UCRL-SM-206606

User's Guide to `fete`: From ENDF/B-VI To ENDL

David Brown and Gerald Hedstrom
Lawrence Livermore National Laboratory, Livermore, CA 94550 USA

Tony Hill
Los Alamos National Laboratory, Los Alamos, NM USA
(Dated: September 17, 2004)

This guide describes how to run the `fete` ENDF/B-VI to ENDL translation code and outlines some of the general features of the translation process. In particular, this guide details how to install the code and supporting scripts, how to prepare an ENDF/B-VI formatted database for translation, how to translate the database, and how to check the translated database. This guide also explains how `fete` treats each ENDF/B-VI reaction and data type.

Contents

I. INTRODUCTION

This guide is a description of the LLNL Computational Nuclear Physics (CNP) Groups's "From ENDF/B-VI To ENDL" (a.k.a. `fete`) code. `fete` translates nuclear reaction data from the ENDF/B-VI to Livermore's internal ENDL [1] format. ENDF/B-VI is both an evaluated nuclear data library [2] and an international standard data format [3] for evaluated nuclear data. Because of the significant difference in both format and content, the translation process is a complicated multi-step process.

All of ENDL data is presented in point-wise linearly interpolated tables while ENDF/B-VI data is presented in a variety of manners, depending on the reaction and quantity in question. The two formats also differ in the arrangement of the data in the databases. Data in the ENDF/B-VI format are divided into materials (abbreviated MAT), which are further divided into "Material Files" (abbreviated MF). An MF is one type of reaction data (e.g. MF=3 is tabular cross section data). These MF's are subdivided into "Material Tapes" (abbreviated MT) which contain the data for individual reactions. Thus MT=16 MF=3 contains (n,2n) reaction cross section data while MT=18 MF=3 contains the total (n,f) reaction cross section. By contrast, ENDL requires

that reactions and nuclei be sorted into nesting directories first according to projectile (e.g. `yi01` means incident neutron), then nuclide (e.g. `za001002` means Z=1 and A=2 so we have deuterium), then reaction. The data for each reaction has several numbers associated with it, namely `yo` to denote the outgoing particles, C to denote reaction type and I to denote reaction property. The ENDL C and I numbers are analogous to the ENDF/B-VI MF and MT numbers, respectively. However, we caution that the mapping from (MF,MT) to (C,I) is not this straightforward as we see in Tables I and II.

In this note, we will first cover the installation and use of the translation codes and supporting scripts. Second, we will detail the translation process and highlight issues of which a user should be aware. Finally, we will discuss some of the outstanding issues in the translation: the extensions that ENDL will need in order to support the rest of the ENDF/B-VI format and some of the bugs found both in the format specifications and in the databases. This note also contains two appendices covering some of the issues in greater detail. The first appendix details the preparation of the databases for translation and the various preprocessing codes. This includes a discussion of the Doppler broadening performed at this step. The second appendix details some of the mathematical issues encountered in translating ENDF/B-VI double differential data. This note does not go into the detailed workings of the code. This is covered in the `fete` Code Reference [4] documentation that accompanies the source code distribution.

II. STEPS IN TRANSLATING

The entire ENDF/B-VI to ENDL translation process is summarized:

1. Install the translation code and scripts
2. Acquire and prepare a database
3. Translate the database
4. Compute energy depositions
5. Check results

In the following subsections, we will explore each step. We assume some familiarity with unix or related systems. This code system is known to work on Redhat Linux versions 8.0 and higher, MacOS X and higher (both using `gcc3.3`) and on Solaris (using `gcc2.95.3`).

A. Installing the translation code & scripts

1. *Getting the translation code*

`fete` is available either as a tarball on Livermore's CNP Group web page [5] or internally on the CNP Group's CVS. The CVS is located at `ndg.llnl.gov:/cvsroot/` and this project's name is `fete`. The tarball unpacks to a directory called `fete/`.

2. *Building the codes*

To build the package, enter the main project directory (`fete/`) and type "`make`." If compilation went well, an executable file called `fete` will be left in the `src/` directory. We recommend moving this executable file as well as the contents of the `scripts/` directory to some place in your environment's path. Our preference is to put all of the executable files in `$NDSPATH/bin/` directory. Here `$NDSPATH` is an environment variable that points to a directory containing this project as well as the other code projects needed for nuclear data processing.

3. *The* `bdfls` *file*

The `bdfls` file contains a variety of data required by several of the codes maintained by the CNP group. In particular, the `bdfls` file contains group boundaries, fluxes, nuclear masses, half-lives, various physical and conversion constants, and atomic excitation levels. `fete`, `endep` [6] and `fudge` [7] all require masses.

A copy of the `bdfls` file is included in the `src/` directory of the `fete` package. You will need to set the `$BDFLSPATH` environment variable to point to the `bdfls` file. If you do not set this variable, `fete` will attempt to load the `bdfls` file from the '.' directory. Furthermore, most of the supporting scripts will fail without this environment variable.

4. *Setting up the environment*

You will need several supporting codes in order to complete the translation:

1. `prepro2000` – if preprocessing a database yourself
2. `endfsplit` – if preprocessing the ENDF/B-VI database yourself
3. `endep` – to compute energy depositions
4. `fudge` – for checking the translation results
5. `PERL`, `Python`, `csh`, a `c++` compiler, `make`

MF	Description	I
1	General information	n/a[a]
2	Resonance parameter data	0[b]
3	Reaction cross sections	0
4	Angular distributions for massive emitted particles	1
5	Energy distributions for massive emitted particles	4
6	Energy-angle distributions for emitted particles (γ's or massive particles)	3 or 4
7	Thermal neutron scattering law data	n/a[c]
8	Radioactivity and fission-product yield data	n/a[d]
9	Multiplicities for radioactive nuclide production	n/a[d]
10	Cross sections for radioactive nuclide production	n/a[d]
12	Multiplicities for photon production	9
13	Cross sections for photon production	0
14	Angular distributions for photon production	4[e]
15	Energy distributions for photon production	4
23	Photo-atomic interaction cross sections	0
27	Atomic form factors or scattering functions for photo-atomic interactions	941
30	Data Covariances obtained from parameter covariances and sensitivities	n/a
31	Data covariances for nubar	n/a
32	Data covariances for resonance parameters	n/a
33	Data covariances for reaction cross sections	n/a
34	Data covariances for angular distributions	n/a
35	Data covariances for energy distributions	n/a
39	Data covariances for radionuclide production yields	n/a
40	Data covariances for radionuclide production cross sections	n/a

[a]MT=451 delayed neutron data is handled separately.
[b]Translated into pointwise data using Red Cullen's `recent` code.
[c]Not implemented in ENDL yet.
[d]Use ACTL for this data type.
[e]Only if not isotropic; if isotropic, this file is ignored since the ENDL default is isotropic.

TABLE I: ENDFB-VI MF numbers and their rough ENDL equivalent.

Information on where and how to obtain these codes can be found on the Computational Nuclear Physics Group's web page [5]. Getting these codes installed and working is your own responsibility, however we provide details on the splitting and preprocessing in Appendix A.

The various control scripts require that you set several environment variables (`$BDFLSPATH`, `$ENDLPATH`, `$FUDGEPATH`), add the path to the `fudge` source to your `$PYTHONPATH` and add the path to all of the executables produced above to your `$PATH`. If you are using `csh`, you may accomplish all of this by adding the following lines (appropriately modified, of course) to your `.cshrc` file:

```
#
# Nuclear Data Processing System setup
#
  setenv NDSPATH /home/joeuser/NDS
  setenv BDFLSPATH $NDSPATH/bdfls
  setenv ENDLPATH $NDSPATH/databases
  setenv FUDGEPATH $NDSPATH/fudge
  setenv PYTHONPATH "$FUDGEPATH/Src:$PYTHONPATH"
  set path = ( $NDSPATH/bin $path )
```

B. Getting and preparing the database

1. *Getting a database from the web*

There are several modern databases that use the ENDF/B-VI format: ENDF/B-VI, JENDL-3.3, JEFF-3.0, BROND-2.2, and CENDL-2.1. ENDF/B-VI [2], JENDL-3.3 [8], and JEFF-3.0 [9] are the most up to date and are the main targets of our translation effort. All three of these databases are available on the web, both in raw and point-wise (i.e. preprocessed) form. Since the preprocessing stage can take quite a bit of time (several hours to a day or more, depending on your hardware), we recommend downloading the preprocessed data. ENDL requires that the data be Doppler broadened to 300 de-

MT	Description	C
1	(n,total)	1
2	(z,elastic)	10
3	(z,non-elastic)	55[a]
4	(z,n)	11[b]
5	(z,anything)	n/a[c]
10	(z,continuum)	n/a[c]
11	(n,2nd)	32
16	(z,2n)	12
17	(z,3n)	13
18	(z,f)	15
19	(n,f) – 1^{st} chance fission.	n/a[d]
20	(n,nf) – 2^{nd} chance fission.	n/a[d]
21	(n,2nf) – 3^{rd} chance fission.	n/a[d]
22	(z,nα)	26
23	(z,n3α)	n/a
24	(z,2nα)	33
25	(z,3nα)	16[e]
27	(n,abs)	n/a[c]
28	(z,np)	20,21[f]
29	(z,n2α)	27
30	(z,2n2α)	n/a
32	(z,nd)	22
33	(z,nt)	24
34	(z,n^{3}He)	25
35	(z,nd2α)	n/a
36	(z,nt2α)	n/a
37	(z,4n)	14
38	(n,3nf) – 4^{th} chance fission.	n/a[d]
41	(z,2np)	29
42	(z,3np)	16
44	(n,n2p)	17
45	(n,npα)	34
50	(z,n[0]) – excitation of g.s.	11[b,g]
51-90	(z,n[n]) – excitation of nth excited state	11[b,g]
91	(z,n[c]) – excitation to continuum	11[b,g]
101	(n,disappearance)	n/a[c]
102	(z,γ)	46
103	(z,p)	40
104	(z,d)	41
105	(z,t)	42
106	(z,^{3}He)	44
107	(z,α)	45
108	(z,2α)	37
109	(z,3α)	n/a
111	(z,2p)	18
112	(z,pα)	48
113	(z,t2α)	42[h]
114	(z,d2α)	n/a
115	(z,pd)	19
116	(z,pt)	39
151	(n,resonance)	n/a
201	(z,Xn)	56[i]
202	(z,Xγ)	55[i]
203	(z,Xp)	50[i]
204	(z,Xd)	51[i]
205	(z,Xt)	52[i]
206	(z,X^{3}He)	53[i]
207	(z,Xα)	54[i]
208-218	Various meson and antiparticle production σ's	n/a
251-253	Various elastic neutrons scattering parameters.	n/a
301	Energy release for total and partial σ's.	n/a
451	Heading or title information (MF=1 only).	n/a[j]
452	(z,f) $\bar{\nu}$ total (prompt plus delayed) fission.	15[k]
454	(z,f) Independent fission product yields.	15[k]
455	(z,f) $\bar{\nu}$ for delayed fission.	15[k]
456	(z,f) $\bar{\nu}$ for prompt fission.	15[k]
457	(z,f) Radioactive decay data.	n/a
458	(z,f) Energy release in fission for incident n's.	15[k]
459	(z,f) Cumulative fission product yields.	15[k]
500	Total charged particle stopping power.	n/a
501	Total photon interaction σ.	70
502	Photon coherent scattering.	71
504	Photon incoherent scattering.	72
505	Imaginary scattering factor.	n/a
506	Real scattering factor.	n/a
515	Pair production, electron field.	n/a
516	Pair production.	74
517	Pair production, nuclear field.	n/a
522	Photoelectric absorption.	73
534-572	Various subshell photoelectric σ's.	n/a
600	(z,p[0])	40[i]
601-648	(z,p[n])	40[i]
649	(z,p[c])	40[i]
650	(z,d[0])	41[i]
651-698	(z,d[n])	41[i]
699	(z,d[c])	41[i]
700	(z,t[0])	42[i]
701-748	(z,t[n])	42[i]
749	(z,t[c])	42[i]
750	(z,^{3}He[0])	44[i]
751-798	(z,^{3}He[n])	44[i]
799	(z,^{3}He[c])	44[i]
800	(z,α[0])	45[i]
801-848	(z,α[n])	45[i]
849	(z,α[c])	45[i]
851	Lumped reaction covariances.	n/a

[a] If MF=12–15, then the outgoing γ's are lumped in the C=55 file, otherwise the data is discarded since it should be stored with individual reactions. The non-γ data is discarded since it is purely derived data.
[b] If the `sum_inelastic` option is set then use the MT=4 data, otherwise use the MT=51–91 data (lumping all γ data into the C=55 file).
[c] Purely a derived file, so it is not needed.
[d] Only total fission data is stored in ENDL.
[e] Below $E_{\rm inc} = 20$ MeV, only ^{7}Li has this reaction and $^7\mathrm{Li}(n,3np)\alpha \equiv {}^7\mathrm{Li}(n,3n\alpha)p$
[f] ENDF/B-VI does not distinguish whether the proton or neutron is emitted first.
[g] Level excitation functions stored in S=1 type files in ENDL.
[h] Below $E_{\rm inc} = 20$ MeV, only ^{10}B has this reaction and $^{10}\mathrm{B}+n \rightarrow t+{}^8\mathrm{Be}^* \rightarrow t+2\alpha$
[i] A slot exists in ENDL but this data is not translated.
[j] ENDL does not yet have a slot for documentation.
[k] Delayed fission information is stored in S=7 type files in ENDL.

TABLE II: ENDFB-VI MT numbers and their rough ENDL equivalent.

Database Name	CVS Project Name	Preprocessed version in CVS?
ENDF/B-V	`ENDF_tapes/ENDF.B-V`	N
ENDF/B-VI	`ENDF_tapes/ENDF.B-VI`	N
JEFF-3.0	`JEFF-3.0`	Y
JENDL-3.3	`JENDL-3.3`	Y

TABLE III: Databases in `ndg.llnl.gov:/cvsroot/`.

grees Kelvin, so care must be take to download the correct database.

2. Getting a database from the CVS

Locally, we store preprocessed and unprocessed data in our CVS repository. At the time of this writing, the repository is located at: `ndg.llnl.gov:/cvsroot/`. Table III lists the databases and whether a preprocessed version of the data is checked into our CVS.

3. Database preparation

The ENDF/B-VI format specification makes no requirement as to how the data is arranged in the library, so each nuclear data library is shipped in a different form. In the case of the ENDF/B-VI database, all of the data is lumped into "tapes," hence the need for "splitting" the database with the `endfsplit` code discussed in Appendix A 1) and then preprocessing. JEFF-3.0 and JENDL-3.3 require unpacking and directory renaming. In the end, we want the data in something like the ENDL standard directory layout [1]:

```
my_db/yi01/za001001/endf.orig
          /za001002/endf.orig
                .
                .
                .
```

Here, the `endf.orig` file is the raw, unprocessed evaluation for one nuclide.

Once this is done, we must preprocess each `orig` file to produce a `prepro` file. The `prepro` file differs from the original file in that:

- All resonances are expanded into a linear-linear interpolable form
- All cross sections are heated to 300 K

This is mainly an automated process using the `prepro2000.csh` script which controls Red Cullen's `linear`, `recent` and `sigma1` codes [10]. This step is detailed in Appendix A 2. After this step, the database will appear as:

```
my_db/yi01/za001001/endf.orig
              "    /endf.prepro
          /za001002/endf.orig
              "    /endf.prepro
                .
                .
                .
```

4. The `mini-db` utility

The `mini-db` utility is a simple script that copies out a subset of targets from one database to another. This is extremely useful for troubleshooting problem nuclides. The utility is invoked with:

```
mini-db <isotope file> <output db> \
    [master copy]
```

Here `[master copy]` is an optional argument and defaults to `$NDSPATH/databases/ENDFB6_prepro/`. The required argument `<isotope file>` is a plain text file containing a list of `za` numbers. The other argument `<output db>` is the uppermost directory of the new, miniature, database you are creating. A subdirectory of `<output db>` called `yi01/` containing the `za` files will be created by the script.

C. Translating the database & computing energy deposition

Translation is very simple: go to the `yi01/` directory, containing all of the `za` subdirectories, and execute `translate_all`. Provided everything is working, you will have a fully translated database in those `za` directories, complete with energy depositions. Your database directory should look something like this:

```
my_db/yi01/za001001/endf.orig
              "    /endf.prepro
              "    /endep_csh.log
              "    /endep.log
              "    /translate_csh.log
              "    /translation.log
              "    /yo00c01i000s000
              "    /yo00c10i000s000
                .
                .
                .
          /za001002/endf.orig
              "    /endf.prepro
                .
                .
                .
```

The next few subsections explain the `translate` and `translate_all` scripts and how to invoke the `fete` code directly. The translation may be "tweaked" using the `fete_options.inp` file and this mentioned below and is discussed in more detail in Section III.

1. `translate` *vs.* `translate_all`

The `translate_all` loops through a set of nuclei and translates them all using `translate`. It may be invoked using

```
translate_all [directory]
```

where `[directory]` is an optional argument that specifies the directory containing the `za` subdirectories. If this argument is missing, the script defaults to processing the current directory.

By contrast, `translate` script automates the translation of a single nuclide and is invoked via

```
translate <za number>
```

This script first invokes `fete` to translate the nuclide's data into ENDL format then it sets up and invokes the `endep.csh` script. The `endep.csh` is the piece of code that actually calls `endep` to compute energy depositions.

2. *Invoking* `fete` *from the command line*

Although we usually invoke `fete` through either the `translate` or `translate_all` scripts, you may wish to invoke `fete` directly on the command line via:

```
fete [-f options_file] [options]
```

`fete` expects that you invoke it in a `za` subdirectory and that an `endf.prepro` file exists in this directory. There are various options that control how the translation proceeds and they can be overridden in the `fete_options.inp` file. These options are described in Section III and listed in Table IV.

The `-f` option allows you to override the name of the `fete_options.inp` file. The options in the `fete_options.inp` file may also be overridden on the command line by prepending a "`-`" to the option, e.g.

```
fete -tol_1d 0.001 -mf5_tol 0.000025
```

D. Checking the translation

We provide some tools for checking and examining the translation. The first, `build_errlist.py`, builds a webpage summary of the translation errors. The second, `view_all.py`, allows you to plot all of the quantities in the newly created ENDL formatted database. The last two utilities, `za_comp` and `yo_comp`, are useful for comparing the contents of ENDL databases and individual `za` directories.

1. *The* `build_errlist.py` *utility*

The `Python` script `build_errlist.py` is a tool to build an HTML-formatted report of all errors encountered in translating a database. It requires the `fudge` package [7] be installed to function and it is invoked with

```
build_errlist.py <database>
```

where `<database>` is the name of the directory containing the database to be checked. This script combs the log files produced during the translation for errors in the translation stage and the energy deposition stage. This script also runs the `fudge` script `check.py` to check for proper ENDL formatting. When finished, an HTML-formatted report will be generated in the `<database>` directory, in the same level as the `yi01/` directory. To browse this report, point your web browser to the `translation_summary.html` file.

The `translation_summary.html` file contains two sections. The first is a hyper-linked table of nuclides and problems encountered during their processing. This table is linked to a page for each nucleus containing copies of the log-files containing details of the problem. The second section is a bulleted list of errors encountered in the translation stage, each with a list of the nuclei that encountered that error.

2. *The* `view_all.py` *utility*

`view_all.py` is a tool that uses `fudge` to plot every dataset in an ENDL formatted database. To use it, type

```
view_all.py <za> [database]
```

where `<za>` is the `za` number of interest and `[database]` is an optional argument for the database containing `<za>`. If this argument is skipped, `view_all.py` will attempt to look for `<za>` in the current directory.

3. *The* `za_comp` *and* `yo_comp` *utilities*

These two utilities allow you to compare the contents of two databases or two individual `za/` subdirectories. To compare two databases, do:

```
za_comp <db1> <db2>
```

To compare the contents of two `za/` subdirectories, do:

```
yo_comp <za1> <za2>
```

III. TRANSLATION DETAILS

In this section, we discuss what is involved in the actual translation. We start with the treatment of reaction cross sections. Then we discuss angle, energy, and joint energy-angle distributions for massive particles (not gammas). Following this, we discuss gamma data and we conclude with a section on delayed fission neutrons. You are encouraged to consult Tables I and II while reading this section.

Key	Value	Description
Interpolation Options		
`tol_1d`	0.005	Relative accuracy of 1-d list interpolation
`cut_off_1d`	0.001	Suspend `tol_1d` below this absolute value (arb. units)[a]
`tol_2d`	0.015	Relative accuracy of 2-d list interpolation
`cut_off_2d`	0.003	Suspend `tol_2d below` this absolute value (arb. units)[a]
`dE_tol`	0.01	Maximum incident energy difference for 2-d lists (MeV)
`log_log_tol`	0.005	Relative error in converting log-log to lin-lin
3d List Options		
`num_mu_3d`	12	2*`num_mu_3d+1` = number of μ bins in 3d expansion
`num_E_in_3d`	12	Number of incident energies for 3d expansion
MF=5 Options (see Section III B 2)		
`num_mf5_bins`	32	Number of equiprobable energy bins
`mf5_shift`	0.00002	Amount by which the first distribution is above threshold[b]
`mf5_tol`	0.00001	Energy tolerance in ENDF/B-VI files (MeV)
Other Options For Specific Types Of Data		
`sum_inelastic`	0	If = 0, use the MT=51–91 individual inelastic data[c]; otherwise, use the MT=4 total inelastic data
`Kalbach_i10`	0	If > 0, calculate the I=10 average energy (see Section III B 3)
`split_gammas`	0	If > 0, split discrete gammas from continuum using S=3 (see Section III C)
`fission_Q`	1	If > 0, use ENDF fission Q value, otherwise subtract Q for delayed fission decays (see Section III D)
General Processing Options		
`skip_date`	1	If > 0, don't bother processing evaluation's date field
`message_level`	0	If = 0, all messages reported; if ≤ 1, `Info` and `Unimplemented` messages are not reported but `Warning` and `SevereError` messages are reported; if > 1, only `SevereError` messages are reported.
`endl_datafield_precision`	7	Precision saved in ENDL files
`max_energy`	20.0	Maximum energy for ENDL files (MeV)
`max_list_size`	500	Maximum length of a 1-d list

[a] For cross section data, this has units of barns; for multiplicities and probabilities, this has no units.
[b] $E_{\rm shift} = E_{\rm thresh}(1 +$ `mf5_shift`$)$.
[c] The accompanying γ data is lumped into the C=55 file.

TABLE IV: Default parameters for the `fete` code. The parameter names are not case-sensitive.

A. General information on the reaction data in ENDF/B-VI.

1. MF=1 (Documentation)

The MF=1 information describes the contents of the file, and aside from the MT=455 file, we don't translate it. The treatment of delayed fission neutron information is detailed in Subsection III D.

2. MF=2 (Resonance Data)

The MF=2 data gives the resonance parameters. We preprocess this data with Red Cullen's `recent` code to convert this information to tabular cross sections [10]. This table is combined with the MF=3 cross section data to form the actual cross section for the reaction.

3. MF=3 (Cross Section Data)

The MF=3 file gives tabular cross section data. As a second preprocessing step, we use Red Cullen's `sigma1` code to Doppler broaden the data to room temperature. The corresponding ENDL file is flagged by I=0.

B. Data on massive particles, ordered by MF number.

1. MF=4 (Outgoing Angular Distributions)

The MF=4 file gives angular distributions for outgoing particles. This is most commonly used for discrete 2-body reactions and it consists of coefficients for Legendre expansions in the center-of-mass frame. ENDL also uses the center-of-mass frame for such data, but it only has a piecewise-linear representation. We approximate the Legendre expansions by piecewise-linear functions. The corresponding ENDL file is flagged by I=1. Note that the sum of the Legendre expansion is sometimes negative for certain outgoing angles and when this happens, we replace these unphysical values by zero.

Interpolation of this data with respect to energy of the incident neutron is usually straightforward, since linear interpolation of Legendre coefficients is equivalent to linear interpolation of the sum of the Legendre series. For some targets, e.g. ^{238}Pu (za094238), ENDF/B-VI uses log-linear interpolationfor the Legendre coefficients. We expand the data to linear-linear interpolation.

In some cases the MF=4 data consists of tabular angular distributions in the laboratory frame. In these cases energy distributions are also given and the two distributions are regarded as uncorrelated. The translation of this data to ENDL is no problem because ENDL uses the same conventions. Such tabular data in MF=4 is "archival data," in the sense that new evaluations for ENDF/B-VI are supposed to provide double differential (joint energy-angle) distributions.

2. MF=5 (Outgoing Energy Distributions)

The MF=5 file contains energy distributions of the outgoing particles. This data is often in laboratory coordinates as is the ENDL equivalent data (I=4). When the data is presented as an interpolatable table, the translation to ENDL format is trivial.

Unfortunately, it is more common for ENDF/B-VI MF=5 data to be given as a table of parameters to use in some formula. Thus, the energy distribution may be given by a single temperature evaporation model:

$$CE' \exp[-E'/\Theta]$$

for $0 \le E' \le E'_{\max}$. Here, E' is the energy of the outgoing particle, and values of Θ (the effective evaporation temperature in MeV) are given in the MF=5 file for various values of the energy of the incident neutron. The value of $E'_{\max}$ is determined by energy conservation, and the constant C is chosen to make the total probability be one. We expand this a formula into a piecewise-linear interpolation table, accurate to an amount specified by `mf5_tol`.

The translation code has a cutoff, `cut_off_1d`, so that we do not impose accurate piecewise-linear interpolation in the exponentially small tail. Specifically, if the energy distribution is denoted by $p = f(E')$ and if $p_{\max}$ is the maximum value of p, then accurate interpolation is not enforced when $f(E') <$ `cut_off_1d` $* p_{\max}$.

What makes translation of this functional ENDF/B-VI data to ENDL format difficult is the interpolation with respect to the energy of the incident neutron. The usual prescription in ENDF/B-VI is to derive such energy distributions by linear interpolation of the values of Θ. Some ENDL processing codes (e.g. `ndfgen` code [11]) use unit-base interpolation while others (e.g. `mcapm` [12]) use equally-probable energy bins, a mathematically similar procedure.

We have found that the unit-base interpolation scheme is usually not compatible with the linear interpolation of the function parameters, e.g. Θ, used by ENDF/B-VI. We therefore chose to ensure that linear interpolation of the equally-probable bin boundaries in `mcapm` is consistent with linear interpolation of the function parameters in ENDF/B-VI.

Another complication of translating ENDF/B-VI energy distributions comes from the ENDF option of representing the data as a weighted average of some number of functional formulas,

$$p = \sum_{i=1}^{N} w_i f_i(E').$$

For example, for an (n, 2n) reaction, with different formulas for the energy distributions of the two outgoing neutrons. So far, this option is implemented only for models in which the functions $f_i(E')$ have the same functional form, such as the evaporation model with different effective temperatures. It is not yet possible to handle such cases as when $f_1(E')$ is an evaporation model and $f_2(E')$ is Watt model.

3. MF=6 (Outgoing Energy–Angle Distributions)

The MF=6 files contain double differential (joint energy-angle) distributions. ENDL has two options for representing such data: I=4 files giving the joint distributions as Legendre expansions in the laboratory frame:

$$p(E, \mu) = \sum_n c_n(E) P_n(\mu)$$

and I=3 files containing tables of joint distributions (again in the laboratory frame). ENDF/B-VI allows both of these options and adds the ability to represent data by a formula. Several formulas are allowed but only two are used in practice: the Kalbach-Mann model [13] and a uniform distribution in phase-space.

When an ENDF/B-VI evaluator uses Legendre expansions we can copy it directly to an ENDL I=4 file as this data is always in laboratory coordinates. When an evaluator uses a table of double differential data, we must

first boost to the laboratory frame if necessary and then "transpose" the data. We must transpose the data before copying it into an I=3 file because the ENDF/B-VI column ordering is E, E' then μ whereas ENDL's is E, μ then E'.

The most common MF=6 model used in ENDF/B-VI is the Kalbach-Mann formula. We evaluate the formula for the number of outgoing angles and energies specified by the `num_mu_3d` option. In the future we hope to automate these parameter choices. One area of difficulty with Kalbach-Mann data is that it is in center-of-mass coordinates. Because the double differential distributions are probability densities, the conversion from center-of-mass to laboratory coordinates includes multiplication by the Jacobian of the transformation. We currently use a finite-difference approximation to this Jacobian. This is detailed in Appendix B. Another trouble with Kalbach-Mann data is that the function parameters are given as histogram data in the center-of-mass frame. There is no way to make an exactly faithful representation in the laboratory frame.

As a check on the accuracy of the translation of the Kalbach-Mann data, we have implemented a calculation of the average energy of the outgoing particles. The standard approach within ENDL is to use the `endep` code to integrate the double differential data in laboratory coordinates. The new method of calculating the average energy uses integration of the ENDF/B-VI data in center-of-mass coordinates. You can use the new method by setting the `Kalbach_i10` option. With the current default values of the number of energies and angles to use in the translation, the disagreement in the two approaches is typically about 2%. We expect that this figure would be improved with a better method for selecting the ENDL data points.

Finally, we need to mention that double differential probability distributions are normalized differently in ENDF/B-VI and ENDL. ENDF/B-VI double differential data is normalized such that $\int_0^\infty \int_{-1}^1 p(E,\mu)\, d\mu\, dE = 1$ assuming $p(E,\mu) = 0$ whenever (E,μ) is outside the range of validity of the data. ENDL double differential data is normalized such that $p(E,\mu) = P_1(\mu)P_3(E,\mu)$ where P_i is the distribution in the I=i file. The individual P_1 and P_3 files are normalized so that $\int_0^\infty P_3(E,\mu)\, dE = 1$ for each value of μ and $\int_{-1}^1 P_1(\mu)\, d\mu = 1$. Again, we use the convention that $P_1 = 0$ and $P_3 = 0$ outside the domain of validity of the data.

C. Gamma data.

In ENDF/B-VI the information on emitted gammas is given in files with MF numbers 12 through 15. This data has been the most difficult to put into ENDL format because the ENDL format is more constraining than the ENDF/B-VI format.

In ENDL, it was customary to lump the gamma information into two categories: (n,γ) and (n,Xγ) (the C=46 and C=55 files). In fact, there are only 7 isotopes in ENDL that have gamma files other than C=55 and C=46 and all of them are light isotopes. The majority of these isotopes have gamma data only for inelastic scattering. In ENDF/B-VI it is customary to leave the gamma data with the associated reaction. We have chosen not to combine all of the ENDF/B-VI gamma data into C=55 files, but to keep it with the reactions.

For some targets and certain combinations of reactions, ENDF/B-VI does have total gamma data. For gammas from inelastic scattering, the combined ENDF/B-VI data is given in MT=4 files. ENDF/B-VI may also combine gamma data from any reactions having MT=3 files. In our translation we join such gamma data and write it to C=55 ENDL files.

1. MF=12 (Multiplicity) and MF=13 (Cross Section Data)

The MF=12 and MF=13 ENDF/B-VI data is closely related. For a given reaction, an MF=12 file gives the multiplicity of emitted gammas, while an MF=13 file gives the product of the multiplicity with the reaction cross section (the gamma production cross section). In ENDL the C=55 file with I=0 gives the gamma production cross section, while for individual reactions only the gamma multiplicity may be given (an I=9 file). Thus, if ENDF/B-VI has MF=13 data for a given reaction, we have to divide by the cross section to produce and ENDL I=9 file, while if ENDF/B-VI has MF=12 for a combination of reactions, we have to multiply by the cross section to produce an ENDL C=55 file.

There are two wrinkles we encounter when processing the MF=12 and MF=13 data. First, the MF=12 or MF=13 files may give detailed information on the γ decay chains. For the translation to ENDL we have to accumulate this information into files of energy distributions and multiplicities. Second, ENDL's mechanism for handling discrete and continuum gammas (S=3 files) requires more detail than is present in the ENDF/B-VI data files. The problem with the ENDL S=3 option is that it requires those channels with continuum gammas and those with different discrete excitation levels each be treated as separate reactions, having their own cross section and outgoing energy distributions. To deal with this problem, we introduced a `split_gammas` option. If set, we split the discrete lines out from the continuum gammas in the individual reactions. Otherwise, we convert the discrete lines to a continuum distribution with approximate δ-functions at the lines. This scheme also handles the case where an ENDF/B-VI evaluator represents continuum gamma distributions as a set of discrete lines with frequencies dependent upon the energy of the incident neutron.

We comment that the ENDL unit-base interpolation never made sense for gamma distributions represented as approximate δ-functions, and fortunately there were are

few targets in ENDL99 which have such data. It does make sense to interpolate equally-probable bins for this data, and that is how the translation code is written.

2. MF=14 (Angular Distributions)

The MF=14 files contain information on angular distributions for gammas. These are usually isotropic, and in ENDL the convention is that a distribution is isotropic if no angular information is given. There are a few reactions in ENDF/B-VI for which the MF=14 data is Legendre coefficients. We convert this data to ENDL I=4 format with Legendre expansions.

3. MF=15 (Energy Distributions)

The MF=15 files are for energy distributions of continuum gammas. The corresponding continuum gamma multiplicity is given as the last table in the MF=12 file, or the gamma production cross section is the last table in the MF=13 file.

D. Delayed neutrons

For delayed fission neutrons in ENDF/B-VI some of the multiplicity data is in the MF=1 MT=455 file and some in the MF=5 MT=455 file, so we need to handle them together.

The MF=1 MT=455 file starts with a list of the time constants for delayed fission neutrons. In ENDL format the delayed neutron data is identified by C=15 (fission), along with S=7 (delayed neutrons). The data corresponding to the different time constants is put into different ENDL blocks, with the time constant located in the x[1] slot. This is an extension of the ENDL format, which originally had only one variety of delayed fission neutron.

For C=15 and S=7 we usually produce an I=7 delayed neutron multiplicity file, with separate blocks corresponding to the different time constants. Note that some fissionable targets in ENDF/B-VI lack this data.

ENDF/B-VI may also give energy distributions for delayed fission neutrons. In that case we write a corresponding C=15, S=7, I=4 ENDL energy distribution file.

IV. CONCLUSIONS

Throughout this manual we have dealt with the detailed differences between ENDF/B-VI and ENDL. As a result of these differences, we have had to add several new reactions to ENDL and these are summarized in Table V. Interestingly, there is one case where ENDL has more information than ENF/B-VI. In ENDL, (n, np) and (n, pn) reactions are treated as separate reactions if energy distributions are given only for the outgoing neutron and proton. This is done because the average energy for the gammas and the residual differs slightly depending on the order of particle emission. ENDF/B-VI has just one reaction, which we treat as (n, np).

C	Reaction
16	(n, 3n p)
17	(n, n 2p)
18	(n, 2p)
19	(n, p d)

TABLE V: Reactions added to ENDL.

In addition to these problems, the databases themselves often have out-of-date or sometimes even wrong data. Fortunately the data is updated regularly and problems are corrected. There are several sources for "bug lists":

- NNDC's "Known Errors and Deficiencies in ENDF/B-VI" page at `http://www.nndc.bnl.gov/csewg/errors.html`
- NJOY's "NJOY 99 Issue Tracker" page at `http://t2.lanl.gov/codes/njoy99/Issues.html`
- CEA's nuclear data listserver archive at `http://www.nea.fr/listsmh/`
- D.A. Brown and G. Hedstrom, "Possible problems in ENDF/B-VI.r8" LLNL Report UCRL-CONF-200686 (2003).

Nevertheless, we would like to establish something more comprehensive and centralized such as a `bugzilla` bug tracking page.

Finally we mention that work that we have yet to do. Most serious issues that remain are:

1. Documentation for evaluations in ENDL is completely decoupled from the actual evaluations.
2. Implement translation of outgoing energy distributions when the distribution is weighted sum of models as discussed in Subsection III B 2.
3. The ^{19}F(n, 2n) reaction in ENDF/B-VI has two separate MF=6 double differential data blocks, corresponding to the two outgoing neutrons. The traditional ENDL format permits only average distributions for the two neutrons.
4. While we have made several changes to the ENDL format, not all downstream codes have been updated to handle our changes. In addition to the ^{19}F(n, 2n) problem just mentioned, there is also problems with discrete gamma data and delated fission neutron data.

5. ENDF/B-VI has uncertainties and covariances for several evaluations now, but ENDL has no way to represent either.

Acknowledgments

This work was performed under the auspices of the U.S. Department of Energy by University of California, Lawrence Livermore National Laboratory under Contract W-7405-Eng-48.

APPENDIX A: PREPROCESSING ENDF/B-VI DATA

There are several steps to convert the ENDF/B-VI database into a form suitable for translation: a) split the database into individual nuclei, b) preprocess the database to get it into a pointwise form and c) Doppler broaden the data to 300 degrees Kelvin.

1. Split the ENDF data

The ENDF/B-VI database ships in a collections of files with the `.tape` extension, each containing several materials. First, we break the raw ENDF/B-VI tapes into smaller files, each file containing a complete evaluation for a single isotope. The individual ENDF/B-VI files are placed in parallel directories with names derived from the `za` number of the isotope. This marks the beginning of the ENDL standardization. The splitting of the ENDF/B-VI data and subsequent file organization is accomplished with the `endfsplit` code. It is currently in `ndg.llnl.gov/cvsroot/endfsplit/`

The `endfsplit` project contains two FORTRAN files, a shell script and a Makefile. Typing `make` will build the two executables, `summary` and `endfsplit`, needed by the `splitter` script. This `splitter` script requires two arguments when launched -

```
splitter <original db> <final db>
```

The first argument is the name of the directory containing the ENDF/B-VI tapes, the second argument is the name of the new directory to contain the new file structure. The script first runs the `summary` code, which reads in the ENDF summary file and produces a new file, `neutron.tapes`, which contains the names of the tapes that have incident neutron reaction data in them. The script then reads in the `neutron.tapes` file and runs the `endfsplit` code on each of the tapes listed. The `endfsplit` code breaks the data tapes into single isotope files and places those files in directories named by their `za` number. The `endfsplit` code will create the new directories if necessary and overwrite without hesitation, insuring the desired evaluation precedence.

For other databases, such as JEFF or JENDL, you may have to split and arrange the databases yourself as the `splitter` code will not work with them.

2. Official IAEA Preprocessing codes

The next step in the processing requires the use of official IAEA codes. We store them locally in cvs as `prepro2000` [10]:

```
ndg.llnl.gov:/cvsroot/prepro2000/
```

Inside the `prepro2000/` directory are machine specific `.BAT` files to build, simply execute the appropriate one. Currently, only the `SUN.BAT` and `LINUX.BAT` systems has been tested by the group. Once the executables are built you can launch the `prepro2000` script. The script requires one argument - the directory containing the split ENDF data:

```
prepro2000 <split db>
```

The script launches three codes sequentially for each evaluation file - `linear`, `recent` and `sigma1`. There are three input files used by these codes called `LINEAR.INP`, `RECENT.INP` and `SIGMA1.INP`. The explicit directions and descriptions for these codes are in the `DOCUMENT/` directory. The default input files have been set to linearize the data to a tolerance of 0.1%, expand resonance data to 0.1% tolerance and then heated to 300 degrees Kelvin with a final tolerance of 0.5%. The heating is applied only to the cross sections, not to the angular distributions or energy distributions. The final preprocessed output file for each isotope is renamed to `endf.prepro` and these are the files read in by the `fete`.

There are several common features to the preprocessing codes. These programs only use the ENDF/B-VI format and can handle data in any version of the ENDF/B-VI format. They assume that the data is correctly coded in the ENDF/B-VI format and perform no error checking. In particular they assume that the redundant information on each line is correct. Sequence numbers are ignored on input, but will be correctly output on all lines. The format of section MF=1, MT=451 and all sections of MF=3 must be correct. The program copies all other sections of data as plain text and as such is insensitive to the correctness or incorrectness. In the current versions of these codes, all MF=3 energies will be output in 'f' (instead of 'e') FORTRAN format in order to allow energies to be written with up to 9 digits of accuracy. In previous versions this was an output option. However use of this option to compare the results of energies written in the normal ENDF/B-VI convention of 6 digits to the 9 digit output from this program demonstrated that failure to use the 9 digit output can lead to large errors in the data just due to translation of the energies to the ENDF/B-VI format.

3. Preprocessing code – `linear`

The purpose of this program is to convert ENDF/B-VI MF= 3, 23 and 27 data to linear-linear interpolable form.

Any section that is already linear-linear interpolable will be thinned. Entire evaluations are output, not just the linearized data cross sections, e.g. angular and energy distributions are also passed to the output file. The default input parameter file, `LINEAR.INP`, sets the output tolerance of 0.1% for the entire incident neutron energy range.

4. Preprocessing code – `recent`

The purpose of this this program is to reconstruct the resonance contribution to the cross section in linearly interpolable form, add in any linearly interpolable background cross section and output the result in the ENDF/B-VI format. The cross sections output by this program will be linearly interpolable over the entire energy range.

The resonance contribution is calculated for total (MT=1), elastic (MT=2), capture (MT=102) and fission (MT=18), added to the background (if any) and output. In addition, if there is a first-chance fission (MT=19) background present the resonance contribution of fission will be added to the background and output. If there is no first chance fission background present the program will not output MT=19.

The formats and conventions for reading and interpreting the data varies from one version of ENDF/B to the next. However, if the section (MF=1, MT=451) describing the data is present, it is possible for this program to distinguish between data in the ENDF/B-IV, V and VI formats and to use the appropriate conventions for each ENDF/B version (see, subroutine `file1` for a description of how this is done). If the data description section is not present, the program will assume the data is in the ENDF/B-VI format and use all conventions appropriate to ENDF/B-V. Users are encouraged to insure that the data description section (MF=1, MT=451) is present in all evaluations.

All energies are read in double precision (by special FORTRAN I/O routines) and are treated in double precision in all calculations. Entire evaluations are output, not just the reconstructed MF=3 cross sections, e.g. angular and energy distributions are also included. The default input parameter file is called `RECENT.INP` and sets the output tolerance to 0.1% for the entire incident neutron energy range.

5. Preprocessing code – `sigma1`

The purpose of this program is to Doppler broaden neutron-induced cross sections. Each section of cross sections (MF=3) is read from the ENDF/B-VI format. The data is Doppler broadened, thinned and output in the ENDF/B-VI format. All cross sections that are used by this program must be tabulated and linearly interpolable in energy and cross section. In the unresolved resonance region it is not possible to exactly define the energy dependence of the cross sections. The average widths and spacings given in ENDF/B-VI are only adequate to define average values of the cross sections. Therefore all cross sections in the ENDF/B-VI format for the unresolved region are really average values which cannot be Doppler broadened using the `sigma1` method (which requires tabulated, linearly interpolable, energy dependent cross sections). Therefore,

- All tabulated points within the unresolved resonance region are copied, without modification or broadening. Adoption of this convention allows subsequent programs to properly define self-shielded, Doppler broadened cross sections in the unresolved resonance region.
- Cross sections are extended as $1/v$ below the upper energy limit of the resolved resonance region and above the lower energy limit of the continuum region (i.e. into the unresolved resonance region).

This convention guarantees a smooth behavior close to the unresolved resonance region boundaries. Entire evaluations are output, not just the broadened MF=3 cross sections, e.g. angular and energy distributions are also included. The default input parameter file, `SIGMA1.INP`, is set up to heat the isotopes to 300 degrees Kelvin and output the cross sections using a 0.5% tolerance.

APPENDIX B: DOUBLE DIFFERENTIAL DATA IN ENDF/B-VI

For joint probability distributions of the energy and direction cosine of emitted particles (double differential data), it is common in ENDF/B-VI [3], Chap. 6 for the data to be given in terms of the Kalbach [13] parameterization. A complicating factor is that this representation is in the center-or-mass coordinate system, while double differential data in `ENDL` is in the laboratory frame. Because we are dealing with a probability density, the transformation formula is

$$p_{\mathtt{lab}}(E_{\mathtt{lab}}, \mu_{\mathtt{lab}}) = p_{\mathtt{cm}}(E_{\mathtt{cm}}, \mu_{\mathtt{cm}}) J(E_{\mathtt{cm}}, \mu_{\mathtt{cm}}) \qquad \text{(B1)}$$

Here, $p_{\mathtt{lab}}$ and $p_{\mathtt{cm}}$ are the probability density in, respectively, laboratory and center-of-mass coordinates, $E_{\mathtt{lab}}$ and $E_{\mathtt{cm}}$ are the energies of the emitted particle, $\mu_{\mathtt{lab}}$ and $\mu_{\mathtt{cm}}$ are the direction cosines of the emitted particle relative to the incident particle, and J is the Jacobian of the mapping from center-of-mass to laboratory coordinates. In Eq. (B1), the dependence of $p_{\mathtt{cm}}$ and the Jacobian, J, on $E_{\mathtt{cm}}$ and $\mu_{\mathtt{cm}}$ is parametric so we must compute $E_{\mathtt{cm}}$ and $\mu_{\mathtt{cm}}$ from $E_{\mathtt{lab}}$ and $\mu_{\mathtt{lab}}$ in practice. The Jacobian must be included, because the probability of finding emitted particles within a region of energies and cosines is the integral of p over that region.

The translation code assumes that the `ENDF/B-VI` double differential is given for particles emitted in all directions $-1 \leq \mu_{\mathtt{cm}} \leq 1$ for a range of energies $E_{\mathtt{min}} \leq E_{\mathtt{cm}} \leq$

$E_{\mathtt{max}}$ with $E_{\mathtt{min}} \geq 0$. That is, in center-of-mass coordinates we have a rectangle with $\Omega_{\mathtt{cm}} = \{(E_{\mathtt{cm}}, \mu_{\mathtt{cm}}) : E_{\mathtt{min}} \leq E_{\mathtt{cm}} \leq E_{\mathtt{max}}, -1 \leq \mu_{\mathtt{cm}} \leq 1\}$. In practice, we always have $E_{\mathtt{min}} = 0$.

The steps involved in finding the probability density in laboratory coordinates are as follows. (1) Find the domain $\Omega_{\mathtt{lab}}$ in laboratory coordinates corresponding to the region $\Omega_{\mathtt{cm}}$ where we have data. (2) For a point $(E_{\mathtt{lab}}, \mu_{\mathtt{lab}})$ in $\Omega_{\mathtt{lab}}$ find its image $(E_{\mathtt{cm}}, \mu_{\mathtt{cm}})$ and evaluate the probability density $p_{\mathtt{cm}}(E_{\mathtt{cm}}, \mu_{\mathtt{cm}})$. (3) Compute an approximate value of $J(E_{\mathtt{cm}}, \mu_{\mathtt{cm}})$ and use (B1) to calculate the probability density in the laboratory frame. Let us expand on these concepts.

1. The domain in the laboratory frame

When ENDF/B-VI data is given in center-or-mass coordinates, it is assumed that the mapping to laboratory coordinates is to be done according to Newtonian mechanics [3], p. 6.5. It is also assumed that the target nucleus is at rest. Consequently, if the masses of the target, projectile, and ejected particle are denoted by, respectively, $M_{\mathtt{targ}}$, $M_{\mathtt{proj}}$, and $M_{\mathtt{eject}}$, then a stationary ejected particle in the center-of-mass frame ($E_{\mathtt{cm}} = 0$) has energy

$$E_0 = \frac{M_{\mathtt{proj}} M_{\mathtt{eject}}}{(M_{\mathtt{targ}} + M_{\mathtt{proj}})^2} E_{\mathtt{in}} \tag{B2}$$

in the laboratory frame if the incident particle has (lab frame) energy $E_{\mathtt{in}}$. The direction of such a particle is forward, $\mu_{\mathtt{lab}} = 1$.

For particles emitted with positive energy, the mapping from center-of-mass to laboratory coordinates is given by

$$E_{\mathtt{lab}} = E_0 + E_{\mathtt{cm}} + 2\mu_{\mathtt{cm}} \sqrt{E_0 E_{\mathtt{cm}}}, \tag{B3}$$

$$\mu_{\mathtt{lab}} = \mu_{\mathtt{cm}} \sqrt{\frac{E_{\mathtt{cm}}}{E_{\mathtt{lab}}}} + \sqrt{\frac{E_0}{E_{\mathtt{lab}}}}. \tag{B4}$$

Some curves of $(E_{\mathtt{lab}}, \mu_{\mathtt{lab}})$ for constant E_0 and $E_{\mathtt{cm}}$ for $-1 \leq \mu_{\mathtt{cm}} \leq 1$ are shown in Figure 1. Note that the direction of emission in the laboratory frame is always forward ($\mu_{\mathtt{lab}} > 0$) if $E_{\mathtt{cm}} < E_0$.

In the case that the ENDF/B-VI data is for $0 \leq E_{\mathtt{cm}} \leq E_{\mathtt{max}}$, there are 3 possibilities for the domain $\Omega_{\mathtt{lab}}$ depending on the size of $E_{\mathtt{max}}$ relative to E_0. If $E_{\mathtt{max}} < E_0$, then we have only forward emission in the lab frame, and $\Omega_{\mathtt{lab}}$ is a disk-shaped region bounded by the line $\mu_{\mathtt{lab}} = 1$ and a curve as in Figure 1 with $\mu_{\mathtt{lab}} > 0$. In this case, the minimum value of $\mu_{\mathtt{lab}}$ is $\min \mu_{\mathtt{lab}} = \sqrt{1 - \frac{E_{\mathtt{max}}}{E_0}}$. If $E_{\mathtt{max}} = E_0$, the region $\Omega_{\mathtt{lab}}$ is bounded below by the curve starting at the origin in Figure 1. This is because the point with $E_{\mathtt{cm}} = E_0$ and $\mu = -1$ maps to $E_{\mathtt{lab}} = 0$. Finally, if $E_{\mathtt{max}} > E_0$, then backward emission is possible in the lab frame and $\Omega_{\mathtt{lab}}$ is bounded by a curve as in Figure 1 with $-1 \leq \mu_{\mathtt{lab}} \leq 1$.

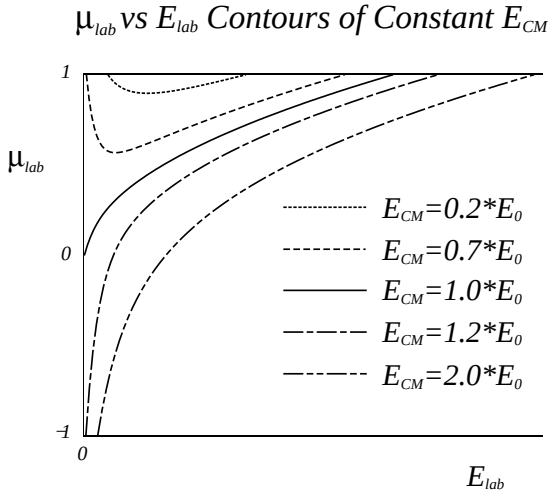

FIG. 1: Mapping center of mass energy and angles into the laboratory frame. Here the center of mass energy, E_{cm}, is given in units of the kinetic energy of the center of mass system, E_0.

If we find that $E_{\mathtt{min}} > 0$ and $E_{\mathtt{min}} \leq E_{\mathtt{cm}} \leq E_{\mathtt{max}}$ for the ENDF/B-VI data, then $\Omega_{\mathtt{lab}}$ is the difference between two of the domains described in the previous paragraph.

2. Center-of-mass probability density

For given $(E_{\mathtt{lab}}, \mu_{\mathtt{lab}})$ in $\Omega_{\mathtt{lab}}$, we may invert Eqs. (B3)–(B4) to get

$$E_{\mathtt{cm}} = E_0 + E_{\mathtt{lab}} - 2\mu_{\mathtt{lab}} \sqrt{E_0 E_{\mathtt{lab}}}, \tag{B5}$$

$$\mu_{\mathtt{cm}} = \mu_{\mathtt{lab}} \sqrt{\frac{E_{\mathtt{lab}}}{E_{\mathtt{cm}}}} - \sqrt{\frac{E_0}{E_{\mathtt{cm}}}}. \tag{B6}$$

We insert these values of $(E_{\mathtt{cm}}, \mu_{\mathtt{cm}})$ into the formulas appropriate to the ENDF/B-VI data [3], Chap. 6.

3. The Jacobian

The Jacobian of the mapping from center-of-mass to laboratory coordinates can be derived several different ways and the result is:

$$J(E_{\mathtt{cm}}, \mu_{\mathtt{cm}}) = \sqrt{\frac{E_{\mathtt{lab}}}{E_{\mathtt{cm}}}}. \tag{B7}$$

Note that this Jacobian has a singularity due to the fact that we have $E_{\mathtt{lab}} = E_0$ and $\mu_{\mathtt{lab}} = 1$ for $E_{\mathtt{cm}} = 0$ and all directions $-1 \leq \mu_{\mathtt{cm}} \leq 1$. Small perturbations cover so much more area in the center-of-mass frame than in the laboratory frame that we have $J(0, \mu_{\mathtt{cm}}) = \infty$ for $-1 \leq \mu_{\mathtt{cm}} \leq 1$. This is not a serious difficulty because differentiation of (B5)–(B6) shows that $J(E_{\mathtt{cm}}, \mu_{\mathtt{cm}}) \sim \frac{1}{\sqrt{E_{\mathtt{cm}}}}$

as $E_{\texttt{cm}} \to 0$, which is an integrable singularity. The Jacobian also has a zero for $E_{\texttt{cm}} = E_0$ and $\mu_{\texttt{cm}} = -1$. This is because the corresponding energy in the laboratory frame is $E_{\texttt{lab}} = 0$ and the direction is arbitrary, $-1 \leq \mu_{\texttt{lab}} \leq 1$.

APPENDIX C: LLNL DISCLAIMER

[1] B. Beck, T. Hill, D. McNabb, "ASCII format specifications for the Evaluated Nuclear Data Libraries (ENDL)," in preparation; R. J. Howerton, R. E. Dye, P. C. Giles, J. R. Kimlinger, S. T.Perkins, and E. F. Plechaty, "Omega: A Cray 1 executive code for LLNL nuclear data libraries", Lawrence Livermore National Laboratory Report UCRL-50400, Vol. 25 (1983).

[2] Evaluated Nuclear Data File `http://www.nndc.bnl.gov/nndc/endf/`.

[3] Cross Section Evaluation Working Group, "ENDF-102 Data Formats and Procedures for the Evaluated Nuclear Data File ENDF-6," Ed. V. McLane, C.L. Dunford, P.F. Rose, Brookhaven National Laboratory Report BNL-NCS-44945, (1997).

[4] `fete` *Code Reference*, part of the `fete` code release in preparation.

[5] LLNL Computational Nuclear Physics Group page `http://nuclear.llnl.gov/`.

[6] G.W. Hedstrom "An explanation of the ENDEP code," Lawrence Livermore National Laboratory Report UCRL-ID-151609 (1999).

[7] B. Beck, "For UpDating ENDL (`fudge`)," manual in preparation.

[8] Japanese Evaluated Nuclear Data Library `http://wwwndc.tokai.jaeri.go.jp/jendl/j33/j33.html`.

[9] Joint European File `http://www.nea.fr/html/dbdata/data/evaluated.htm`.

[10] D. Cullen, `prepro2000` Computer Code available from `http://www-nds.iaea.org/ndspub/endf/prepro/`.

[11] Beck B., "LLNL's Deterministic Transport Access Routines and Data: Documentation for the Nuclear Data Files (ndf) and the libndf.a access routines," Lawrence Livermore National Laboratory Report UCRL-MA-147647 (2002).

[12] Hedstrom G., Cox L., and Perkins S., "The Content and Structure of MCF Files," Lawrence Livermore National Laboratory Report UCRL-ID-127438, Rev. 1 July (1997).

[13] C. Kalbach, "Systematics of continuum angular distributions: Extensions to higher energies", Phys. Rev. C **37** (1988) 2350–2369.