RESONANCETABLES-2.1

Database for thermal cross sections, MACS and average resonance parameters

Arjan Koning and Dimitri Rochman

s

nds.iaea.org/talys

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

December 2025

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, and initiating the '3 T's': the TALYS nuclear model code, Total Monte Carlo uncertainty propagation and the TENDL nuclear data library. As of 2025, his h-index is 62 with more than 26 000 citations.
Although Arjan is currently in a managerial role, he aims to keep his scientific creativity alive by maintaining and extending TALYS plus all software that emerges from that. Pleas from his friends to also spend time on other things are sometimes honoured.

Preface

RESONANCETABLES is a directory-structured database with compiled and evaluated data for experimental thermal cross sections, resonance integrals, Maxwellian-averaged cross sections (MACS) and other average quantities in the resonance range.

RESONANCETABLES is, along with its counterparts ENDFTABLES and EXFORTABLES, another initiative which is deemed necessary for a future with efficient nuclear data evaluation, in which machine learning and automation will play a large role. The objective is to have all important fundamental nuclear data readily available for direct inclusion in software. This is rather normal for codes like e.g. MCNP, which requires a complete processed nuclear data library such as ENDF/B or JEFF for all nuclides, ready to use. For some reason, for software and data that produce evaluated nuclear data libraries this is generally not seen as necessary. Here nuclides are evaluated and formatted one by one, and retrieval of experimental data by e.g. a graphical web interface is seen as sufficient for evaluation purposes. Manual insertion of values in an evaluated data file is still being done. It is probably not surprising that the production of the TENDL library relies strongly on the availability of 'fundamental' databases as the ones presented here, but we think they can also be helpful for the other libraries, which are still produced by manual evaluation. Also, these databases can help assessing the overall quality of a nuclear data library, not only TENDL, and be used to validate nuclear model codes, or fine-tune the parameters of nuclear models. In that sense, this initiative can be seen as a modest extension of IAEA's RIPL project, which was a major step forward in efficient nuclear data evaluation. Ideally RESONANCETABLES should be replaced by a versatile API in the future.

License, contact and reference

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

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

- When RESONANCETABLES is used for your reports, publications, etc., please make a proper reference to the code. At the moment this is:
  When you refer to the application of this software:
  D. Rochman, A.J. Koning, J.-Ch. Sublet, A statistical analysis of evaluated neutron resonances with TARES for JEFF-3.3, JENDL-4.0, ENDF/B-VIII.0 and TENDL-2019, Nuclear Data Sheets 163, 163 (2020).
  When you refer to something particular of this tutorial:
  A.J. Koning and D. Rochman, RESONANCETABLES-2.1: Database for thermal cross sections, MACS and average resonance parameters, IAEA NDS Document Series IAEA(NDS)-234, December 2020

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

Acknowledgements

I wish to thank a few persons who have contributed to the present state of RESONANCETABLES:

- Dimitri Rochman for providing the numerical tables of some of the input data for this database and helpful advice,
- Shin Okumura for providing many experimental data from EXFOR,
- Jura Kopecky for information on the various original source files used for this database,
- Andrej Trkov for providing the Kayzero database,
- Anatoli Ignatyuk and other RIPL members for constructing the starting databases.

Arjan Koning

Contents

1. Introduction

Neutron reaction data in and below the resonance range are crucial parts of neutron data libraries, especially for low-energy neutron applications. Therefore, resonance analyses are often performed on raw reaction data to produce resonance parameter evaluations. A proper evaluation reproduces, apart from the experimental resonances, the experimental value of the thermal cross section for all open channels as well as average resonance properties. Often, these evaluations are performed and published for a single nuclide, mostly when new experimental data comes available. Compilations, and in some cases evaluations of such data for all nuclides exist, with probably the various versions of the Atlas of Neutron Resonances by Mughabghab [1] and the Reference Input Parameter Library (RIPL) [2] as the most famous examples.

These databases of resonance parameters were often made manually, i.e. numbers read from publications were inserted one by one with a word processor, unavoidably leading to errors. Also the evaluator could decide that not one, but the average of several measured values would end up in the database, without any documentation on why and how this average was obtained. Actually, it is often difficult to distinguish between a compilation (adoption of raw data without any further evaluation) or an evaluation, especially if in the evaluation one compiled value among a few alternative values is chosen as 'the' answer.

In an ideal world, there should exist one mother database, with EXFOR as the best candidate, which contains all the experimental values. An evaluated compilation of all relevant data of one type would be produced by an automated procedure combined with detailed knowledge, which discards, includes, corrects and averages various data from EXFOR to come to a final evaluated value with an uncertainty margin. Unfortunately, we are not in that situation. There is probably consensus about what are valuable and reasonably complete evaluations (the latest version of the Atlas is an obvious candidate) but HOW the final numbers in such databases were produced remains a secret to the author, although sometimes one can recognize a one-to-one correspondence with the original published values. The same is true for high-quality evaluated data available in ENDF formatted libraries. The process to create such an evaluation may be documented in a publication

or at the top of the ENDF file, in which claims may have been made why the used value for e.g. the thermal capture cross section is the best to date. Again, this detailed info exists for several nuclides but one needs to dig through many publications and reports. Often, after reproducing the data file in pointwise format, one can guess which data was adopted.

Besides repeating the evaluation consistently for the entire nuclide chart, which is a huge task and one may wonder whether there is an individual left with both the knowledge and willingness to do that, all we can do on short term is to collect these data and compare it with other compilations and perhaps try to create a 'best' database for each type of data.

RESONANCETABLES is merely a start to this and is restricted to important nuclear data values at only a few important energies (thermal energy, 30 keV) or an average over the entire resonance range. Individual resonances and their parameters are not considered here (yet).

The databases presented here are entirely based on previous compilations (e.g. RIPL), evaluations (e.g. Mughabghab's various versions of the Atlas of Neutron Resonances), and raw experimental data sets (EXFOR). A Fortran code *resonancetables.f90* has been written which reads in the entire collection of mother databases, performs some operations, and produces the RESONANCETABLES database. During this database translation, statistical tests are performed on the various mother databases and the results of these tests are written to various diagnostic files, revealing possible outliers, which can be taken into account for correction of the data of the mother databases. The most important output is a logical directory structure for all relevant data in an easy readable and unified format.

Hence, RESONANCETABLES is a directory-structured database, derived from the basic files mentioned above. In a different project, ENDFTABLES, which concerns the production of all ENDF data libraries in tabular format together with its statistical analysis, the values from those ENDF libraries are compared with the adopted values from this project. These ENDF values are included in our tables as well.

Eventually, this database could be replaced by an API which allows users to extract specific data at command-line level. That does not yet exist however. The produced database is in so-called YANDF (Yet Another Nuclear Data Format) format, as explained in an Appendix. It will be easy to translate the output in JSON format if this is deemed useful.

1.1 This tutorial

After this Introduction, you will find the following,

Chapter 2: Installation guide for RESONANCETABLES. A general outline of the contents of the RESONANCETABLES database, and also an explanation on how to (optionally) regenerate the database yourself.

Chapter 3: A description of the input to RESONANCETABLES: The mother database with compilations and evaluations by others.

Chapter 4: A description of the priority rules for processing the mother database into the selected database

Chapter 5: The selected database.

Chapter 6: Outlook and conclusions.

2. Installation and getting started

2.1 The RESONANCETABLES package

In what follows we assume RESONANCETABLES will be installed on a Linux or MacOS operating system. In total, you will need about 20 Mb of free disk space to install RESONANCETABLES. RESONANCEtables contains the following directories and files:

- *LICENSE* is the license file,
- *README.md* outlines the contents of the package,
- *install_resonancetables.bash*, *code_build.bash* and *path_change.bash* are scripts that take care of the installation,
- *source/* is the Fortran source of the code that performs the entire database creation and checking of the data. Most users will however only need the results already provided by us and not want to rerun the code,
- *files/* the mother database, i.e. the input files to *resonancetables.f90*,
- *doc/* contains the documentation: this manual in pdf format,
- *thermal/* the produced thermal cross section database,
- *macs/* the produced Maxwellian-Averaged Cross Section (MACS) database,
- *resonance/* the produced average resonance parameter database,
- *libs/* contains the comparison of the database values with those of the major evaluated nuclear data libraries. These files are not produced by RESONANCETABLES but imported from another project.

Most users may only be interested in the contents of the database. **Thus, you do not need to further install or run anything and can use the database directly.** However, if you want to regenerate the database yourself with the *resonancetables.f90* code, continue with the following section.

2.2 Installation

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

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

or by getting the tar file

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

Although most users will only be interested in the final RESONANCETABLES database, there may be reasons to regenerate the entire database from scratch again, such as:

- your local changes in *resonancetables.f90* which improve or extend the database, (of course I advise to share that with the author),
- newer versions of the mother databases,
- etc.

We here provide the necessary steps to do the installation, For a Linux or MacOs system, the installation is expected to be handled by the *install_resonancetables.bash* script, as follows

- if necessary, edit *code_build.bash* and set the first two variables: the name of your compiler and its flags.
- **install_resonancetables.bash**

An alternative installation option is

- **cd resonancetables/source**
- **make**

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

- **cd resonancetables/source**
- Ensure that RESONANCEtables can read directories with data. This is done in subroutine *machine.f90*. If *code_build.bash* has not already replaced the path name in *machine.f90*, do it yourself. We think this is the only machine dependence of RESONANCEtables. We expect no complaints from the compiler.
- **gfortran -c *.f90**
- **gfortran *.o -o resonancetables**
- **mv resonancetables ../bin**

If you run the RESONANCETABLES code, it will overwrite all existing directories. Note that RESONANCETABLES has no input file, it only has one task: to produce a complete collection of resonance quantities. Hence you may type

resonancetables

A run will take about 5 minutes.

3. The mother database

3.1 Compiled and evaluated databases

The input files for RESONANCETABLES, the so-called mother databases, come in a variety of completeness, readability and formats, from different contributors. They are stored in the directory *files/*. Often these are multi-column files (some *very* multi) with many different quantities, blanks for cases where there are no data, etc. and quite some effort was invested to properly read these input files.

At the moment, the following files in the mother database are processed:

- *atlas_2006.txt*: Mughabghab 2006 Atlas of neutron resonances [3], including many corrections by Dimitri Rochman
  - Thermal (n,el), (n,γ), (n,f), (n,α) cross sections, total, prompt, and delayed nubar
- *global.2023.txt*: Mughabghab 2018 Atlas of neutron resonances [1], including many corrections by Dimitri Rochman
  - Thermal (n,tot), (n,el), (n,γ), (n,f), (n,p), (n,α) cross sections, total, prompt, and delayed nubar
  - Maxwellian-averaged cross sections (MACS)
  - Thermal (n,γ), (n,f) resonance integrals
  - Average neutron resonance spacings D_0, D_1, neutron strength functions S_0, S_1, and average radiative widths Γ_γ
- *sukhoruchkin.txt*: Sukhoruchkin 2015 Neutron resonance parameters [4], digitized by Dimitri Rochman
  - Thermal (n,γ), (n,f), (n,p), (n,α) cross sections
  - Thermal (n,γ), (n,f) resonance integrals
  - MACS
- Reference Input Parameter Library (RIPL) [2]
  - *thermal.ripl*: Thermal neutron capture cross section database compiled by Jura Kopecky,
  - *resonance_0.ripl2*: Average resonance parameters D_0, S_0, Γ_γ, compiled and evaluated

by Anatoli Ignatyuk for RIPL-2.
  - *resonance_1.ripl2*: Average resonance parameters D_1, S_1, Γ_γ, compiled and evaluated by Anatoli Ignatyuk for RIPL-2.
  - *resonance_0.ripl3*: Average resonance parameters D_0, S_0, Γ_γ, compiled and evaluated by Anatoli Ignatyuk for RIPL-3.
  - *resonance_1.ripl3*: Average resonance parameters D_1, S_1, Γ_γ, compiled and evaluated by Anatoli Ignatyuk for RIPL-3.
  - *obninsk.dat*: File with number of resonances taken into account for evaluation, compiled and evaluated by Anatoli Ignatyuk for RIPL-2.
- Other compilations and evaluations
  - *kayzero.txt*: Kay-zero database for thermal (n,γ) cross sections, obtained from Andrej Trkov [5]
  - *firestone2022.txt*: database for thermal (n,γ) cross sections, obtained from Rick Firestone [6]
  - *resint.juko*: Resonance Integral database by Jura Kopecky [7]
  - *macs_kadonis.ng*: KADONIS database for MACS [8]
  - *macs_astral.ng*: ASTRAL database for MACS [9]
  - *macs_bao.ng*: Bao database for MACS [10]
- EXFOR database [11] for all the above quantities
  - *exforfiles*: EXFOR data from Shin Okumura parsed database for,
    * resonance parameters, https://github.com/shinokumura/resonance_data,
    * thermal cross sections, https://github.com/shinokumura/thermaldata
    * direct EXFOR entry search with the Data Explorer, e.g. https://nds.iaea.org/dataexplorer/exfor/entry/40007-0
  - *~/exfortables*: EXFOR data from EXFORtables [12]

3.2 Values from evaluated nuclear data libraries

To get a complete overview of all possible values, we have also included the values extracted from the evaluated nuclear data libraries, TENDL-2023 [13], ENDF/B-VIII.1 [14], JEFF-4.0 [15], JENDL-5.0 [16] and CENDL-3.2 [17]. This gives at least an indication of which values other evaluators have adopted. The values for the thermal cross sections, resonance integrals and MACS have been calculated using the INTER code [18] on the nuclear data files in pointwise form, produced with PREPRO's RECENT module [19] and have been put in tables in the *libs/* directory.

These tables have been produced with the script *libmaker* which is used to analyze the world's nuclear data libraries and to put the data in tabular format. The actual C/E validation of the world's nuclear data libraries with the RESONANCETABLES database is done in that process, which produces the so-called *libraries/* directory needed for several of our codes. The results are however available in the files in the *libs/* subdirectory of RESONANCETABLES.

4. Processing the mother database

The largest challenge in a project like this is to determine the *best* value for each case. This is what is part of nuclear data evaluation and this has only been done to a restricted extent here as most, if not all, of that is done by the evaluator of the mother databases. By intercomparing databases we could at least filter out the most obvious errors and make a correction table which acts on the mother database to produce the selected values. (For example, this had to be done for the thermal cross sections from RIPL). Ideally, the *best* value comes from international consensus of specialists. Here we speak of the *selected* value, which means “selected by us”.

4.1 Priorities for the selected database

The priority for adopting values from the various databases is based on date ("newer is better"), experience and proven quality. For thermal cross sections and resonance integrals, the order of adoption is

1. Kayzero database
2. Mughabghab 2018 Atlas
3. Sukhoruchkin 2015 Atlas
4. Mughabghab 2006 Atlas
5. RIPL or Kopecky database
6. EXFOR (the most recent value)

For MACS, the order of adoption is

1. KADONIS database
2. ASTRAL database
3. Bao database
4. Mughabghab 2018 Atlas
5. Sukhoruchkin 2015 Atlas
6. EXFOR (the most recent value)

For D_0, D_1, the order of adoption is

1. RIPL
2. Mughabghab 2018 Atlas
3. EXFOR (the most recent value)

For S_0, S_1 and Γ_γ, the order of adoption is

1. Mughabghab 2018 Atlas
2. RIPL
3. EXFOR (the most recent value)

So the procedure is as follows, if a value exists in 1, it is adopted, if not, if a value exists in 2, it is adopted, etc. In future versions of the code these priorities can be changed if appropriate, also on a nuclide-by-nuclide basis. For that, we need input from experts.

4.2 Ratios between databases

To compare different mother databases for the same quantity, it is helpful to define a ratio,

$$R = \frac{\text{value of mother database}}{\text{value of selected database}} \tag{4.1}$$

i.e. the mother database is one of the individual databases mentioned in the previous Section and the selected database is the one obtained via the aforementioned priority rules. For every produced database, a column with ratios is included, see the next Chapter.

5. The selected database

5.1 The RESONANCETABLES database

We have stored the final data directory-wise per reaction type. RESONANCETABLES consists of directories *thermal/*, for thermal cross sections, *macs/*, for Maxwellian-averaged cross sections (MACS) and *resonance/*, for average resonance parameters and integrals, respectively. These directories contain all reaction data which **so far** could be processed from the mother database. All files are given in YANDF format.

We first outline the classification of the filename extensions used in the database,

5.1.1 Thermal cross sections

The directory *thermal/* contains subdirectories

- *el*: thermal elastic cross sections
- *na*: thermal (n,α) cross sections
- *na-g*: thermal (n,α) cross sections to ground state
- *na-m*: thermal (n,α) cross sections to isomer
- *nf*: thermal fission cross sections
- *ng*: thermal (n,γ) cross sections
- *ng-g*: thermal (n,γ) cross sections to ground state
- *ng-m*: thermal (n,γ) cross sections to isomer
- *ng-n*: thermal (n,γ) cross sections to second isomer
- *np*: thermal (n,p) cross sections
- *nu*: thermal total nubar
- *nud*: thermal delayed nubar
- *nup*: thermal prompt nubar
- *tot*: thermal total cross sections

Each directory has two subdirectories, *nuc* and *all*.

5.1.2 Per nuclide: the nuc directory

In this directory, all entries from compilations, EXFOR and nuclear data libraries are given in one file per nuclide. Also the selected value and its source is given as well as statistical averages over the various classes of data and the ratio of all the data over the selected value. The files have names like *Ag109_ng.txt*, and that particular example is given below. As is clear, all required metadata are given in YANDF format.

```
# header:
#   title: Ag107(n,g) thermal cross section
#   source: Resonancetables
#   date: 2025-05-26
# target:
#   Z: 47
#   A: 107
#   nuclide: Ag107
# reaction:
#   type: (n,g)
# observables:
#   selected value [b]:  3.243000E+01
#   selected value uncertainty [b]:  2.210000E+00
#   selected value source: Kayzero
#   number of values: 19
#   average value [b]:  3.355059E+01
#   relative standard deviation [%]:      26.030437
# quantity:
#   type: Compilation
#   average value:  3.556000E+01
#   relative standard deviation [%]:       5.443347
# datablock:
#   columns: 10
#   entries: 5
##    Author                         Type           Year           Value         dValue
##      []              []             []             []             [b]           [b]
      RIPL-3                  Compilation          2004       3.537000E+01   1.200000E+00
      Mughabghab-2006         Compilation          2006       3.760000E+01   1.200000E+00
      Sukhoruchkin            Compilation          2015       3.760000E+01   1.200000E+00
      Mughabghab-2018         Compilation          2018       3.480000E+01   7.000000E-01
      Kayzero                 Compilation          2018       3.243000E+01   2.210000E+00
# quantity:
#   type: EXFOR
#   average value:  3.554000E+01
#   relative standard deviation [%]:       4.549551
# datablock:
#   columns: 10
#   entries: 5
##    Author                         Type           Year           Value         dValue
##      []              []             []             []             [b]           [b]
      T.B.Ryves               EXFOR                1971       3.720000E+01   0.000000E+00
      T.B.Ryves               EXFOR                1974       3.760000E+01   1.200000E+00
      F.Farina Arbocco        EXFOR                2013       3.420000E+01   0.000000E+00
```

```
      F.Farina Arbocco        EXFOR                2013        3.520000E+01   0.000000E+00
      F.Farina Arbocco        EXFOR                2013        3.350000E+01   0.000000E+00
# quantity:
#   type: EXFOR spectrum-averaged
#   average value:  2.677000E+01
#   relative standard deviation [%]:      55.152290
# datablock:
#   columns: 10
#   entries: 5
##    Author                          Type            Year           Value         dValue
##      []              []              []              []             [b]           [b]
      G.A.Linenberger         EXFOR                1946        2.530000E+01   0.000000E+00
      L.Seren                 EXFOR                1947        5.000000E-02   1.000000E-02
      L.Seren                 EXFOR                1947        4.430000E+01   8.860000E+00
      H.Pomerance             EXFOR                1952        2.990000E+01   2.392000E+00
      F.Farina Arbocco        EXFOR                2013        3.430000E+01   9.000000E-01
# quantity:
#   type: Nuclear data library
#   average value:  3.702782E+01
#   relative standard deviation [%]:       4.098966
# datablock:
#   columns: 10
#   entries: 4
##    Author                          Type            Year           Value         dValue
##      []              []              []              []             [b]           [b]
      cendl3.2                NDL                  2019        3.862100E+01   0.000000E+00
      tendl.2023              NDL                  2023        3.750040E+01   0.000000E+00
      endfb8.1                NDL                  2024        3.746450E+01   0.000000E+00
      jeff4.0                 NDL                  2025        3.452540E+01   0.000000E+00
```

These files also help to uncover errors and to determine whether there is a large variation among databases in general. A metric for the latter is the value of the relative standard deviation.

5.1.3 Compilations and other collections: the all directory

Here the data are summarized into Z, A tables. Each *all* subdirectory has files like given below, where we give examples for the thermal (n,γ) reaction:

- *selected_ng.txt*, probably the most important file, since this file contains what we at the moment consider to be the best choice for each entry, and it is also the most complete. This file is the result of the aforementioned priority rules. Note that for all these files, the reference for each entry is given in the last column. Occasionally, this is a reference to an EXFOR entry, which shows that not all existing data have been taken into account in some compilations and evaluations. Whether such cases are overlooked or left out on purpose (because e.g. the values are unreliable) is not known. The EXFOR entries can be directly inspected from https://github.com/shinokumura/resonance_data, and for thermal cross sections, https://github.com/shinokumura/thermaldata.
- *RIPL-3_ng.txt*, *Mughabghab-2018_ng.txt* etc. Here the original 'mother' databases are presented in the unified YANDF format.
- *tendl.2023_ng.txt*, *endfb8.1_ng.txt*, etc. Here the values as derived from the major nuclear data libraries are presented.

Each line of a cross section file contains charge number of target (Z), mass number of target (A), isomer of target (Liso), cross section in barn (xs), uncertainty of cross section in barn (dxs), the ratio over the selected value and if appropriate (usually for EXFOR) the reference.

The files in the *thermal/ng/all* directory are:

```
EXFOR_ng.txt
Kayzero_ng.txt
Mughabghab-2006_ng.txt
Mughabghab-2018_ng.txt
RIPL-3_ng.txt
Sukhoruchkin_ng.txt
cendl3.2_ng.txt
endfb8.1_ng.txt
jeff4.0_ng.txt
jendl5.0_ng.txt
selected_ng.txt
tendl.2023_ng.txt
```

and similar for all other reaction channels.

Below is the top of the output of *thermal_ng.txt*:

```
# header:
#   title: (n,g) thermal cross section
#   source: Resonancetables
#   date: 2025-05-26
# reaction:
#   type: (n,g)
# quantity:
#   type: thermal cross section
# datablock:
#   columns: 9
#   entries: 580
##       Z              A            Liso          Value         dValue        Reference
##      []             []              []            [b]           [b]             []
         1              0               0       3.325000E-01  6.000000E-04  Firestone
         1              1               0       3.326000E-01  7.000000E-04  Mughabghab-201
         1              2               0       4.950000E-04  6.000000E-06  Mughabghab-201
         1              3               0       6.000000E-06  0.000000E+00  Sukhoruchkin
         2              3               0       5.500000E-05  3.000000E-06  Mughabghab-201
         3              0               0       7.116000E+01  2.400000E-01  Firestone
         3              6               0       3.930000E-02  7.000000E-04  Mughabghab-201
         3              7               0       4.420000E-02  5.000000E-04  Mughabghab-201
         4              0               0       8.830000E-03  2.200000E-04  Firestone
         4              7               0       1.550000E-01  0.000000E+00  P.Bassi
         4              9               0       8.270000E-03  1.300000E-04  Mughabghab-201
         4             10               0       1.000000E-03  0.000000E+00  Mughabghab-201
.....................
```

5.1.4 Maxwellian-averaged cross sections

The directory *macs/* contains a subdirectory with 30 keV data

- *ng/*: MACS (n,γ)

This directory has two subdirectories, *nuc* and *all* with a similar structure as that described above. The files in the *macs/ng/all* directory are:

```
Astral_macs.txt
Bao_macs.txt
EXFOR_macs.txt
Kadonis_macs.txt
Mughabghab-2018_macs.txt
Sukhoruchkin_macs.txt
cendl3.2_macs.txt
endfb8.1_macs.txt
jeff4.0_macs.txt
jendl5.0_macs.txt
selected_macs.txt
tendl.2023_macs.txt
```

5.1.5 Average resonance data

The directory *resonance/* contains subdirectories with average resonance data

- *Ig/*: (n,γ) resonance integrals
- *If/*: (n,f) resonance integrals
- *D0/*: average neutron s-wave resonance spacings
- *D1/*: average neutron p-wave resonance spacings
- *D2/*: average neutron d-wave resonance spacings
- *S0/*: average neutron s-wave strength functions
- *S1/*: average neutron s-wave strength functions
- *gamgam/*: average neutron radiative widths for l=0
- *gamgam1/*: average neutron radiative widths for l=1

These directories has two subdirectories, *nuc* and *all* with a similar structure as that described above.

The files in the *resonance/gamgam/all* directory are:

```
EXFOR_gamgam.txt
Mughabghab-2018_gamgam.txt
RIPL-2_gamgam.txt
RIPL-3_gamgam.txt
selected_gamgam.txt
```

5.2 Comparison with nuclear data libraries

Outside this database, we have all world nuclear data libraries available in pointwise format. This allows to give a numerical and graphical comparison of the values of the data libraries and the current database, where we take our 'selected' database, i.e. the one chosen according to the priority rules, as reference.

The files in the *libs/* directory also contain the root-mean-square values (F-value) and a few other statistical measures for the deviation of nuclear data libraries from the database. The files with *_bin* in their filename contain a histogram with the C/E values as well as the F-value, the chi-2 value, the number of experimental values that could be compared with the library N, as well as the number of cases inside a certain range from experiment.

For thermal capture cross sections the statistical summary is

```
==> cendl3.2_bin.therm <==
# F (< 20%) N   N < 5%      N < 20%     N < 50%     Chi-2 (< 20%)
#   1.034 227  149(0.656)  201(0.885)  211(0.930)  11.248

==> endfb8.1_bin.therm <==
# F (< 20%) N   N < 5%      N < 20%     N < 50%     Chi-2 (< 20%)
#   1.021 414  315(0.761)  374(0.903)  391(0.944)   5.938

==> jeff3.3_bin.therm <==
#     F     Chi-2 N   N inside 5% N inside 20% N inside 50%
#   1.024   4.772 425 307(0.722)  377(0.887)  397(0.934)

==> jeff4.0_bin.therm <==
# F (< 20%) N   N < 5%      N < 20%     N < 50%     Chi-2 (< 20%)
#   1.015 422  359(0.851)  405(0.960)  411(0.974)   5.491

==> jendl5.0_bin.therm <==
# F (< 20%) N   N < 5%      N < 20%     N < 50%     Chi-2 (< 20%)
#   1.020 431  316(0.733)  369(0.856)  384(0.891) 226.477

==> tendl.2023_bin.therm <==
# F (< 20%) N   N < 5%      N < 20%     N < 50%     Chi-2 (< 20%)
#   1.016 443  360(0.813)  408(0.921)  419(0.946)  95.943

==> tendl.2024_bin.therm <==
# F (< 20%) N   N < 5%      N < 20%     N < 50%     Chi-2 (< 20%)
#   1.011 439  389(0.886)  425(0.968)  429(0.977)   6.075
```

For MACS this is

```
==> cendl3.2_bin.MACS <==
# F (< 20%) N   N < 5%      N < 20%     N < 50%     Chi-2 (< 20%)
#   1.069 175   46(0.263)  104(0.594)  145(0.829)  17.565

==> endfb8.1_bin.MACS <==
# F (< 20%) N   N < 5%      N < 20%     N < 50%     Chi-2 (< 20%)
#   1.064 273   94(0.344)  192(0.703)  242(0.886)  13.338

==> jeff3.3_bin.MACS <==
#     F     Chi-2 N   N inside 5% N inside 20% N inside 50%
#   1.076 17.448 274  74(0.270)  186(0.679)  242(0.883)

==> jeff4.0_bin.MACS <==
# F (< 20%) N   N < 5%      N < 20%     N < 50%     Chi-2 (< 20%)
#   1.062 267  100(0.375)  190(0.712)  238(0.891)  10.213

==> jendl5.0_bin.MACS <==
# F (< 20%) N   N < 5%      N < 20%     N < 50%     Chi-2 (< 20%)
#   1.064 272   94(0.346)  195(0.717)  243(0.893)  11.083

==> tendl.2023_bin.MACS <==
# F (< 20%) N   N < 5%      N < 20%     N < 50%     Chi-2 (< 20%)
#   1.077 276   72(0.261)  183(0.663)  238(0.862)  14.880

==> tendl.2024_bin.MACS <==
# F (< 20%) N   N < 5%      N < 20%     N < 50%     Chi-2 (< 20%)
#   1.052 273  117(0.429)  196(0.718)  240(0.879)   9.862
```

For the neutron capture resonance integrals this is

```
==> cendl3.2_bin.RI <==
# F (< 20%) N   N < 5%      N < 20%     N < 50%     Chi-2 (< 20%)
#   1.034 219  121(0.553)  164(0.749)  191(0.872)   3.068

==> endfb8.1_bin.RI <==
# F (< 20%) N   N < 5%      N < 20%     N < 50%     Chi-2 (< 20%)
#   1.030 385  240(0.623)  307(0.797)  346(0.899)   4.115

==> jeff3.3_bin.RI <==
#     F     Chi-2 N   N inside 5% N inside 20% N inside 50%
#   1.038   8.500 392 201(0.513)  283(0.722)  328(0.837)

==> jeff4.0_bin.RI <==
# F (< 20%) N   N < 5%      N < 20%     N < 50%     Chi-2 (< 20%)
#   1.043 387  179(0.463)  264(0.682)  318(0.822)   4.644

==> jendl5.0_bin.RI <==
# F (< 20%) N   N < 5%      N < 20%     N < 50%     Chi-2 (< 20%)
#   1.038 412  205(0.498)  282(0.684)  342(0.830)   4.023

==> tendl.2023_bin.RI <==
# F (< 20%) N   N < 5%      N < 20%     N < 50%     Chi-2 (< 20%)
#   1.043 418  177(0.423)  266(0.636)  323(0.773)   4.179

==> tendl.2024_bin.RI <==
# F (< 20%) N   N < 5%      N < 20%     N < 50%     Chi-2 (< 20%)
#   1.045 415  174(0.419)  262(0.631)  318(0.766)   4.626
```

Below are comparisons as histograms for C/E values as well as scatter plots as a function of A. These may be useful for nuclear data library projects that want to identify and solve outlier cases, or declare that our selected database does not contain the correct values.

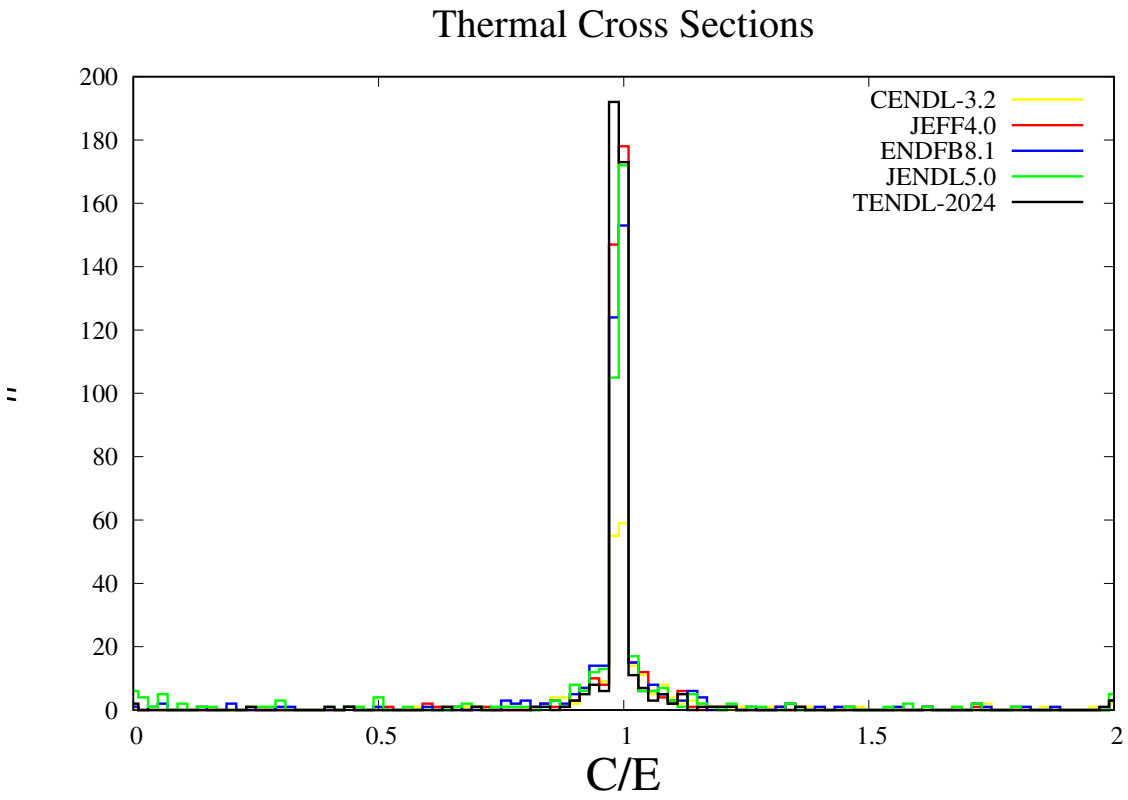

Figure 5.1: C/E histogram for thermal cross sections

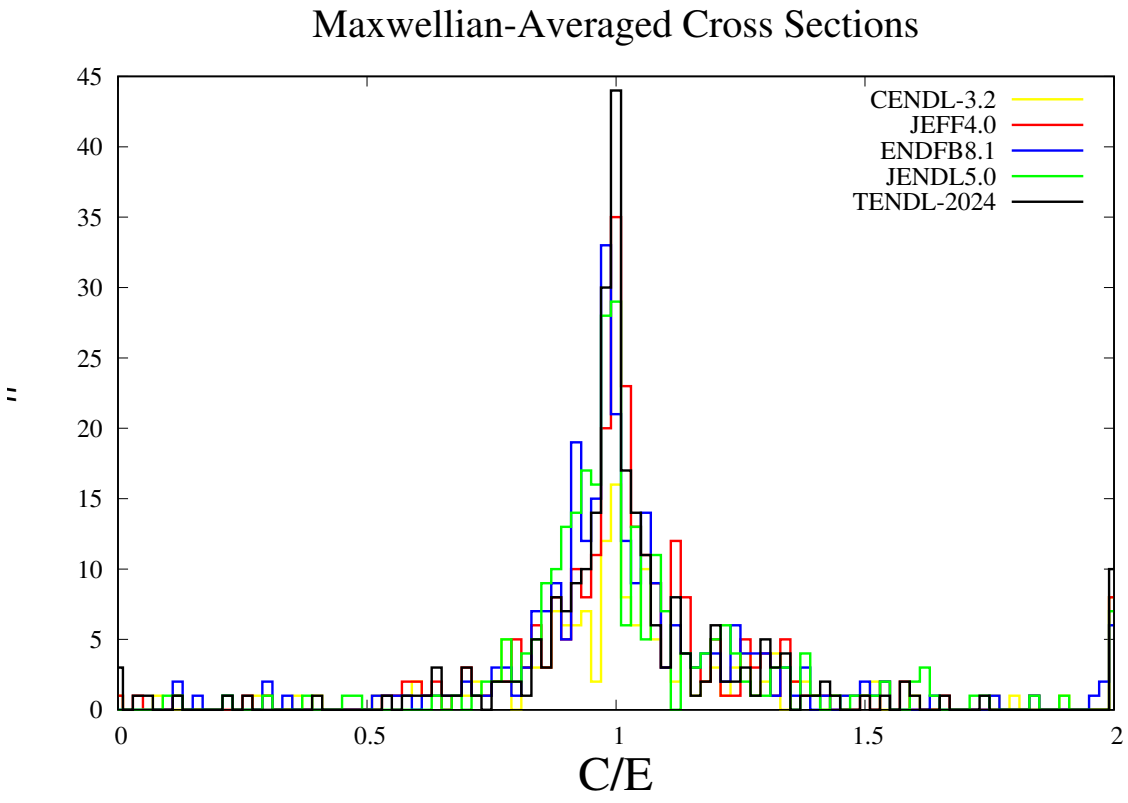

Figure 5.2: C/E histogram for MACS

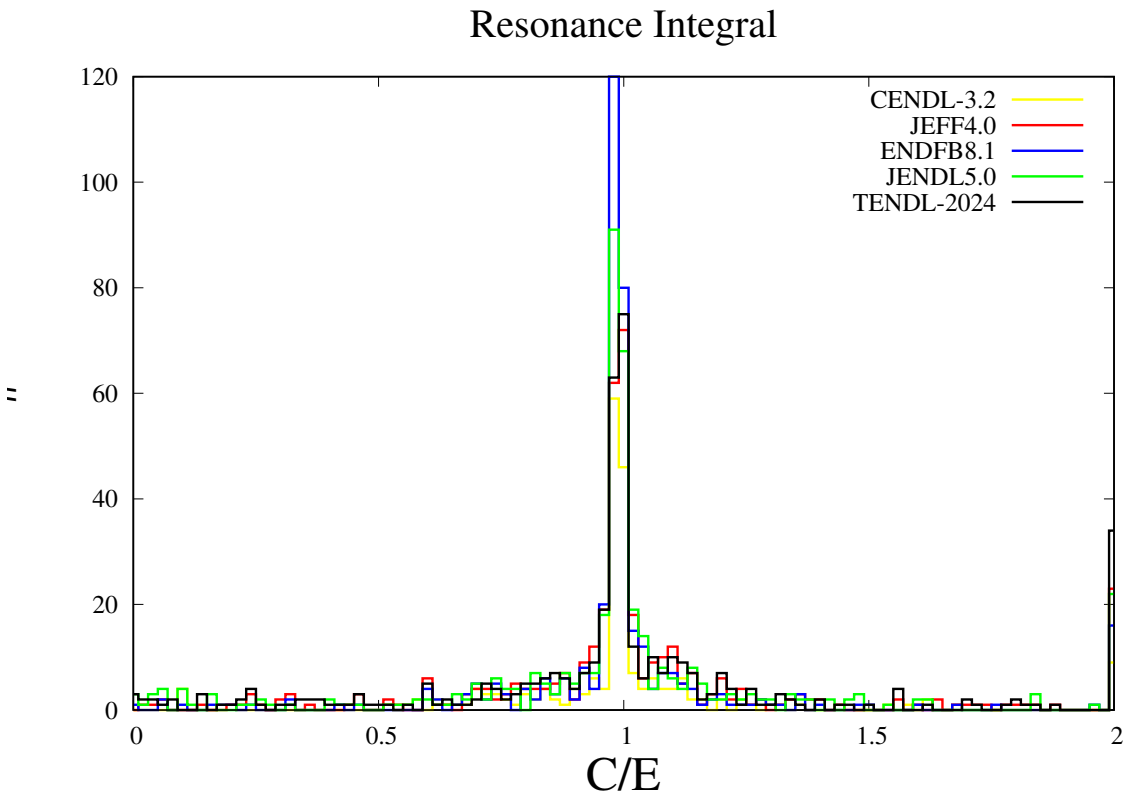

Figure 5.3: C/E histogram for Resonance Integral

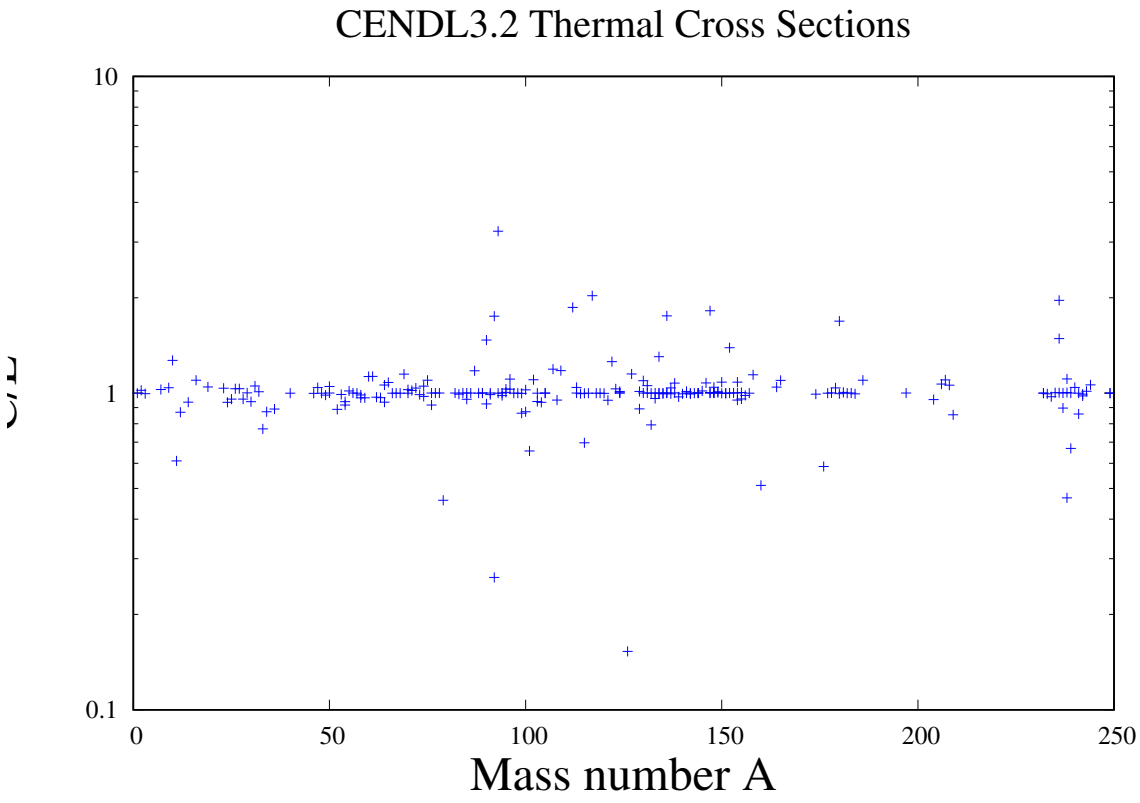

Figure 5.4: C/E values for thermal cross sections for CENDL-3.2

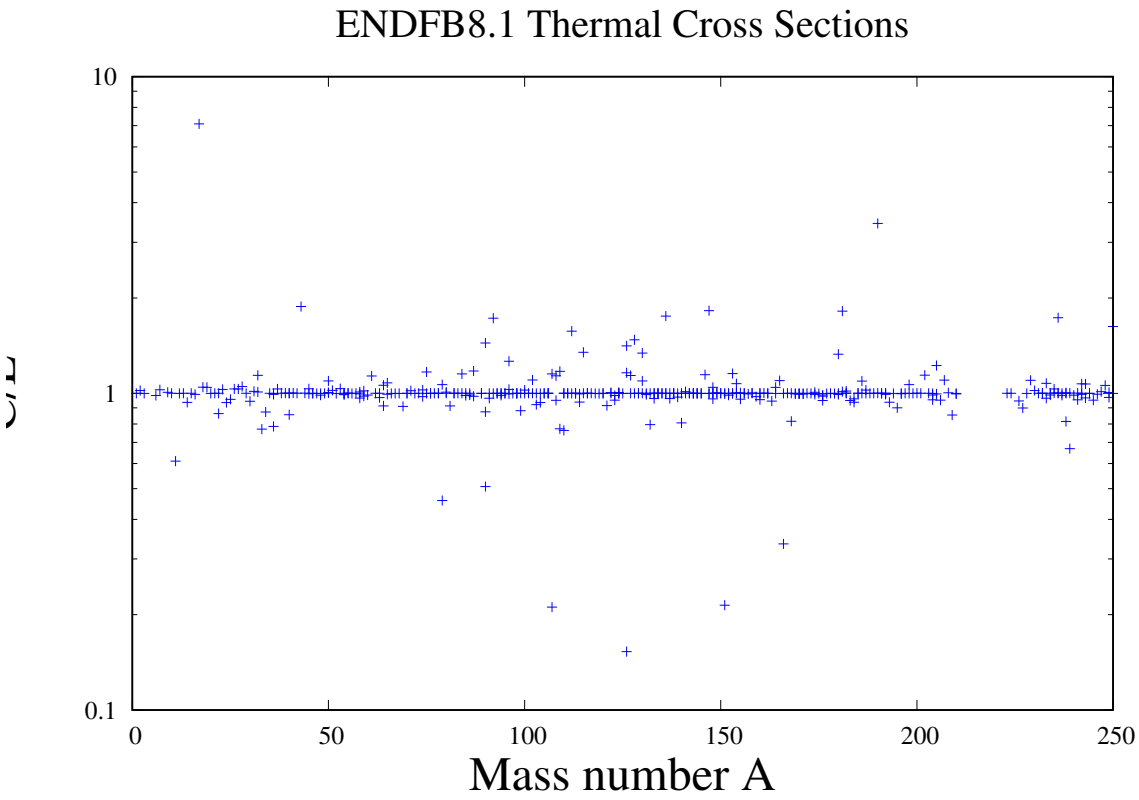

Figure 5.5: C/E values for thermal cross sections for ENDF/B-VIII.1

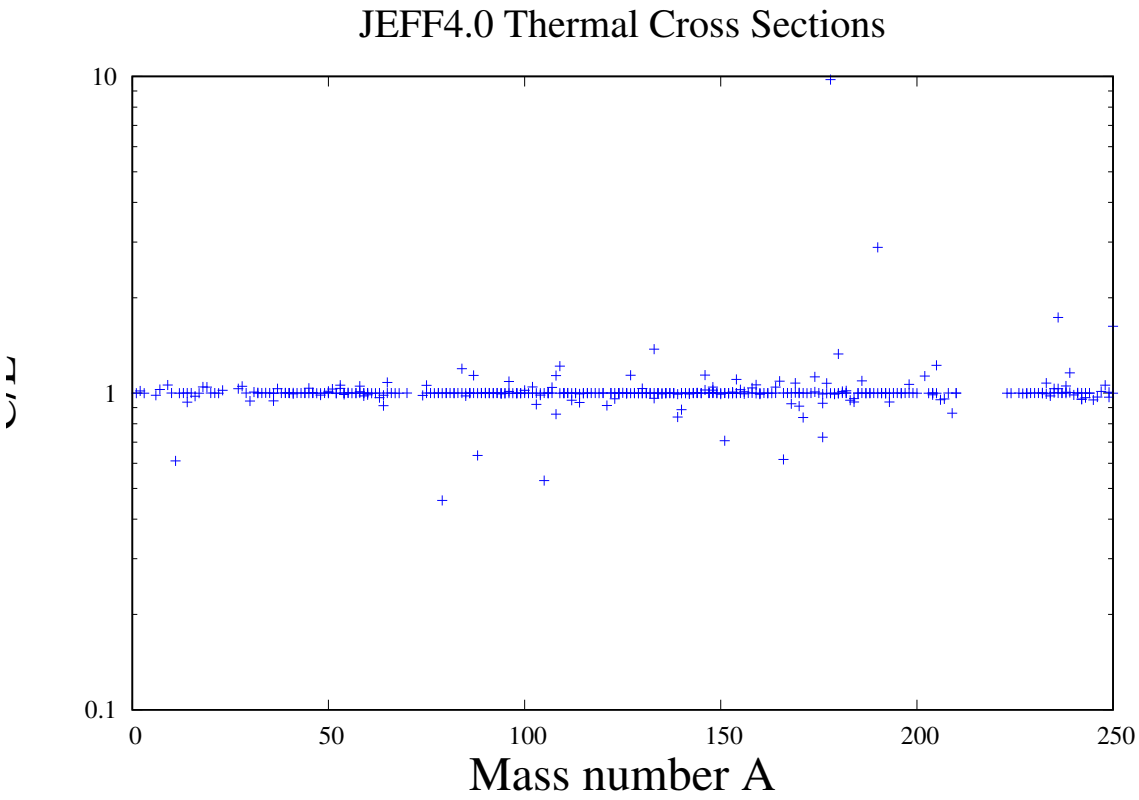

Figure 5.6: C/E values for thermal cross sections for JEFF-3.3

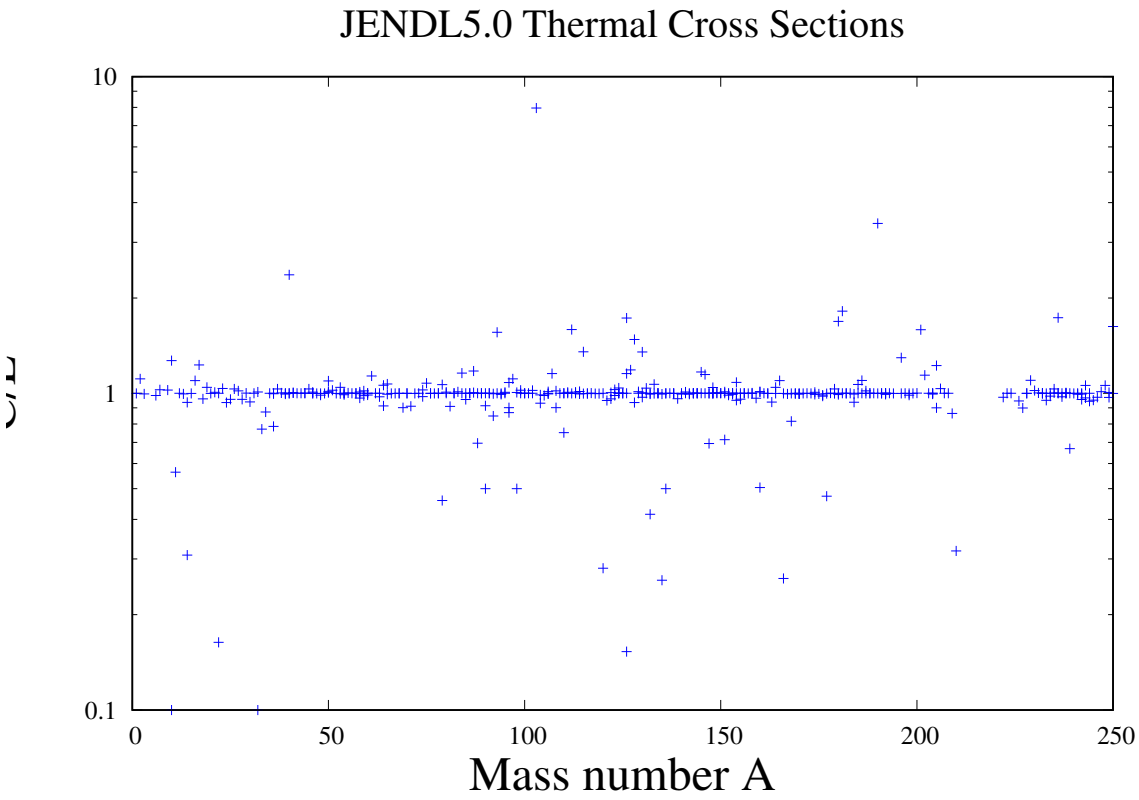

Figure 5.7: C/E values for thermal cross sections for JENDL-5.0

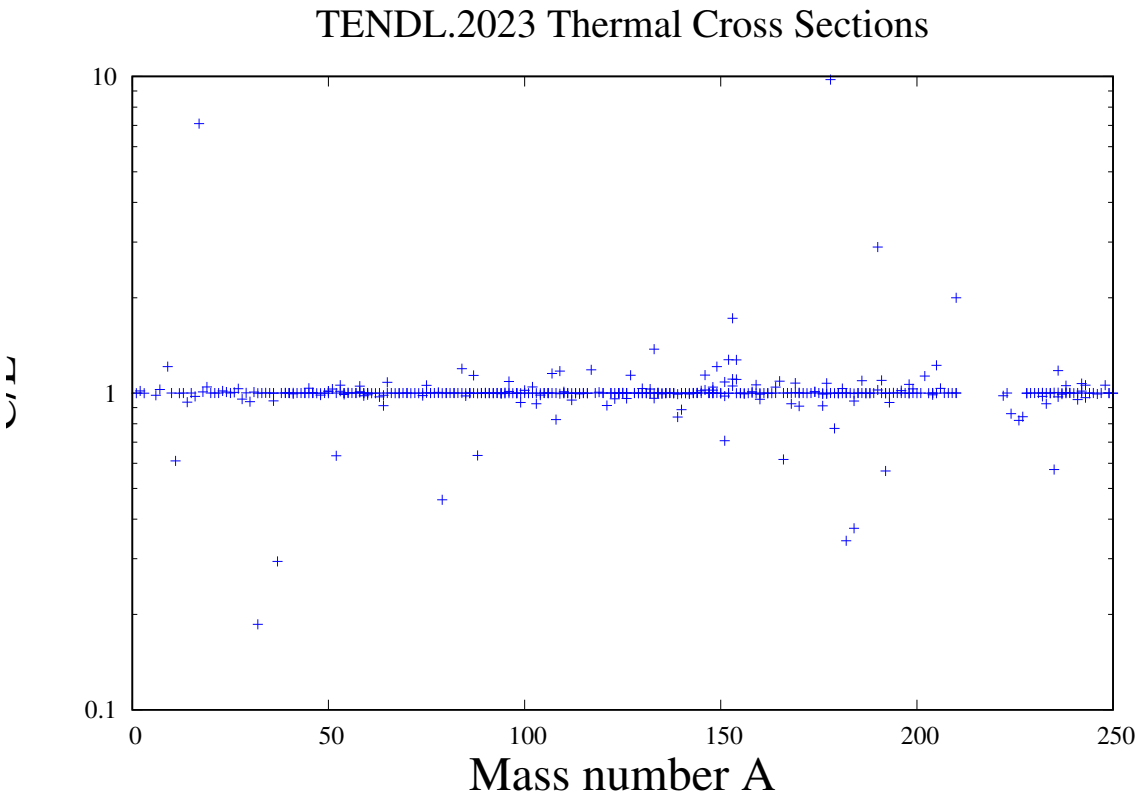

Figure 5.8: C/E values for thermal cross sections for TENDL-2023

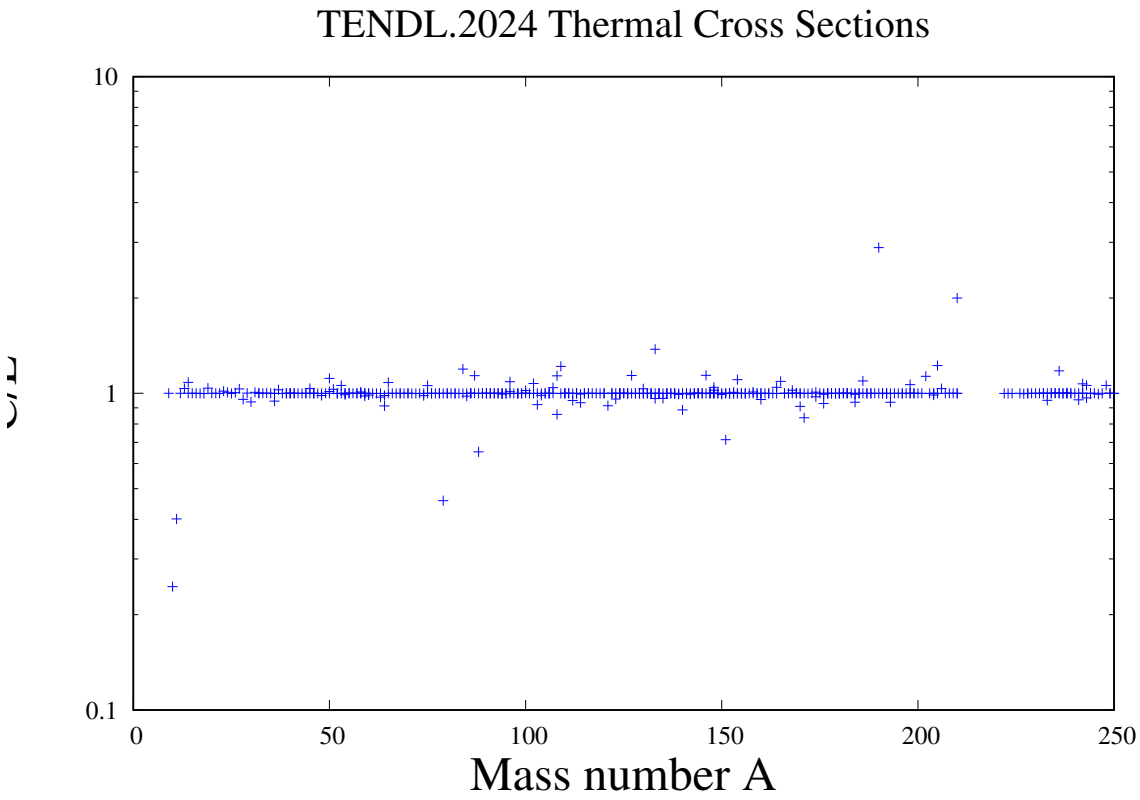

Figure 5.9: C/E values for thermal cross sections for TENDL-2024

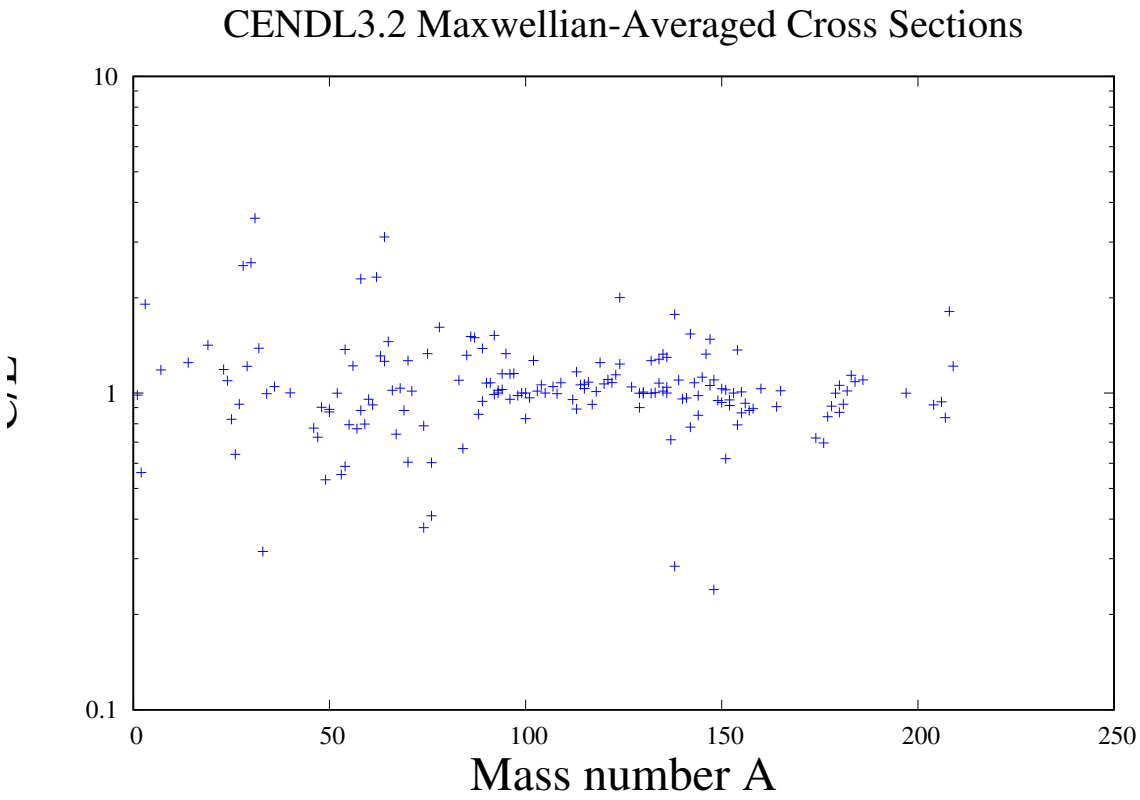

Figure 5.10: C/E values for MACS for CENDL-3.2

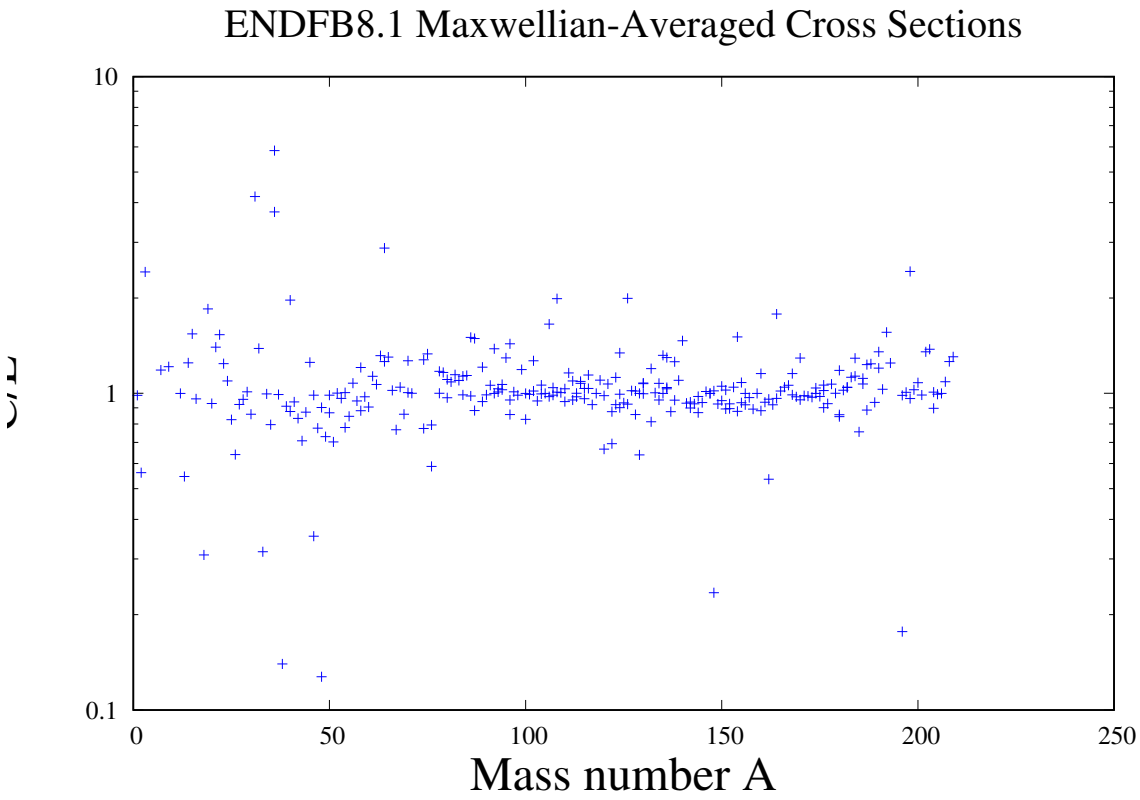

Figure 5.11: C/E values for MACS for ENDF/B-VIII.1

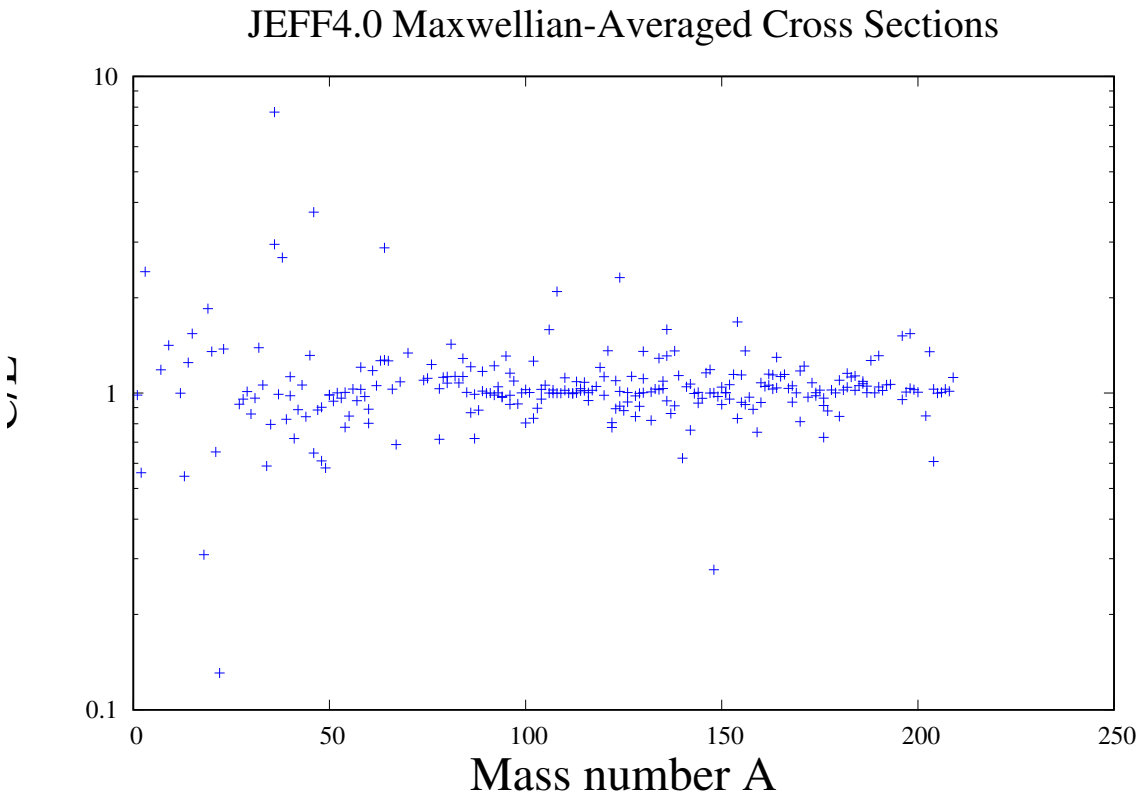

Figure 5.12: C/E values for MACS for JEFF-4.0

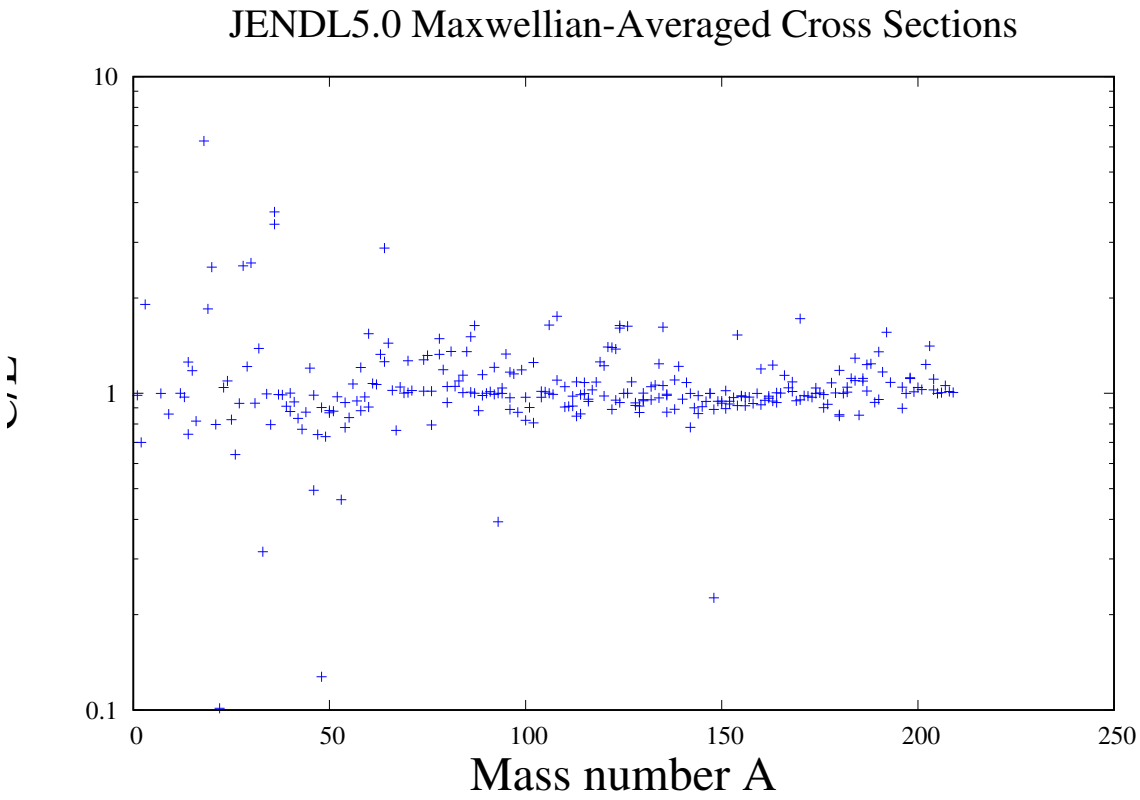

Figure 5.13: C/E values for MACS for JENDL-5.0

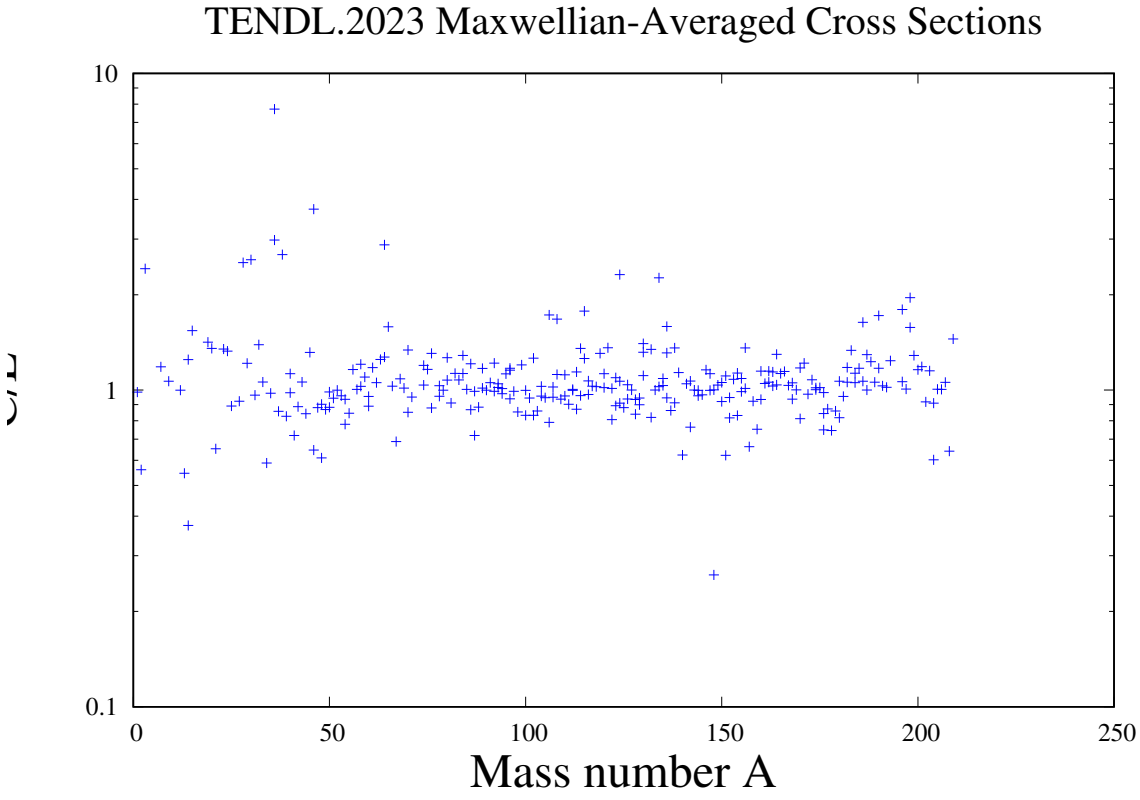

Figure 5.14: C/E values for MACS for TENDL-2023

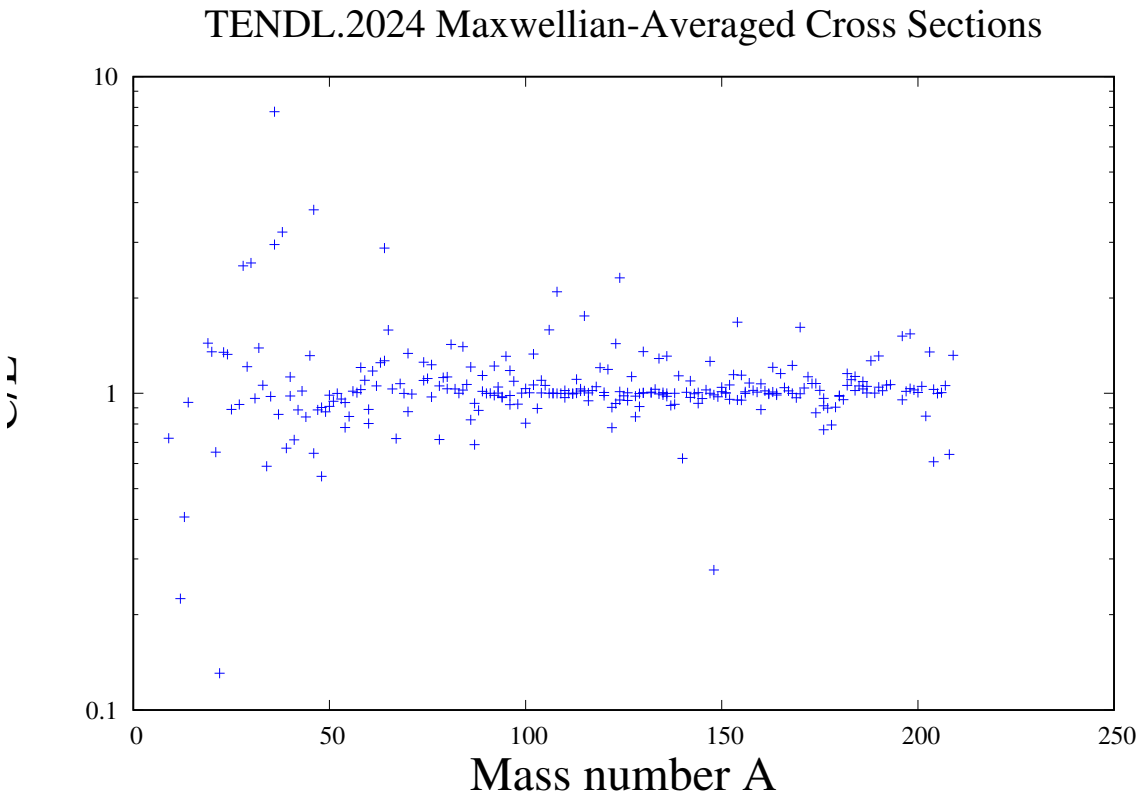

Figure 5.15: C/E values for MACS for TENDL-2023

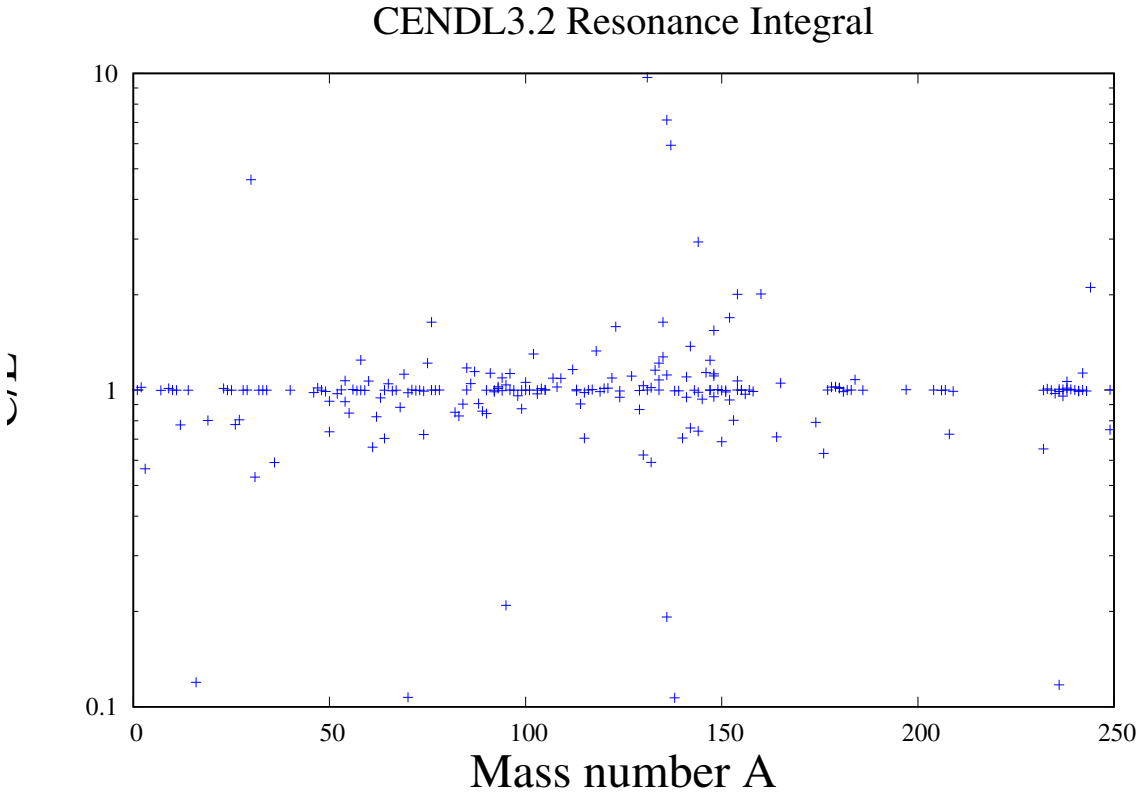

Figure 5.16: C/E values for resonance integrals for CENDL-3.2

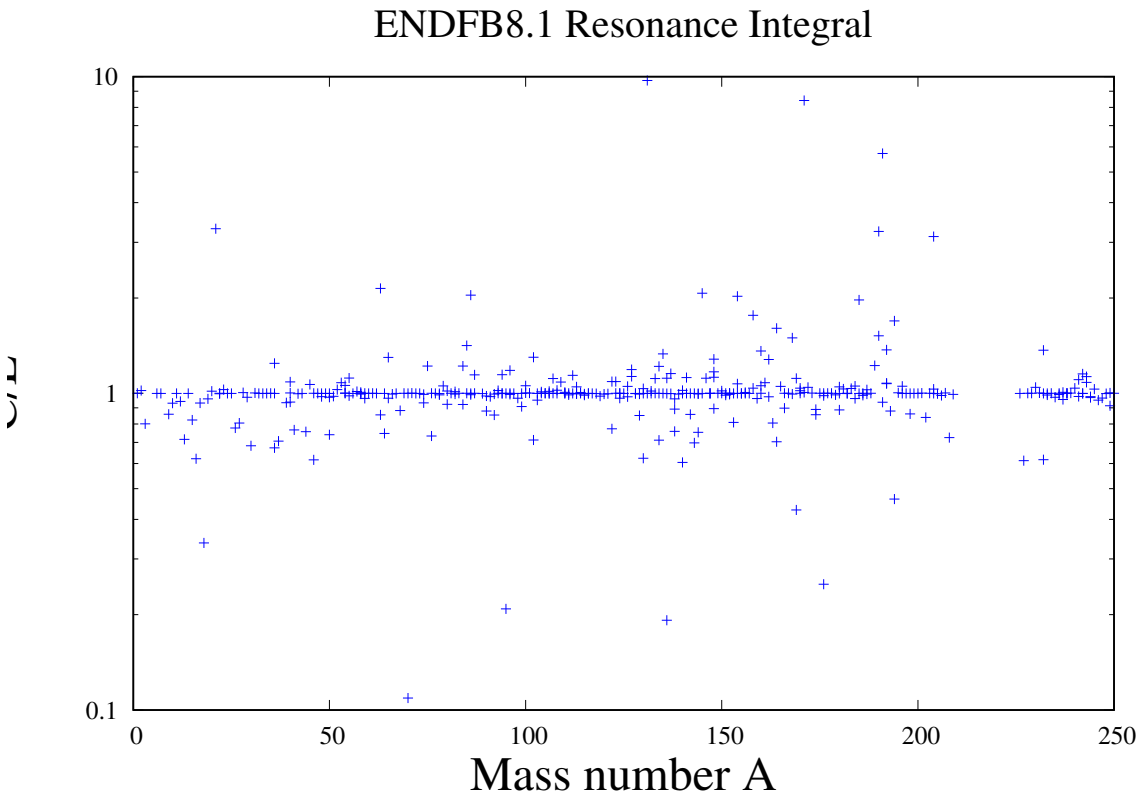

Figure 5.17: C/E values for resonance integrals for ENDF/B-VIII.1

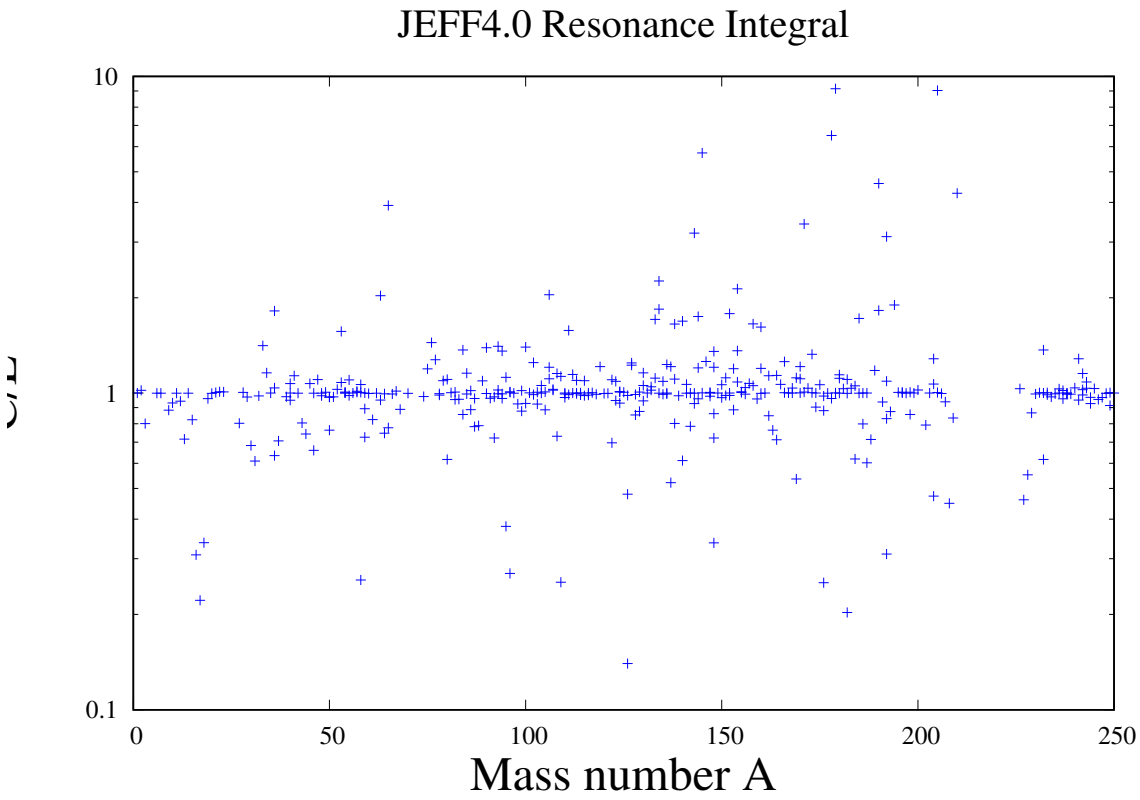

Figure 5.18: C/E values for resonance integrals for JEFF-4.0

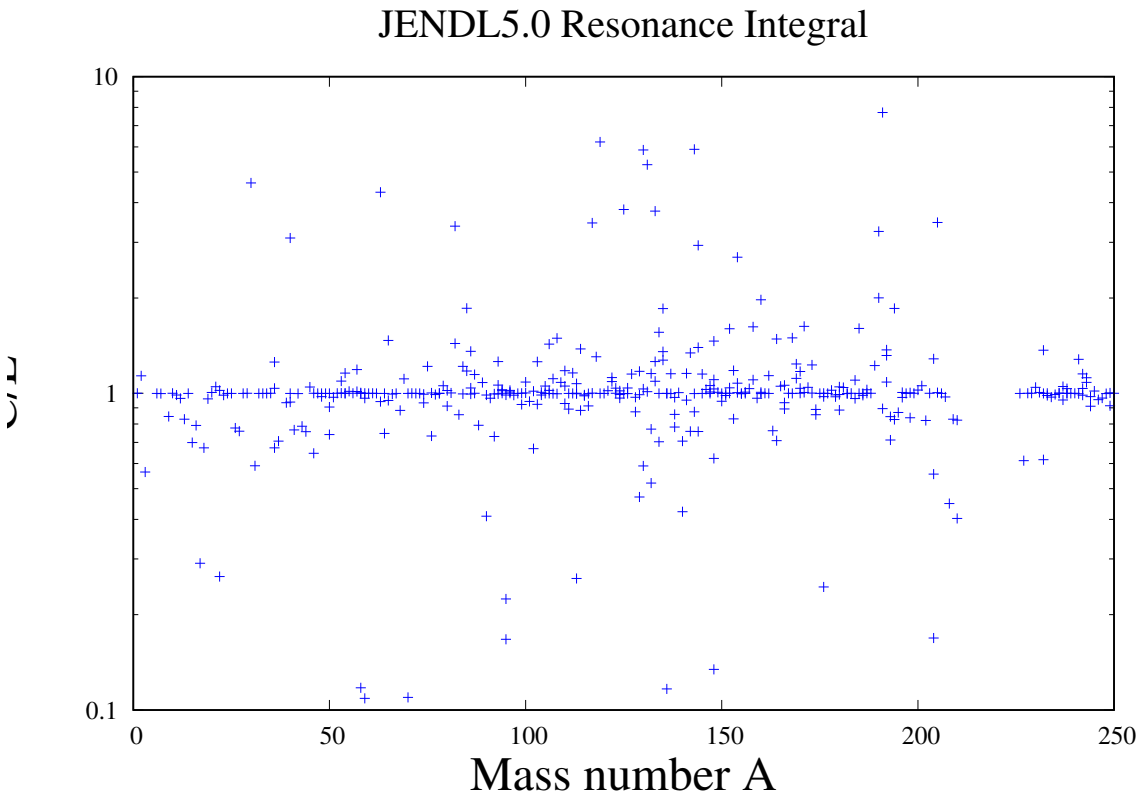

Figure 5.19: C/E values for resonance integrals for JENDL-5.0

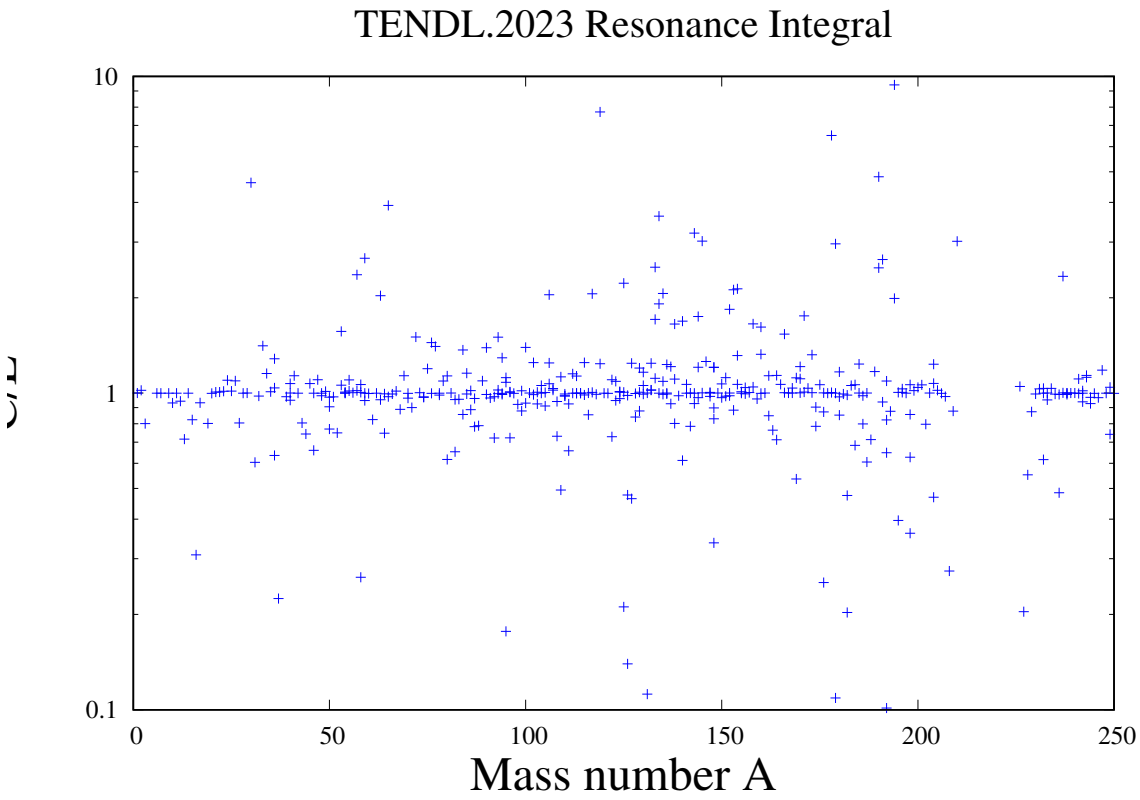

Figure 5.20: C/E values for resonance integrals for TENDL-2023

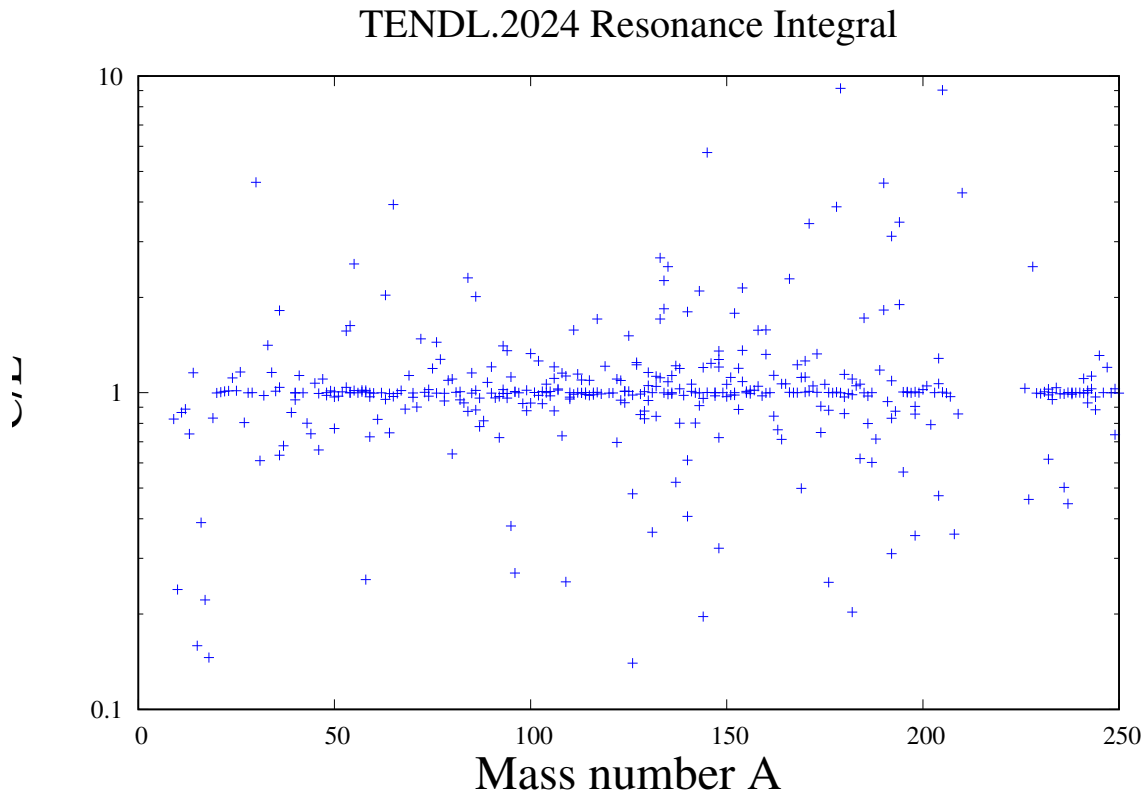

Figure 5.21: C/E values for resonance integrals for TENDL-2024

6. Outlook and conclusions

This manual describes RESONANCETABLES, a directory structured database derived from other databases, containing thermal cross sections, MACS and average resonance data. All data are presented in unified x-y-dy tables, and classified according to target nucleus and reaction.

This is supposed to be a first step towards a modern evaluation of these quantities whereby the selected databases are completely reproducible from the original data and are retrievable by API's. "First step" here means that ideally an API would be available which would retrieve these data according to user-specific requests on command line level. Instead we have here produced the entire database.

The current 'selected' databases have been constructed according to simple priority rules. It would be very useful to have an international working group to come to a well-evaluated database for these quantities, possible including the complete set of known resonance parameters.

Bibliography

[1] S. F. Mughabghab. *Atlas of Neutron Resonances*. 6th ed. The Netherlands: Elsevier, 2018.

[2] 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.

[3] S. F. Mughabghab. *Atlas of Neutron Resonances*. 5th ed. The Netherlands: Elsevier, 2018.

[4] S.I. Sukhoruchkin and Z.N. Soroko. *Neutron Resonance parameters*. 5th ed. Germany: Landolt-Bornstein, 2015.

[5] URL: http://www.kayzero.com/k0naa/k0naaorg/k0-ISC.html.

[6] R. B. Firestone. *Renormalization of Pile Oscillator Thermal Neutron Capture Cross Section Data*. Technical Report INDC(USA)-109. International Atomic Energy Agency, 2021. URL: https://www-nds.iaea.org/publications/indc/indc-usa-0109.pdf.

[7] J. Kopecky. *private communication*.

[8] I. Dillmann et al. "KADoNiS v0.3 - The third update of the Karlsruhe Astrophysical Database of Nucleosynthesis in Stars". In: *Proceeding of the workshop EFNUDAT Fast Neutrons - scientific workshop on neutron measurements, theory and applications*. Geel, Belgium, 2009.

[9] Diego Vescovi et al. "The ASTRAL database for neutron-capture nucleosynthesis studies". In: *EPJ Web of Conferences* 279 (2023), p. 11011. DOI: 10.1051/epjconf/202327911011.

[10] Z. Y. Bao et al. "Neutron cross sections for nucleosynthesis studies". In: *Atomic Data and Nuclear Data Tables* 76.1 (2000), pp. 70–154. DOI: 10.1006/adnd.2000.0833.

[11] N. Otuka et al. "Towards a more complete and accurate experimental nuclear reaction data library (EXFOR): International collaboration between nuclear reaction data centres (NRDC)". In: *Nuclear Data Sheets* 120 (2014), pp. 272–276.

[12] A.J. Koning. *EXFORTABLES-1.0: An experimental nuclear reaction database based on EXFOR*. Vol. IAEA-NDS-0235. 2020.

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

[14] D.A. Brown et al. “ENDF/B-VIII.0: The 8th Major Release of the Nuclear Reaction Data Library with CIELO-project Cross Sections, New Standards and Thermal Scattering Data”. In: *Nuclear Data Sheets* 148 (2018), p. 1.

[15] A. J. M. Plompen et al. “The joint evaluated fission and fusion nuclear data library, JEFF-3.3”. In: *European Physical Journal A* 56 (2020), p. 181.

[16] K. Shibata et al. “JENDL-4.0: A New Library for Nuclear Science and Engineering”. In: *Journal of Nuclear Science and Technology* 48.1 (2011), pp. 1–30.

[17] Z.G. Ge et al. “The Updated Version of Chinese Evaluated Nuclear Data Library (CENDL-3.1)”. In: *J. Kor. Phys. Soc.* 59 (2011), p. 1052.

[18] C. Dunford. *ENDF Utility Codes Release 6.10*. Tech. rep. IAEA report, IAEA-NDS-29. Brookhaven National Lab., 1995.

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

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, EXFOR or other basic nuclear data sources. 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. Also, it is used for a compilation of all resonance parameters. 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 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 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. We note that we only focus on the actual data, not the complete metadata of EXFOR, which can be obtained from the original EXFOR entries. (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.

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 from TALYS or ENDF 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 can be 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. It is not too difficult to remove the '#' and parse the above file to JSON with either your own script or, as we already verified, with an AI assistant.

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. Also, compiled data for e.g. thermal neutron cross sections, resonance parameters, Maxwellian-averaged cross sections etc. fit well into this schema.

A.2.1 header

All YANDF files start with the same keywords:

- **header**:
  - **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 often 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**:
  - **library**: one of the NDL's such as ENDFB8.1, JENDL5.0, JEFF3.3, TENDL-2023, 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**:
  - **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
  - **level energy [MeV]**: only for discrete levels: the level energy as given by EXFOR

A.2.4 target

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

- **target**:
  - **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**:
  - **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**:
  - **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
```