nhigh

Number of random high energy runs before TASMAN proceeds with low-energy runs only. This keyword is used in combination with **#Ehigh**, and is meant to reduce the computational time required for the production of a converged covariance matrix.

For TENDL, **#Nhigh 0** is used.

Examples

#Nhigh 10

Range

$0 \leq$ **#Nhigh** $\leq$ **#Ntalys**.

Default

#Nhigh = #Ntalys, i.e. all random runs are performed for all energies.

Ehigh

Energy cutoff of high energy runs. This keyword is used in combination with **#Nhigh**. Below **#Ehigh** the results are based on more random runs than above **#Ehigh**. This number is determined by **#Nhigh**.

For TENDL, **#Ehigh 30.** is used.

Examples

#Ehigh 30.

Range

1. $\leq$ **#Ehigh** $\leq$ **1000.**

Default

#Ehigh 1000.

select

Flag for randomizing only specific parts of an ENDF-6 general purpose file, so one can perform a sensitivity study on the type of nuclear data (per MF, or per MF/MT). The rest of the data file will contain the results of the first, unperturbed run.

Examples

      #select n
      #select y

Range

y or **n**

Default

#select n

Comments

This keyword is only relevant if **#tmc y**.

tmcoffset

Offset for starting creation of ENDF-6 files. It means that all runs before **#tmcoffset** is reached are just to build up statistical convergence.

Examples

#tmcoffset 10

Range

$0 \leq$ **#tmcoffset** $\leq$ **#Ntalys**.

Default

#tmcoffset 0.

Comments

This keyword is only relevant if **#tmc y**.

covrand

Flag to update the ENDF-6 covariance files (MF31,32 etc.) *during* the Monte Carlo process, so one can monitor the convergence of the covariance matrix.

Examples

> **#covrand n**
> **#covrand y**

Range

y or **n**

Default

#covrand n

Comments

This keyword is only relevant if **#tmc y**.

weight

Flag to use weights for random samples. This means including the likelihood for Bayesian update.

Examples

> **#weight n**
> **#weight y**

Range

y or **n**

Default

#weight y

Eweight

Flag to use energy-dependent weights.

Examples

#Eweight n
#Eweight y

Range

y or **n**

Default

#Eweight n

resonance

Flag for variation of resonance parameters for an ENDF-6 general purpose file, by making a system call to the resonance code TARES, or by reading the random resonance parameter database.

Examples

> **#resonance n**
> **#resonance y**

Range

y or **n**

Default

#resonance y

Comments

This keyword is only relevant if **#tmc y**.

runtares

Flag to produce resonance parameters for an ENDF-6 general purpose file in TASMAN with a system call to TARES.

Examples

#runtares n
#runtares y

Range

y or **n**

Default

#runtares n, i.e. values are read from *resbase/*, a pre-produced database.

Comments

This keyword is only relevant if **#tmc y**.

tareslib

Library name for adoption of resonance parameters. The default is to take them from the built-in database of resonances of TARES, but also the three main libraries, **endfb8.0**, **jeff3.3**, **jendl4.0**, can be entered.

Examples

#tareslib endfb8.0

Range

endfb8.0, jeff3.3, jendl4.0 or blank (Atlas).

Default

blank (not given) (i.e. use the built-in database)

Comments

This keyword is only relevant if **#tmc y**.

background

Name of the file to be adopted for background cross sections in MF3/MT1,2,3,18,102, in case there is a background cross section to be added to the resonance parameters of MF2. This is a TEFAL keyword that can be transferred to a TMC run.

Examples

   #background n-Pu239.endfb8.0

Range

The **#background** file should exist in either your working directory or in the *libraries/* database.

Default

If the file of **adopt 2 151** comes from one of the libraries in the TEFAL database, e.g. *n-Pu239.endfb8.0*, that same file is taken for **#background** as default, and the keyword in the example above is not necessary. In other cases **#background** does not have a default.

Comments

This keyword is only relevant if **#tmc y**.

getcalcscript

Script to read in pre-calculated results.

Examples

#getcalcscript talysrun.sh

Range

The script should exist.

Default

#getcalcscript none

lrf7

Flag to use LRF7 for resonance range.

Examples

#lrf7 n
#lrf7 y

Range

y or **n**

Default

#lrf7 n

Comments

This keyword is only relevant if **#tmc y**.

nubar

Flag for variation of the average number of fission neutrons for an ENDF-6 general purpose file.

Examples

#nubar n
#nubar y

Range

y or **n**

Default

#nubar n

Comments

This keyword is only relevant if **#tmc y**.

runnubar

Flag to produce average number of fission neutrons for ENDF-6 general purpose file in TASMAN with a system call to TAFIS.

Examples

> **#runnubar n**
> **#runnubar y**

Range

y or **n**

Default

#runnubar n, i.e. values are read from *nubarbase/*, a pre-produced database.

Comments

This keyword is only relevant if **#tmc y**.

tafislib

Library name for adoption of the average number of fission neutrons. The default is not to specify it and to let it be determined by the TEFAL code. Also the libraries **endfb8.0**, **jeff3.3** and **jendl4.0** can be entered.

Examples

> **#tafislib endfb8.0**

Range

endfb8.0, **jeff3.3**, **jendl4.0** or none (=blank)

Default

#tafislib none

Comments

This keyword is only relevant if **#tmc y**.

fns

Flag for variation of the fission neutron spectrum for an ENDF-6 general purpose file.

Examples

#fns n
#fns y

Range

y or **n**

Default

#fns n

Comments

This keyword is only relevant if **#tmc y**.

runfns

Flag to produce fission neutron spectrum for an ENDF-6 general purpose file in TASMAN with a system call to TANES.

Examples

#runfns n
#runfns y

Range

y or **n**

Default

#runfns n, i.e. values are read from *fnsbase/*, a pre-produced database.

Comments

This keyword is only relevant if **#tmc y**.

taneslib

Library name for adoption of the fission neutron spectrum. The default is not to specify it and to let it be determined by the TEFAL code. Also the libraries **endfb8.0**, **jeff3.3**, and **jendl4.0** can be entered.

Examples

    #taneslib endfb8.0

Range

endfb8.0, **jeff3.3**, **jendl4.0** or none (=blank)

Default

#taneslib none

Comments

This keyword is only relevant if **#tmc y**.

prepro

Flag to run PREPRO codes for pointwise cross section construction for random activation files.

Examples

#prepro n
#prepro y

Range

y or **n**

Default

#prepro y

Comments

This keyword is only relevant if **#tmc y**.

njoy

Flag to run NJOY in case of TMC.

Examples

> **#njoy n**
> **#njoy y**

Range

y or **n**

Default

#njoy y

Comments

This keyword is only relevant if **#tmc y**.

purr

Flag to use PURR in NJOY.

Examples

#purr n
#purr y

Range

y or **n**

Default

#purr n

Comments

This keyword is only relevant if **#tmc y**.

proconly

Flag for processing of ENDF default file only in case of TMC.

Examples

#proconly n
#proconly y

Range

y or **n**

Default

#proconly y

Comments

This keyword is only relevant if **#tmc y**.

extparvar

Flag to include external parameter file.

Examples

#extparvar y
#extparvar n

Range

y or n

Default

#extparvar n

7.4 Keywords for optimization ("search") to experimental and evaluated data

The following keywords are only relevant for **#mode 3** and **4**.

searchmode

Search mode for optimization of parameters.

Examples

searchmode 1: Simulated annealing
searchmode 2: Nelder-Mead
searchmode 3: Rosenbrock
searchmode 4: Csendes

Range

$1 \leq$ **searchmode** ≤ 4

Default

searchmode 4

error

Flag for inclusion of experimental uncertainties as given by EXFOR. If disabled, a constant uncertainty of 10 % will be assumed.

Examples

#error n
#error y

Range

y or **n**

Default

#error y

errlim

Lower limit for experimental error. Values below **errlim** will be set equal to **errlim**.

Examples

#errlim 0.03

Range

$0 \leq$ **errlim** $\leq$ **1.e10**

Default

#errlim 0.03.

Edependence

Flag to include energy-dependent uncertainties from global TALYS calculations, according to Eq. 5.19.

Examples

    #Edependence n
    #Edependence y

Range

y or **n**

Default

#Edependence y

adevi

Parameter for energy dependence of global uncertainty see Eq. 5.19. On the input line, we read **#adevi**, MT number, value

Examples

#adevi 107 1.3

Range

$0 \leq$ **adevi**

Default

#adevi 0.6 for MT1,2,3,102, 1 for other channels.

bdevi

Parameter for energy dependence of global uncertainty see Eq. 5.19. On the input line, we read **#bdevi**, MT number, value

Examples

#bdevi 103 1.3

Range

$0 \leq$ **bdevi**

Default

#bdevi 0.3 for MT1,2,3,102, 1 for other channels.

cdevi

Parameter for energy dependence of global uncertainty see Eq. 5.19. On the input line, we read **#cdevi**, MT number, value

Examples

#cdevi 16 1.3

Range

$0 \leq$ **cdevi**

Default

#cdevi 0. for MT1,2,3, 2 for MT104,105,106, 1 for other channels.

ddevi

Parameter for energy dependence of global uncertainty see Eq. 5.19. On the input line, we read **#ddevi**, MT number, value

Examples

   #ddevi 22 1.3

Range

$0 \leq$ ddevi

Default

#ddevi 15. for MT16, 20 for MT102, 12 for other channels.

Ecent

Parameter for energy dependence of global uncertainty see Eq. 5.19. On the input line, we read **#Ecent**, MT number, value

Examples

#Ecent 54 1.3

Range

0 ≤ Ecent

Default

#Ecent 5. for MT4, 3 for MT51-91, 20 for MT102, 6 for other channels.

chi2

Parameter to set the goodness-of-fit estimator for optimization to experimental data, see Chapter 6 for their definition.

Examples

#chi2 1 (χ^2)
#chi2 2 (F)
#chi2 3 (R)
#chi2 4 (F.D^{power})

Range

$1 \leq$ **chi2** ≤ 4

Default

#chi2 2 (F)

equisample

Flag to sample with equal frequency from all parameter bins. This ensures that the parameter space is sampled somewhat faster than with "pure" random sampling.

Examples

> **#equisample n**
> **#equisample y**

Range

y or **n**

Default

#equisample y

deltaE

Flag to use the dE of the experimental energy grid as a weight for the goodness-of-fit estimator . As available experimental data tend to be unequally distributed over energy, enabling this flag means that we first average them over energy groups to obtain a more realistic goodness-of-fit estimator.

Examples

> **#deltaE n**
> **#deltaE y**

Range

y or **n**

Default

#deltaE n

dexp

Flag to include experimental uncertainty in C/E.

Examples

#dexp y
#dexp n

Range

y or n

Default

#dexp y

Esearch

Energy range, given as start and end energy, for optimization to experimental data. This can be used to constrain the search in a selected energy region.

Examples

#Esearch 13. 15. (search only for energies between 13. and 15. MeV)

Range

0. $\leq$ **#Esearch** $\leq$ **1000.**

Default

#Esearch 0. 1000. (all energies)

Esearch1

Start energy for optimization to experimental data

Examples

#Esearch1 13.

Range

0. $\leq$ #Esearch1 $\leq$ 1000.

Default

not used

Esearch2

End energy for optimization to experimental data

Examples

    #Esearch2 33.

Range

0. $\leq$ #Esearch2 $\leq$ 1000.

Default

not used

fracmax

Fraction of maximum cross section to be included in the optimization. On the input line, we read **#fracmax**, MT number, and value.

Examples

#fracmax 103 0.05

Range

0. $\leq$ **#fracmax** $\leq$ **1.**

Default

#fracmax 0.1

Efracmax

Starting energy for the search for the maximum cross section, for the determination of energy-dependent variation. On the input line, we read **#Efracmax**, MT number, and value.

Examples

#Efracmax 103 0.03

Range

0. $\leq$ #Efracmax $\leq$ 10.

Default

#Efracmax 0.1

libinclude

Flag to include nuclear data from an existing library in the process. This keyword is very flexible. You may include a library for all channels, per channel, or all libraries per channel or for all channels. Finally, a weight can be given for the importance in the optimization process. Note that as soon as **#libinclude** is used with an MT number and/or library name, all other MT numbers or libraries are automatically excluded. On the input line, we read **#libinclude**, optionally an MT number, optionally the nuclear data library, and optionally the weight.

Examples

> **#libinclude 102 all**
> **#libinclude 1 endfb8.0 1.5**
> **#libinclude jendl4.0**

Range

#libinclude can be used for all MT numbers and libraries equal to **endfb8.0, jendl4.0, jeff3.3, cendl3.1, irdff2.0, eaf.2010, tendl.2019**. The weight should be between 0. and 1.e10.

Default

Not used.

libexclude

Flag to exclude nuclear data from an existing library in the process. This keyword is very flexible. You may exclude a library for all channels, per channel, or all libraries per channel or for all channels. Note that as soon as **#libexclude** is used with an MT number and/or library name, all other MT numbers or libraries are automatically included. On the input line, we read **#libexclude**, optionally an MT number, and optionally the nuclear data library.

Examples

> **#libexclude 102 all**
> **#libexclude 1 endfb8.0**
> **#libexclude jendl4.0**

Range

#libexclude can be used for all MT numbers and libraries equal to **endfb8.0, jendl4.0, jeff3.3, cendl3.1, irdff2.0, eaf.2010, tendl.2019**.

Default

Not used.

Fmax

Maximum value per point for goodness-of-fit estimator F. This means that outliers can be disregarded in the analysis.

Examples

#Fmax 2.5

Range

1. $\leq$ #Fmax $\leq$ 1.e38

Default

#Fmax 1.e38 i.e. no points are discarded.

talinclude

Flag to include nuclear data from a default TALYS calculation in the process. This keyword is very flexible. You may include data for all channels, or per channel. Finally, a weight can be given for the importance in the optimization process. Note that as soon as **#talinclude** is used with an MT number, all other MT numbers or libraries are automatically excluded. On the input line, we read **#talinclude**, optionally an MT number, and optionally the weight.

Examples

> **#talinclude all**
> **#talinclude 1 1.5**

Range

#talinclude can be used for all MT numbers. The weight should be between 0. and 1.e10.

Default

Not used.

talexclude

Flag to exclude nuclear data from a default TALYS calculation in the process. This keyword is very flexible. You may exclude data for all channels, or per channel. Note that as soon as **#talexclude** is used with an MT number, all other MT numbers or libraries are automatically included. On the input line, we read **#talexclude**, and an MT number.

Examples

> **#talexclude all**
> **#talexclude 1**

Range

#talexclude can be used for all MT numbers.

Default

Not used.

expinclude

Flag to include experimental data from EXFOR in the process. This keyword is very flexible. You may include a data set for all channels, per channel, or all data sets per channel or for all channels. Finally, a weight can be given for the importance in the optimization process. Note that as soon as **#expinclude** is used with an MT number and/or data set, all other MT numbers or libraries are automatically excluded. On the input line, we read **#expinclude**, optionally an MT number, optionally the data set, and optionally the weight. The experimental data set can be either given by author name or by subentry number.

Examples

#expinclude 102 all
#expinclude 103 Nemilov1973.tot
#expinclude 103 40211002

Range

#expinclude can be used for all MT numbers and filenames as present in the *newbase* database. The weight should be between 0. and 1.e10.

Default

Not used.

expexclude

Flag to exclude experimental data from EXFOR in the process. This keyword is very flexible. You may exclude a data set for all channels, per channel, or all data sets per channel or for all channels. Note that as soon as **#expexclude** is used with an MT number and/or data set, all other MT numbers or libraries are automatically included. On the input line, we read **#expexclude**, optionally an MT number, optionally the data set. The experimental data set can be either given by author name or by subentry number.

Examples

> **#expexclude 102 all**
> **#expexclude 103 Nemilov1973.tot**
> **#expexclude 103 40211002**

Range

#expexclude can be used for all MT numbers and filenames as present in the *newbase* database.

Default

Not used.

chi2max

Maximum value per point for goodness-of-fit estimator χ^2. This means that outliers can be disregarded in the analysis.

Examples

> **#chi2max 2.5**

Range

1. $\leq$ **#chi2max** $\leq$ **1.e38**

Default

#chi2max 1.e38 i.e. no points are discarded.

maxexpsets

Number of experimental data sets included before weight reduction. This avoids too much weight on a reaction channel that is measured many times.

Examples

#maxexpsets 10

Range

$1 \leq$ **#maxexpsets** $\leq$ **1000000**

Default

#maxexpsets 1000000

expclass

Quality class of experimental data to be included in the search.

Examples

> **#expclass 1** (only high-quality data)
> **#expclass 2** (only high- and medium quality data)
> **#expclass 3** (all data)

Range

$1 \leq$ **#expclass** ≤ 3

Default

#expclass 3 (all data)

outsearch

Level of detail for which the output of the search results is given, ranging from everything (5) to very compact (1).

Examples

#outsearch 2

Range

$1 \leq$ **#outsearch** ≤ 5

Default

#outsearch 5 (all data)

gofmode

Goodness-of-fit weight formula for likelihood function.

Examples

#gofmode 1: $w = exp(-chi2point/2)/w0$
#gofmode 2: $w = exp(-(chi2point/chi2point0)/2)/w0$
#gofmode 3: $w = exp(-(chi2point/chi2point0) ** 2)/w0$
#gofmode 4: $w = exp(-chi2/2)/w0$
#gofmode 5: $w = exp(-(chi2/chi20)/2)/w0$
#gofmode 6: $w = exp(-(chi2/chi20) ** 2)/w0$

Range

$1 \leq$ **#gofmode** ≤ 6

Default

#gofmode 1

global

Flag to use global TALYS cross sections in the search.

Examples

#global n
#global y

Range

y or **n**

Default

#global n

readpar

Flag to sample parameters from pre-constructed (posterior) distribution.

Examples

> **#readpar n**
> **#readpar y**

Range

y or **n**

Default

#readpar n

sens

Flag to use sensitivity coefficients for weights for optimization process.

Examples

#sens n
#sens y

Range

y or **n**

Default

#sens n

readsens

Flag to read sensitivity coefficients from tables to assist in optimization process.

Examples

> **#readsens n**
> **#readsens y**

Range

y or **n**

Default

#readsens n

sample

Flag to sample experimental data randomly for every random TALYS run.

Examples

#sample n
#sample y

Range

y or **n**

Default

#sample n

sort

Flag to sort experimental data in increasing energies in the output.

Examples

> **#sort n**
> **#sort y**

Range

y or **n**

Default

#sort y

magnet

Flag to perturb TALYS data such that a good compromise between calculated and experimental data occurs. This performed by a weighted averaging between the TALYS result and experimental data.

Examples

> **#magnet n**
> **#magnet y**

Range

y or **n**

Default

#magnet n

dsmooth

Parameter for energy smoothing. This keyword is only active if **magnet y**.

Examples

#dsmooth 1.3

Range

1.e-3 $<$ **#dsmooth** $<$ **10**.

Default

#dsmooth 0.5

Tweight

Weight of TALYS vs experiment. This keyword is only active if **magnet y**.

Examples

#Tweight 1.7

Range

0. $<$ **#Tweight** $<$ **1.e10**.

Default

#Tweight 0.5

weightpower

Power for weight of random run, in the exponent of the likelihood function. This keyword is only active if **weight y**.

Examples

> **#weightpower 3.**

Range

$0.1 <$ **#weightpower** < 10.

Default

#weightpower 2

zmin

Minimal target charge number for which experimental data are included in a multi-nuclide search.

Examples

#zmin 26

Range

$1 <$ **#zmin** < 110 and **#zmax** $>$ **#zmin**

Default

#zmin 1

zmax

Maximal target charge number for which experimental data are included in a multi-nuclide search.

Examples

#zmax 26

Range

$1 <$ **#zmax** $< $ **110** and **#zmax** $>$ **#zmin**

Default

#zmax 110

amin

Minimal target mass number for which experimental data are included in a multi-nuclide search.

Examples

#amin 56

Range

$1 < $ **#amin** $ < $ **339** and **#amax** $ > $ **#amin**

Default

#amin A-10, where A represents the most abundant isotope of element Z

amax

Maximal target mass number for which experimental data are included in a multi-nuclide search.

Examples

#amax 60

Range

1 < **#amax** < **339** and **#amax** > **#amin**

Default

#amax A+10, where A represents the most abundant isotope of element Z

ld

Flag for optimization of level density parameters.

Examples

> **#ld n**
> **#ld y**

Range

y or **n**

Default

#ld n

psf

Flag for optimization of photon strength functions.

Examples

#psf n
#psf y

Range

y or **n**

Default

#psf n

e1vary

Flag to automatically vary keywords for E1 radiation.

Examples

#e1vary n
#e1vary y

Range

y or **n**

Default

#e1vary y

m1vary

Flag to automatically vary keywords for M1 radiation.

Examples

#m1vary n
#m1vary y

Range

y or **n**

Default

#m1vary n

7.5   Keywords for parameter optimization

parameters

Flag to vary tabulated nuclear model parameters.

Examples

     #parameters n
     #parameters y

Range

y or **n**

Default

#parameters n

keyvary

Keyword from TALYS parameter list that will be varied.

Examples

> **#keyvary rvadjust**
> **#keyvary M2constant**

Range

#keyvary will work on keywords for TALYS parameters.

Default

#keyvary not used.

partvary

Restriction of parameter variation to certain particles.

Examples

#partvary n
#partvary a

Range

#partvary must be equal to **g, n, p, d, t, h** or **a**

Default

#partvary not used.

ZAvary

Z and A of parameters read from parameter list to be varied.

Examples

#ZAvary 41 93
#ZAvary 92 238

Range

#ZAvary must have Z, A values inside TALYS range

Default

#ZAvary not used.

ZAskip

Z and A of parameters read from parameter list to be skipped for variation.

Examples

#ZAskip 41 93
#ZAskip 92 238

Range

#ZAskip must have Z, A values inside TALYS range

Default

#ZAskip not used.

adeep

Depth in mass number, counted from the compound nucleus, for which parameters are varied in a search.

Examples

#adeep 0

Range

0 < **#adeep** < **numA** where **numA** is the maximum A number

Default

#adeep is not given.

zdeep

Depth in charge number, counted from the compound nucleus, for which parameters are varied in a search.

Examples

#zdeep 0

Range

$0 <$ **#zdeep** $<$ **numZ** where **numZ** is the maximum Z number

Default

#zdeep is not given.

allpar

Flag to vary all nuclear model parameters

Examples

#allpar n
#allpar y

Range

y or **n**

Default

#allpar n

minbar

Minimum number of fission barriers for parameter variation.

Examples

> **#minbar 1**

Range

$0 \leq$ **#minbar** ≤ 3

Default

#minbar 0

Comments

This parameter is helpful when one wants to optimize the level density and fission parameters on the fission barrier, but not on the ground state. In that case, put **#minbar 1**.

maxbar

Maximum number of fission barriers for parameter variation.

Examples

#maxbar 2

Range

$0 \leq$ **maxbar** ≤ 3

Default

#maxbar 3

7.6 Miscellaneous keywords

cwidth

Global scaling factor for the parameter uncertainties to be used in either the optimization or covariance method. This globally reduces or increases the uncertainty bands, instead of doing this parameter by parameter.

Examples

#cwidth 2.

Range

$0 <$ #cwidth < 10.

Default

#cwidth 1.

average

The meaning of the "average" cross section of the final result.

Examples

> **#average 1**: average
> **#average 2**: value from the first run
> **#average 3**: optimum value

Range

$1 \leq$ **#cwidth** ≤ 3

Default

#average 2

autoinclude

Flag to automatically in- or exclude important channels. If **#autoinclude y**, MT1,2,3,4,16,17,18,22,28,51-54,102,103,107 are included for comparison with experimental data. If **#autoinclude n**, each MT needs to be included manually.

Examples

> **#autoinclude n**
> **#autoinclude y**

Range

y or **n**

Default

#autoinclude y

cross

Flag to run TASMAN for covariances or optimization of cross sections.

Examples

> **#cross n**
> **#cross y**

Range

y or **n**

Default

#cross y

gamma

Flag to run TASMAN for covariances or optimization of discrete gamma ray transitions.

Examples

#gamma n
#gamma y

Range

y or **n**

Default

#gamma n

maxgam

Maximum number of discrete levels for which specific gamma ray transitions are considered for covariances or optimization.

Examples

#maxgam 2

Range

$0 \leq maxgam \leq 4$

Default

#maxgam 4

angle

Flag to run TASMAN for covariances or optimization of angular distributions.

Examples

#angle n
#angle y

Range

y or **n**

Default

#angle n

legendre

Flag to run TASMAN for covariances or optimization of Legendre coefficients.

Examples

#legendre n
#legendre y

Range

y or **n**

Default

#legendre n

spectra

Flag to run TASMAN for covariances or optimization of emission spectra.

Examples

#spectra n
#spectra y

Range

y or **n**

Default

#spectra n

discrete

Flag to run TASMAN for covariances or optimization of discrete level cross sections.

Examples

> **#discrete n**
> **#discrete y**

Range

y or **n**

Default

#discrete n

prod

Flag to run TASMAN for covariances or optimization of particle production cross sections.

Examples

#prod n
#prod y

Range

y or **n**

Default

#prod n

production

Flag to run TASMAN for covariances or optimization of isotope production.

Examples

> **#production n**
> **#production y**

Range

y or **n**

Default

#production n

residual

Flag to run TASMAN for covariances or optimization of residual production cross sections.

Examples

#residual n
#residual y

Range

y or **n**

Default

#residual n

macs

Flag to run TASMAN for covariances or optimization of Maxwellian averaged cross sections.

Examples

> **#macs n**
> **#macs y**

Range

y or **n**

Default

#macs n

gamgam

Flag to run TASMAN for covariances or optimization of average radiative width.

Examples

#gamgam n
#gamgam y

Range

y or **n**

Default

#gamgam n

integral

Flag to run TASMAN for covariances or optimization of integral activation data.

Examples

> **#integral n**
> **#integral y**

Range

y or **n**

Default

#integral n

fiscor

Adjustment parameter for fission uncertainties. For actinides, the relative uncertainty of 2nd, 3rd, etc fission is increased to get more realistic results. For this we use the fiscor parameter, which we multiply with some fission related parameters.

Examples

fiscor 1.7

Range

$0.1 \leq$ **fiscor** $\leq$ **10.**

Default

fiscor 3. for actinides, **1** otherwise.

ecis

Flag to repeat ECIS calculations in a second run.

Examples

#ecis n
#ecis y

Range

y or **n**

Default

#ecis y

isomer

Flag to run TASMAN for covariances or optimization of isomeric production cross sections.

Examples

#isomer n
#isomer y

Range

y or **n**

Default

#isomer n

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

inponly

Flag to produce only TALYS input files, without performing random TALYS runs. This enables using input files for future (or parallel) use.

Examples

#inponly n
#inponly y

Range

y or **n**

Default

#inponly n

talysversion

Version of the nuclear model code TALYS.

Examples

#talysversion talys-0.72

Range

The executable should be present in the bin/ directory.

Default

#talysversion talys

tefalversion

Version of the ENDF-6 format generator TEFAL.

Examples

#tefalversion tefal-1.70

Range

The executable should be present in the bin/ directory.

Default

#tefalversion tefal

tanesversion

Version of the code for the fission neutron spectrum, TANES.

Examples

#tanesversion tanes-1.70

Range

The executable should be present in the bin/ directory.

Default

#tanesversion tanes

taresversion

Version of the code for neutron resonances, TARES.

Examples

#taresversion tares-1.70

Range

The executable should be present in the bin/ directory.

Default

#taresversion tares

tafisversion

Version of the code for integral fission quantities TAFIS.

Examples

#tafisversion tafis-1.70

Range

The executable should be present in the bin/ directory.

Default

#tafisversion tafis

offset

Offset for numbering of random files. This can be helpful in long calculations, e.g. for Total Monte Carlo, when different runs on various processors can produce different sets of random files that can later be merged.

Examples

#offset 25

Range

$0 \leq \text{#offset} \leq 10000$

Default

#offset 0

seed

Seed for the random number generator.

Examples

#seed 20032703

Range

$0 <$ #seed $> 1.e9$

Default

#seed 19990303

7.7 Keywords of TASMAN

Table 7.4: The keywords of TASMAN.

Keyword	Range	Default	Page
#acf	y,n	n	97
#adeep	0-339,	not given	179
#adevi	> 0.	various values	129
#allpar	y,n	n	181
#amax	0-339, #amax > #amin	A + 10	168
#amin	0-339, #amax > #amin	A - 10	167
#angle	y,n	n	191
#autoinclude	y,n	y	187
#average	1-3	2	186
#background	filename	file of **adopt 2 151**	110
#band	y,n	n	76
#bdevi	> 0.	various values	130
#block	y,n	n	88
#cdevi	> 0.	various values	131
#change			70
#chi2	1-4	3	134
#chi2max	1.-1.e38	1.e38	150
#covrand	y,n	n	104
#cross	y,n	y	188
#cwidth	> 0	1.	185
#ddevi	> 0.	various values	132
#defonly	y,n	y	95
#deltaE	y,n	n	136
#dev	> 0.	various values	81
#dexp	y,n	y	137
#discrete	y,n	n	194
#dsmooth	1.e-3 - 10.	0.5	162
#e1vary	y,n	y	171
#eaf	y,n	n	98
#Ecent	> 0.	various values	133
#ecis	y,n	y	202
#Edependence	y,n	y	128
#Efracmax	0. - 10.	0.1	142
#Ehigh	1 - 1000.	1000.	101
#equisample	y,n	y	135
#errlim	0.-1.e10	0.03	127
#error	y,n	y	126
#Esearch	0.-1000.	0.-1000.	138
#Esearch1	0.-1000.	0.-1000.	139
#Esearch2	0.-1000.	0.-1000.	140

Continuation of **Table 7.4.**

Keyword	Range	Default	Page
#Eweight	y,n	n	106
#expclass	1-3	3	152
#expexclude			149
#expinclude			148
#extparvar	y,n	n	123
#fiscor	0.1-10.	3.	201
#format	format name	YANDF	87
#Fmax	1.-1.e38	1.e38	145
#fns	y,n	n	116
#fracmax	0.-1.	various values	141
#gamgam	y,n	n	199
#gamma	y,n	n	189
#gauss	y,n	y	77
#getcalcscript	name	none	111
#global	y,n	n	155
#gofmode	1-6	1	154
#gpf	y,n	y	91
#inponly	y,n	n	205
#integral	y,n	n	200
#isomer	y,n	n	203
#kdburn	y,n	y	80
#keyvary	TALYS keyword	not used	175
#ld	y,n	n	169
#legendre	y,n	n	192
#libexclude	library name	none	144
#libinclude	library name	none	143
#liso	0-2	0	99
#lrf7	y,n	n	112
#m1vary	y,n	n	172
#macs	y,n	n	198
#magnet	y,n	n	161
#maxbar	0-3	3	183
#maxexpsets	1-1000000	1000000	151
#maxgam	1-4	4	190
#mcak	y,n	y	79
#metropolis	y,n	y	78
#minbar	0-3	0	182
#mode	1-4	1	71
#mt	y,n	n	96
#Nburn	> 0	#Ntalys	74

Continuation of **Table 7.4.**

Keyword	Range	Default	Page
#Nhigh	> 0	#Ntalys	100
#njoy	y,n	y	120
#Ntalys	> 0	10	73
#nubar	y,n	n	113
#offset	0-10000	0	211
#ompinc	y,n	y	204
#omponly	y,n	n	204
#outsearch	1-5	5	153
#parameters	y,n	n	174
#partvary	g,n,p,d,t,h,a	not used	176
#prepro	y,n	y	119
#proconly	y,n	y	122
#prod	y,n	n	195
#proconly	y,n	y	122
#production	y,n	n	196
#psf	y,n	n	170
#purr	y,n	n	121
#readpar	y,n	n	156
#readsens	y,n	n	158
#residual	y,n	n	197
#resonance	y,n	y	107
#runfns	y,n	n	117
#runnubar	y,n	n	114
#runtares	y,n	n	108
#s20	y,n	n	92
#s30	y,n	n	93
#s60	y,n	n	94
#sample	y,n	n	159
#save	y,n	n	75
#searchmode	1-4	4	125
#seed	0-1.e9	19990303	212
#select	y,n	n	102
#sens	y,n	n	157
#sort	y,n	y	160
#source	name of source	TALYS-2.0	85
#spectra	y,n	n	193
#tafislib	name	none	115
#tafisversion	name	tafis	210
#talexclude			147
#talinclude			146

Continuation of **Table 7.4.**

Keyword	Range	Default	Page
#talysversion	name	talys	206
#taneslib	name	none	118
#tanesversion	name	tanes	208
#tareslib	name	none	109
#taresversion	name	tares	209
#tefalversion	name	tefal	207
#tmc	y,n	n	90
#tmcoffset	≥ 0	0	103
#Tweight	0. - 1.e10	0.5	163
#user	your name	no default	86
#weight	y,n	y	105
#weightpower	0.1 - 10.	2	164
#ZAskip	0-124, 0-339	not given	178
#ZAvary	0-124, 0-339	not given	177
#zdeep	0-124,	not given	180
#zmax	1-110, #zmax > #zmin	110	166
#zmin	1-110, #zmax > #zmin	1	165

8. Outlook and conclusions

This manual describes TASMAN, a code for automatic optimization of TALYS parameters to experimental data and generating uncertainty data with TALYS. With the TASMAN-TALYS-TEFAL combination we can cover the whole path from fundamental nuclear reaction models to the creation of complete data libraries for nuclear applications, including uncertainty propagation. In the last few years, the TASMAN code has been used to show the feasibility of the Monte Carlo covariance method and also to produce covariances for ENDF-6 data libraries. Such covariance data have for example become part of the TENDL and JEFF libraries. Total Monte Carlo (TMC) has also first been realized with TASMAN.

With TASMAN, we have made it possible to generate a complete covariance matrix for nuclear reactions using a Monte Carlo method. The only necessary ingredients are a flexible, validated, and robust nuclear model code that can be called numerous times, and enough computer power. Nuclear model parameters, fed to the TALYS code by means of keywords, are randomly sampled from a Gaussian, uniform or general (numeric) distribution, and the subsequent running of TALYS for many of such cases leads to a full covariance matrix for all cross sections, spectra etc., and nuclear model parameters. Beyond that, TASMAN can be set in Total Monte Carlo (TMC) mode, with which a statistical ensemble of random nuclear data libraries for an isotope is made. This removes the need of covariance matrices in uncertainty propagation. Of course, various aspects of TASMAN can be improved: credibility of the parameter uncertainties, reproduction of experimental uncertainties, the issue of parameter correlations and a general uncertainty assessment of the TALYS results. This last objective requires a serious review of the EXFOR database. With that, one could perform a comparison of blindly calculated TALYS results against all experimental data of the periodic table of elements. The average deviation from experiment would enable us to assess the most pessimistic uncertainties for unmeasured reaction channels. Such a systematical approach towards nuclear data evaluation is as yet non-existent, while it would contribute to a well-founded justification for specific new measurements, and could have a significant impact on nuclear applications that require uncertainties.

Finally, the TMC option changes the view on nuclear data evaluation and validation altogether: although at the moment still seen as too innovative perhaps, we expect this to become a standard method within a decade.

With a clean up version of the EXFOR database, another growing application of TASMAN will be the automatic search for the 'best' set of nuclear model parameters. TASMAN is already used heavily for that and we hope that users will take this further.

A. Yet Another Nuclear Data Format: YANDF

YANDF stands for 'Yet Another Nuclear Data Format'. As it name suggests, it is inspired by the YAML markup language, as YAML is the serialization format which in my view is closest to being human and computer readable at the same time. YANDF is an attempt to unify the nuclear data structure for data sets which come from either TALYS, ENDF nuclear data libraries or EXFOR. The format is aimed to be relatively simple. Once data are stored in YANDF, processing can be done independently from the particular source of the data into JSON, ENDF, GNDS etc. data formats. This may be useful for processing of data from any of these categories for numerical operations, plotting, ML applications etc. The main reason for its construction was a consistent output for the TALYS nuclear model code to enable easier processing towards various applications, but at the same time we took ENDF and EXFOR along in the process. The serialization of YANDF has a key-value schema which is not as non-descriptive as ENDF and not as heavy as GNDS. It aims to give a compact yet complete description of a nuclear reaction in terms of metadata. The source of the data, TALYS, ENDF or EXFOR may give rise to some keywords which are different, but in general the structure is the same for each of these categories. The metadata and associated keywords are supposed to be a direct classification of a nuclear reaction as defined by nuclear physics, such as found in textbooks or journal articles. This means that nuclear reaction observables are leading in the description, and not the ENDF format with its MF/MT numbers and secondary relative distributions and neither the EXFOR format with its emphasis on experimental methods and details. As the role of a nuclear model code is to provide an estimate of all nuclear reaction observables as commonly defined in nuclear physics, we have taken TALYS as the basis for the schema. As EXFOR stores experimental observables, the step from EXFOR to YANDF is almost only a format change: the main keywords are the same as the one of TALYS and all experimental details are stored in their original format for the moment. (The main challenge for EXFOR is to determine *which* EXFOR entries correspond to the reaction channels as defined in ENDF or in TALYS output files. Several different EXFOR categories of data may have to be included or

excluded.) In ENDF libraries, only some data are observables (cross sections, nubar), but most of the data need to be processed into observables using operations on data in different parts of the data file. (Some codes for this exist, like Dece, we have used ENDFtables.)

Before we describe the format in more detail we give 3 examples of a typical YANDF file, for the same nuclear reaction ^{235}U(n,f). This is the file coming from TALYS

```
# header:
#   title: U235(n,f) cross section
#   source: TALYS-2.0
#   user: Arjan Koning
#   date: 2023-11-24
#   format: YANDF-0.1
# target:
#   Z: 92
#   A: 235
#   nuclide: U235
# reaction:
#   type: (n,f)
#   ENDF_MF: 3
#   ENDF_MT: 18
# datablock:
#   quantity: cross section
#   columns: 2
#   entries: 24
##        E              xs
##     [MeV]            [mb]
   1.000000E-11   0.000000E+00
   2.530000E-08   0.000000E+00
   2.000000E-07   0.000000E+00
   1.000000E-06   3.049857E+05
   1.000000E-05   9.592149E+04
.....
```

This is the file for the ENDF-B/VIII.0 data library

```
# header:
#   title: U235(n,f) cross section
#   source: ENDF
#   user: Arjan Koning
#   date: 2023-11-26
#   format: YANDF-0.1
# endf:
#   library: endfb8.0
#   author: IAEA CIELO Collaboration
#   year: 2017
# target:
#   Z: 92
#   A: 235
```

```
#     nuclide: U235
# reaction:
#    type: (n,f)
#    Q-value [MeV]:  1.934054E+02
#    E-threshold [MeV]:  1.000000E-11
#    ENDF_MF: 3
#    ENDF_MT: 18
# datablock:
#    quantity: cross section
#    columns: 4
#    entries: 333
##        E              xs             xslow           xsup
##      [MeV]           [mb]            [mb]            [mb]
    1.000000E-11   0.000000E+00   0.000000E+00   0.000000E+00
    2.250000E-03   0.000000E+00   0.000000E+00   0.000000E+00
    2.250000E-03   2.634378E+03   2.397892E+03   2.870864E+03
    2.250014E-03   2.668097E+03   2.428584E+03   2.907610E+03
    2.250056E-03   2.769988E+03   2.521328E+03   3.018648E+03
....
```

This is the file for one of the experimental data sets in EXFOR

```
# header:
#    title: U235(n,f) cross section
#    source: EXFOR
#    user: Arjan Koning
#    date: 2023-09-18
#    format: YANDF-0.1
# exfor:
#    author: Moore
#    year: 1978
#    subentry: 10629004
#    X4 reaction: 92-U-235(N,F),,SIG
#    X4 source: IAEA-NDS C5 file, database version 2023-07-18
#    X4 link: https://nds.iaea.org/EXFOR/10629004
# target:
#    Z: 92
#    A: 235
#    nuclide: U235
# reaction:
#    type: (n,f)
#    ENDF_MF: 3
#    ENDF_MT: 18
# datablock:
#    quantity: cross section
#    columns: 5
#    entries: 3777
```

```
##         E             dE            xs            dxs        Normalization
##       [MeV]          [MeV]         [mb]          [mb]             []
       1.625000E-06   0.000000E+00  1.304000E+04  1.539000E+02   1.000000E+00
       1.675000E-06   0.000000E+00  1.256000E+04  1.510000E+02   1.000000E+00
       1.725000E-06   0.000000E+00  1.270000E+04  1.549000E+02   1.000000E+00
       1.775000E-06   0.000000E+00  1.226000E+04  1.527000E+02   1.000000E+00
.....
```

Obviously, for EXFOR we have several files which in metadata only differ in the **exfor** keyword. The metadata in the above files completely defines the U235(n,f) reaction.

A.1 Format

The YANDF format is almost equal to the well-known YAML format. If the '#' is removed from the first columns of the metadata header of the above file, we almost have a YAML file. The difference is that we do not quote strings and that the data are given in multi-column format. This means that indentation of the key-value pairs is essential, which is the price that YAML pays for not having to include computational symbols such as '', '[', ',' as in e.g. JSON The above files show the most general keywords without any indentation, while sub-keywords are indented by two spaces, subsub-keywords by 4 spaces, etc. As there are many users who want to use numerical data directly from the file, as in gnuplot or other software, we have chosen to use a '#' at the start of every metadata line. We assume it will not be difficult to remove the '#' and parse the above file to JSON, with either your own script or a patched yaml2json converter.

A YANDF file only contains what is relevant. For example, in the above case there is no specification of any isomeric level in either the target or residual nucleus. Hence, we have decided to leave all 'inactive' metadata out. Parsers will have to take this feature into account.

A.2 Keywords and values

The main keywords should be general enough to describe nuclear reaction observables from at least TALYS, ENDF or EXFOR, but also for additional quantities such as e.g. level densities, photon strength functions, radioisotope yields, etc. as written by TALYS to output files.

A.2.1 header

All YANDF files start with the same keywords:
- **header**: the header of the file describing the main metadata,
    - **title**: the title, generally constructed from the other metadata, enabling the user to see directly which nuclear reaction this concerns
    - **source**: the source of the datafile, this is usually TALYS, ENDF or EXFOR. If calculated uncertainties and covariance data are available, this can also be another source, like e.g. TASMAN
    - **user**: the name of the person who produced this file (e.g. in TALYS you can change the hard-wired name into your own)
    - **date**: the date of the production of this file in yyyy-mm-dd format
    - **format**: version of the YANDF format

A.2.2 endf

When the source is an ENDF library, we have the keywords

- **endf**: specific information from an ENDF file
    - **library**: one of the NDL's such as ENDFB8.0, JENDL5.0, JEFF3.3, TENDL-2021, CENDL3.2 etc,
    - **author**: the author of the evaluation as extracted from the ENDF file
    - **year**: the year of the evaluation as extracted from the ENDF file

A.2.3 exfor

When the source is EXFOR, we have the keywords

- **exfor**: specific information of an experimental data set from EXFOR
    - **author**: first author of the experimental work
    - **year**: the year of the publication of the measurement
    - **subentry**: the EXFOR subentry number
    - **X4 reaction**: the particular EXFOR reaction code as extracted from EXFOR, for checking purposes
    - **X4 source**: the version of EXFOR, and the specific computational form of starting database
    - **X4 link**: the https link to the EXFOR subentry, for all experimental details

A.2.4 target

The first part of a nuclear reaction specification is the target nucleus.

- **target**: the target nucleus
    - **Z**: the charge number
    - **A**: the mass number
    - **nuclide**: the nuclide name

The above keywords are always present. In addition, isomeric level information can be provided by the **level** keywords described below.

A.2.5 reaction

The nuclear reaction may have several keywords for a complete description.

- **reaction**: the nuclear reaction
    - **type**: the nuclear reaction channel
    - **Q-value [MeV]**: the Q-value (only specified when appropriate)
    - **E-threshold [MeV]**: the incident energy threshold (only specified when appropriate)
    - **ENDF_MF**: the ENDF MF number for specification of the type of data
    - **ENDF_MT**: the ENDF MT number for specification of the reaction channel

A.2.6 residual

Often, but not always, a nuclear reaction leads to a well-defined residual nucleus.

- **residual**: the residual nucleus
    - **Z**: the charge number
    - **A**: the mass number
    - **nuclide**: the nuclide name

The above keywords are always present when **residual** is present. In addition, isomeric level information can be provided by the **level** keyword described below.

A.2.7 datablock

Before we read the data, we need to know what we are reading and in what format.

- **datablock**: Description of the data block that follows below.
  - **quantity**: the physical quantity that we are reading
  - **columns**: the number of columns
  - **entries**: the number of entries

Below these keywords always follow 2 lines starting with '##', one with the quantities and the other one with the units.

A.2.8 Keyword: level

The **level** keyword describes the data of a discrete level. It can appear as an keyword under

- **target**, when the target is in an isomeric state
- **reaction**, for scattering off a discrete level
- **residual**, when the residual nuclide is in an isomeric state, or for gamma-ray transitions between discrete states

It is described by

- **level**: Description of discrete level
  - **isomer**: the isomeric number
  - **number**: the level number
  - **energy [MeV]**: the level energy
  - **spin**: the level spin
  - **parity**: the level parity
  - **half-life [sec]**: the half life

In general, the indentation for **level** is 2 spaces, i.e. one below the main keyword, but for discrete level gamma-ray transitions the final level is specified at 4 spaces.

A.2.9 parameters

This is a TALYS-specific keyword. It contains the nuclear models and parameters used in the calculation. It starts with

- **parameters**:

after which various parameters can be given. Here is an example for a level density output file

```
# parameters:
#   ldmodel keyword: 5
#   level density model: Hilaire-Goriely tables
#   Nlow: 8
#   Ntop: 17
#   ctable: -1.621100E-01
#   ptable: -5.763700E-01
```

A.2.10 observables

This is a TALYS-specific keyword. It contains the observables estimated by TALYS used in the calculation. It starts with

- **observables**:

after which various observables can be given. Here is an example for a level density output file

```
# observables:
#    experimental D0 [eV]:  1.200000E+01
#    experimental D0 unc. [eV]:  1.300000E+00
#    theoretical D0 [eV]:  1.245919E+01
#    Chi-2 D0:  1.247688E-01
#    C/E D0:  1.038266E+00
```

Bibliography

[1]   A.J. Koning, S. Hilaire, and S. Goriely. "TALYS: modeling of nuclear reactions". In: *European Physical Journal A* 59 (2023), p. 131.

[2]   D. Smith. "'Covariance matrices for nuclear cross sections derived from nuclear model calculations". In: *Argonne report ANL/NDM-15* (2004).

[3]   P. Dossantos-Uzarralde E. Bauge S. Hilaire. "Evaluation of the covariance matrix of neutronic cross sections with the backward-forward Monte Carlo Method". In: *ND-2007* (2007).

[4]   R. Capote and D.L. Smith. "An investigation of the performance of the unified Monte Carlo method for neutron cross section data evaluation". In: *NDS* 109 (2008), p. 2768.

[5]   AJ Koning. "Bayesian Monte Carlo method for nuclear data evaluation". In: *The European Physical Journal A* 51.12 (2015), p. 184.

[6]   A.J. Koning et al. "TENDL: Complete Nuclear Data Library for innovative Nuclear Science and Technology". In: *Nuclear Data Sheets* 155 (2019), p. 1.

[7]   Arjan J Koning and Dimitri Rochman. "Towards sustainable nuclear energy: Putting nuclear physics to work". In: *Annals of Nuclear Energy* 35.11 (2008), pp. 2024–2030.

[8]   A. Trkov, M. Herman, and D.A. Brown. *ENDF-6 Formats Manual, Data Formats and Procedures for the Evaluated Nuclear Data Files ENDF/B-VI, ENDF/B-VII and ENDF/B-VIII*. CSEWG Document ENDF-102, Report BNL-203218-2018-INRE, SVN Commit: revision 215. 2012.

[9]   J.-Ch. Sublet, P. Ribon, and M. Coste-Delcaux. *CALENDF-2010: User Manual*. Report CEA-R-6277. 2011.

[10] D.E. Cullen. "PREPRO 2021 - ENDF/B6 Pre-processing codes". In: *Technical report IAEA-NDS-0238, IAEA* (2021).

[11] P. Talou. *Prompt fission neutrons calculations in the madland-nix model*. Tech. Rep. LA-UR-07-8168, Los Alamos National Laboratory, Los Alamos, NM, USA. 2007.

[12] D. Madland and J. Nix. "New Calculation of Prompt Fission Neutron Spectra and Average Prompt Neutron Multiplicities". In: *Nucl. Sci. and Eng.* 81 (1982), p. 213.

[13] A. Wahl. *Systematics of fission-product yields*. Tech. Rep. LA-13928, Los Alamos National Laboratory, Los Alamos, NM, USA, May (2002). 2002.

[14] A.J. Koning and D. Rochman. "Modern nuclear data evaluation with the TALYS code system". In: *Nuclear Data Sheets* 113 (2012), p. 2841.

[15] J. Raynal. "Notes on ECIS94". In: *CEA Saclay Report* CEA-N-2772 (1994).

[16] A.J. Koning and J.P. Delaroche. "Local and global nucleon optical models from 1 keV to 200 MeV". In: *Nuclear Physics A* 713.3 (2003), pp. 231–310. ISSN: 0375-9474. DOI: https://doi.org/10.1016/S0375-9474(02)01321-0. URL: https://www.sciencedirect.com/science/article/pii/S0375947402013210.

[17] Efrem Sh Soukhovitskii et al. "Global coupled-channel optical potential for nucleon-actinide interaction from 1 keV to 200 MeV". In: *Journal of Physics G: Nuclear and Particle Physics* 30.7 (May 2004), p. 905. DOI: 10.1088/0954-3899/30/7/007. URL: https://dx.doi.org/10.1088/0954-3899/30/7/007.

[18] Shigueo Watanabe. "High energy scattering of deuterons by complex nuclei". In: *Nuclear Physics* 8 (1958), pp. 484–492. ISSN: 0029-5582. DOI: https://doi.org/10.1016/0029-5582(58)90180-9. URL: https://www.sciencedirect.com/science/article/pii/0029558258901809.

[19] D.G. Madland. "Recent results in the development of a global medium-energy nucleon-nucleus optical model potential". In: *Proceedings of a Specialists' Meeting on preequilibrium nuclear reactions, Semmering, Austria, February 10-12 1988* (), p. 103.

[20] V. Avrigeanu, M. Avrigeanu, and C. Mihaelescu. "Further explorations of the α-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.

[21] AJ Koning. "New working methods for nuclear data evaluation: how to make a nuclear data library?" In: *International Conference on Nuclear Data for Science and Technology*. EDP Sciences. 2007, pp. 679–684.

[22] A.J. Koning, S. Hilaire, and S. Goriely. "Global and local level density models". In: *Nuclear Physics A* 810.1 (2008), pp. 13–76. ISSN: 0375-9474. DOI: https://doi.org/10.1016/j.nuclphysa.2008.06.005. URL: https://www.sciencedirect.com/science/article/pii/S0375947408005903.

[23] J. Kopecky and M. Uhl. "Test of gamma-ray strength functions in nuclear reaction model calculations". In: *Phys. Rev. C* 41 (5 May 1990), pp. 1941–1955. DOI: 10.1103/PhysRevC.41.1941. URL: https://link.aps.org/doi/10.1103/PhysRevC.41.1941.

[24] R. Capote et al. "RIPL - Reference Input Parameter Library for Calculation of Nuclear Reactions and Nuclear Data Evaluations". In: *Nucl.Data Sheets* 110 (2009), p. 3107.

[25] DM Filipescu et al. "Photoneutron cross sections for samarium isotopes: Toward a unified understanding of (γ, n) and (n, γ) reactions in the rare earth region". In: *Physical Review C* 90.6 (2014), p. 064616.

[26] S. Hilaire, Ch. Lagrange, and A.J. Koning. "Comparisons between various width fluctuation correction factors for compound nucleus reactions". In: *Annals of Physics* 306.2 (2003), pp. 209–231. ISSN: 0003-4916. DOI: `https://doi.org/10.1016/S0003-4916(03)00076-9`. URL: `https://www.sciencedirect.com/science/article/pii/S0003491603000769`.

[27] A.J. Koning and M.C. 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. ISSN: 0375-9474. DOI: `https://doi.org/10.1016/j.nuclphysa.2004.08.013`. URL: `https://www.sciencedirect.com/science/article/pii/S037594740400870X`.

[28] S. Goriely et al. "Gogny-HFB+QRPA dipole strength function and its application to radiative nucleon capture cross section". In: *Phys. Rev. C* 98 (1 July 2018), p. 014327. DOI: `10.1103/PhysRevC.98.014327`. URL: `https://link.aps.org/doi/10.1103/PhysRevC.98.014327`.