FRENDY: A New Nuclear Data Processing Code being Developed at JAEA

Japan Atomic Energy Agency (JAEA)
Kenichi Tada
Outline

• Background
• Overview of FRENDY
• Comparison of processing results between FRENDY and NJOY
• Conclusions
Overview of nuclear data processing and FRENDY
Importance of nuclear data processing

- Cross section library is the fundamental data for the neutronics calculations
- Reliability of the cross section library has large impact on the neutronics calculation

NJOY is widely used to generate cross section library in Japan
Number of engineers in Japan

- Neutronics calculation code users
  - More than 1,000
- Nuclear data processing code users
  - 1~2 in each company
  - Total: 20~30?
- Expert of nuclear data processing
  - Less than 10
- Technical tradition of nuclear data processing is important
  - Deeply understanding of the nuclear data processing is required to appropriately generate the cross section library
Present situation of nuclear data processing in JAEA

• JAEA provides nuclear data library and many neutronics calculation codes
• The nuclear data processing code had not been developed
  • Imported nuclear data processing code are used
  • JAEA cannot release the nuclear data processing code for our neutronics calculation codes
• Development of domestic nuclear data processing code were desired
Development of nuclear data processing code FRENDY

- JAEA started developing a new nuclear data processing code FRENDY in 2013
  - **FRom Evaluated Nuclear Data library** to any application
  - To process the nuclear data library by JAEA’s nuclear application codes users **with simple input file**
- The first goal is processing the nuclear data for continuous energy Monte Carlo codes
  - For MVP, PHITS of JAEA and MCNP of LANL
Features of FRENDY

- Utilization of modern programming techniques
  - C++, BoostTest library, Git
  - Improvement of quality and reliability
- Consideration of maintainability, modularity, portability and flexibility
  - Encapsulate all classes
  - Minimize the function
  - Maintain the independence of each module
- Processing methods of FRENDY is similar to NJOY99
- Reflecting requests of nuclear data processing code users
  - Development of FRENDY is supported by many organizations and companies in Japan

Development system of FRENDY

- Development of FRENDY is supported by many organizations concerning nuclear data processing in Japan.
- Reflecting requests of nuclear data processing code users.

**Users group**
- JENDL committee
- Nuclear data processing WG
  - Member
  - University, regulatory agency, manufacturer

**Development team**
- Reactor physics group
- Nuclear data group

Discuss development of FRENDY

Report the development status

Requests (function, user interface, ...)

Report the development status
Structure of FRENDY

- Modularity is carefully considered
- Modules of FRENDY can be used in other calculation code by adding only a few lines

Diagram:

- ENDF-6 format
- Endf6Parser /Writer
- Endf6 Converter
- NuclearData Object
  - Resonance Reconstructor
  - DopplerBroader
  - GasProduction CrossSection Calculator

Implemented module
Not implemented module

- Gnds format
- Gnds Parser /Writer
- Gnds Converter
- GNDS format
- HeatingCross SectionGenerator
  - ThermalScattering DataProcessor
  - UnresolvedResonance DataProcessor

- Endf6 Converter
- AceDataGenerator
- AceDataObject
- AceDataParser/Writer
- ACE format
GNDS format

• Developed by OECD/NEA/NSC/WPEC/SG38
  • Currently, maintained by WPEC/EGGNDS

• Completely different from ENDF-6 format
  • Utilizing Extensible Markup Language (XML)
  • It will be used not only for nuclear data file, but also other data file, e.g., cross section library and nuclear structure data file

• LLNL develops FUDGE code to convert ENDF-6 format to GNDS format
  • FUDGE code also processes nuclear data file to generate cross section library for LLNL’s neutronics calculation codes

https://ndclx4.bnl.gov/gf/project/gnd/
https://www.oecd-nea.org/science/wpec/gnds/
Example of ENDF-6 format (MF=3)

(n,2n) XS of Fe-56 from JENDL-4.0

| ZA | AWR | MT/ | ZA | AWR | MT/ | ZA | AWR | MT/ | ZA | AWR | MT/ | ZA | AWR | MT/ | ZA | AWR | MT/ | ZA | AWR | MT/ | ZA | AWR | MT/ | ZA | AWR | MT/ | ZA | AWR | MT/ | ZA | AWR | MT/ | ZA | AWR | MT/ |
| 0.000000-0 | 1.170000-7 | 1.622410-2 | 1.200000-7 | 4.800450-2 | 2.605600+4 | 5.545440+1 | 0 | 0 | 0 | 0 | 0 | 0 | 2.120700+7 | 1.120700+7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

ZA, AWR : 1000.0 × Z+A, mass quantities for materials
QM: Mass-difference Q value (eV)
QI : Reaction Q value
LR : Complex or “breakup” reaction flag
Example of GNDS format

(n,2n) cross section for Fe-56 from JENDL-4.0

<reaction label="29" outputChannel="n[multiplicity: '2']
+ Fe55 + gamma" date="1987-03-01" ENDF_MT="16">
  <crossSection nativeData="linear">
    <linear xData="XYs" length="11" accuracy="0.001">
      <axes>
        <axis index="0" label="energy_in" unit="eV"
          interpolation="linear,linear" frame="lab"/>
        <axis index="1" label="crossSection" unit="b"
          frame="lab"/>
      </axes>
      <data> 1.14e7 0.00000 1.17e7 0.0162241 1.20e7 0.0480045
            1.30e7 0.21382 1.40e7 0.3891650 1.50e7 0.5134000
            1.60e7 0.58175 1.70e7 0.6107500 1.80e7 0.6118000
            1.90e7 0.59770 2.00e7 0.5759000 </data>
    </linear>
  </crossSection>
  <outputChannel genre="NBody" Q="-11202700 eV">
    <product name="n" label="n" multiplicity="2"
      ENDFconversionFlag="MF6">
      <distributions nativeData="Legendre">
        <Legendre nativeData="LegendrePointwise">
        </Legendre>
      </distributions>
    </product>
  </outputChannel>
</reaction>

(n,2n) reaction

Reaction type

Cross Section

Interpolation

Cross section data

Secondary energy and angular distribution
Advantage for using the FRENDY’s original nuclear data format

• FRENDY uses independent internal nuclear data format
  • NuclearDataObject class
• Minimizing the impact by the change of nuclear data format
  • Developer and users are not necessary to consider the nuclear data format
• Consideration of a new data format GNDS
  • GNDS format can be addressed if another set of parser, writer and converter classes are implemented
Input file of FRENDY

• FRENDY treats two types of the input format
  • FRENDY’s original input format
  • NJOY compatible

• Simple input format
  • Nuclear data file name and processing mode are only required for the processing
    • FRENDY has recommended value in the source code
    • User can also change (override) parameters
Input format of FRENDY and NJOY

- Input parameters of FRENDY consist of “input data name” and “input data”
  - Comment line is similar to C/C++
    - // or /* ~ */

- Input parameters of NJOY are hard to understand
  - This input format is so difficult for beginners

---

**Sample input of FRENDY**

```plaintext
ace_fast_mode // Processing mode
nucl_file_name U235.dat
ace_file_name U235.ace
temp 296.0
```

---

**Sample input of NJOY**

```plaintext
reconr
  20  21
  'pendf tape for JENDL-4 U235'
  9228
  1.00e-03  0.00
  0
broadr
  20  21  22
  9228  1
  1.00e-03  -5.0E+2
  296.0
  0
gaspr
  20  22  23
purr
  20  23  25
  9228 1  7 20 500
  296.0
  1E10 1E4 1E3 300 100 30
  0
acer
  20  25 0 30 31
  1 1 1 0.30
  'ACE file for JENDL-4 U235'
  9228 296.0
  1 1
  1 1 1
stop
```
Development schedule of FRENDY

- FRENDY ver.1 will be released in the next spring
  - Generation of ACE file
- Generation of multi-group cross-section library will be implemented in the near future
  - Processing covariance data and calculation of KERMA factor will also be implemented
Present status of nuclear data processing code development

• Development of nuclear data processing code is started in many institute
  • To process their own nuclear data library
  • To handle new nuclear data format GNDS

【Nuclear data processing codes development in the world】

<table>
<thead>
<tr>
<th>Comparison of nuclear data processing code</th>
<th>V&amp;V</th>
<th>Close relationship with users and evaluators</th>
<th>Special focus on domestic utilization including nuclear regulators</th>
<th>Using latest programming technique</th>
<th>Treatment of new nuclear data format</th>
<th>Ease in use for beginners</th>
<th>NJOY compatible I/O</th>
<th>Continuing update and maintenance</th>
<th>Human resources</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Existing code</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NJOY2016</td>
<td>△</td>
<td>○</td>
<td>○</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>○</td>
<td>△</td>
<td>1.5</td>
</tr>
<tr>
<td>PREPRO</td>
<td>△</td>
<td>○</td>
<td>△</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>△</td>
<td>1</td>
</tr>
<tr>
<td><strong>New code</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NJOY21</td>
<td>○</td>
<td>○</td>
<td>△</td>
<td>○</td>
<td>○</td>
<td>×</td>
<td>○</td>
<td>○</td>
<td>2.5</td>
</tr>
<tr>
<td>FRENDY</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>1.5</td>
</tr>
</tbody>
</table>
Comparison of processing results between FRENDY and NJOY
Comparison of processing results

• Processing results of FRENDY are compared to those of NJOY99.393 for verification
  • All nuclei in JENDL-3.3 and JENDL-4.0 are compared
  • We found several programming errors in NJOY

• Calculation conditions
  • Temperature : 296.0 K
  • Tolerance (error): 0.01%
Comparison of processing time

- The processing time to generate ACE files is compared
  - Processing time of FRENDY is similar to that of NJOY
  - Adoption of the fixed energy grid affects the calculation time of the TLS data
- Cause of difference
  - Calculation method
  - Programming language
  - Adopting dynamic array

<table>
<thead>
<tr>
<th></th>
<th>FRENDY</th>
<th>NJOY</th>
<th>F/N</th>
</tr>
</thead>
<tbody>
<tr>
<td>$^1$H</td>
<td>0.1</td>
<td>0.2</td>
<td>0.5</td>
</tr>
<tr>
<td>$^{16}$O</td>
<td>3.1</td>
<td>0.8</td>
<td>3.9</td>
</tr>
<tr>
<td>$^{56}$Fe</td>
<td>18.7</td>
<td>9.1</td>
<td>2.1</td>
</tr>
<tr>
<td>$^{235}$U</td>
<td>821.7</td>
<td>841.0</td>
<td>1.0</td>
</tr>
<tr>
<td>$^{238}$U</td>
<td>507.5</td>
<td>709.1</td>
<td>0.7</td>
</tr>
<tr>
<td>$^{239}$Pu</td>
<td>348.7</td>
<td>534.9</td>
<td>0.7</td>
</tr>
<tr>
<td>$^1$H in H$_2$O</td>
<td>213.8</td>
<td>14.8</td>
<td>14.4</td>
</tr>
<tr>
<td>$^1$H in ZrH</td>
<td>101.7</td>
<td>58.6</td>
<td>1.7</td>
</tr>
<tr>
<td>Graphite</td>
<td>116.9</td>
<td>9.5</td>
<td>12.3</td>
</tr>
</tbody>
</table>

*Intel Xeon CPU E7-8857 v2 (3.00GHz, turbo 3.60GHz)
Comparison of Doppler broadening

- The processing results of FRENDY are similar to those of NJOY99
  - The elastic scattering cross section shows the characteristics difference at the low energy region (less than $1.0 \times 10^{-3}$ eV)
    - The calculation of the cross section at 0.0 eV is different
  - Other nuclei also show similar difference

<table>
<thead>
<tr>
<th>Incident neutron energy [eV]</th>
<th>XS [barn]</th>
</tr>
</thead>
<tbody>
<tr>
<td>$1E+0$</td>
<td>$1E-4$</td>
</tr>
<tr>
<td>$1E-12$</td>
<td>$1E-8$</td>
</tr>
<tr>
<td>$1E-2$</td>
<td>$1E+0$</td>
</tr>
<tr>
<td>$1E+2$</td>
<td>$1E+4$</td>
</tr>
</tbody>
</table>

\[
\frac{(\text{FRENDY-NJOY99})}{\text{NJOY99}}
\]

\[
<^{238}\text{U}, \text{fission, 300 K}> \quad <^{238}\text{U}, \text{elastic scattering, 300 K}>
\]
Calculation of cross section at 0.0 eV

- The cross section at 0.0 eV is required to calculate the Doppler broadened cross section at low energy region.
- NJOY approximates that the cross section follows the 1/v law.
  - Since the elastic scattering cross section at the low energy region is constant, this approximation is not appropriate.
- FRENDY uses linear extrapolation to calculate it.
  - Linear extrapolation is appropriate for other reaction types which obey the 1/v law.
Difference of incoherent inelastic
- Utilization of fixed energy grid -
  • NJOY only calculates the incoherent inelastic XS on 117 energy grids
    • Other energy grids are interpolated using the 5th order Lagrange interpolation
  • The fixed energy grid is not appropriate for a material of which the cross section is oscillated
    • This difference may have impact on the TRIGA reactor

<Incoherent inelastic scattering XS (H in ZrH, 400 K)>
Verification of ACE file generating function

- Comparison of $k_{\text{eff}}$ values of ICSBEP benchmark
  - MCNP sample input files in ICSBEP handbook
    - 79 benchmark experiments, 752 critical configurations
  - Calculation results are not compared to the experimental results
    - Many of sample input files were not intended to be used for the strict validation

- All processes to generate the ACE file are processed by FRENDY and NJOY99.393
  - The processing methods of FRENDY are similar to those of NJOY
    - The programming errors in NJOY is also implemented in FRENDY for the verification

- Processing condition
  - Nuclear data library : JENDL-4.0
  - Temperature : 296.0 K
  - Tolerance (error) : 0.1 %
  - Ladder number : 100
Comparison for integral experiments

- $k_{\text{eff}}$ values of FRENDY are similar to those of NJOY99
  - Differences are not so varied with the neutron spectra and the major fissile materials
- FRENDY properly generates ACE files
Conclusions

• Overview of nuclear data processing
  • Nuclear data processing code is not just a converter
  • It performs many processes to generate cross section library

• Overview of FRENDY
  • Utilization of modern programming techniques
  • Simple input format
  • Reflecting requests of nuclear data processing code users

• Comparison of the processing results
  • Processing results of FRENDY are compatible to those of NJOY99.393/2012.08
Release of FRENDY ver. 1

• FRENDY Ver.1 will is released
  • From our web site or NEA Data-Bank
  • FRENDY Ver.1 is only generates ACE files
    • Generation of multi-group cross section library will be implemented in the near future
  • FRENDY Ver.1 is open source software
    • 2-Clause BSD license

• Manual of FRENDY Ver. 1 is published from JAEA
  • JAEA-Data/Code 2018-014
  • The input instructions and the details of processing method are described