LA-UR-10-04652

Approved for public release;
distribution is unlimited.

Title: Methods for Processing ENDF/B-VII with NJOY

Author(s): R.E. Macfarlane & A.C. Kahler

Intended for: Publication in Nuclear Data Sheets

Los Alamos National Laboratory, an affirmative action/equal opportunity employer, is operated by the Los Alamos National Security, LLC for the National Nuclear Security Administration of the U.S. Department of Energy under contract DE-AC52-06NA25396. By acceptance of this article, the publisher recognizes that the U.S. Government retains a nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or to allow others to do so, for U.S. Government purposes. Los Alamos National Laboratory requests that the publisher identify this article as work performed under the auspices of the U.S. Department of Energy. Los Alamos National Laboratory strongly supports academic freedom and a researcher's right to publish; as an institution, however, the Laboratory does not endorse the viewpoint of a publication or guarantee its technical correctness.

Form 836 (7/06)

Available online at www.sciencedirect.com
ScienceDirect
Nuclear Data Sheets 111 (2010) 2739–2890
Nuclear Data Sheets
www.elsevier.com/locate/nds

Methods for Processing ENDF/B-VII with NJOY

R. E. MacFarlane* and A. C. Kahler†
Nuclear and Particle Physics, Astrophysics and Cosmology
Theoretical Division
Los Alamos National Laboratory,
Los Alamos, NM 87545

(Received 2 July 2010; revised received 16 September 2010; accepted 1 October 2010)

The NJOY Nuclear Data Processing System is widely used to convert evaluations in the Evaluated Nuclear Data Files (ENDF) format into forms useful for practical applications such as fission and fusion reactor analysis, stockpile stewardship calculations, criticality safety, radiation shielding, nuclear waste management, nuclear medicine procedures, and more. This paper provides a description of the system's capabilities, summary descriptions of the methods used, and information on how to use the code to process the modern evaluated nuclear data files from ENDF/B-VII. It begins with the generation of pointwise libraries, including reaction and resonance reconstruction, Doppler broadening, radiation heating and damage, thermal scattering data, unresolved resonance data, and gas production. It then reviews the production of libraries for the continuous-energy Monte Carlo code MCNP, multigroup neutron, photon, and particle cross sections and matrices, and photon interaction data. The generation of uncertainty information for ENDF data is discussed, including new capabilities for calculating covariances of resonance data, angular distributions, energy distributions, and radioactive nuclide production. NJOY's ability to prepare thermal scattering data evaluations for bound moderators (which was used during the preparation of the ENDF/B-VII library) is described. The strong plotting capabilities of NJOY are summarized. Many examples of black&white and color Postscript plots are included throughout the paper. The capabilities of NJOY to output multigroup data in several different formats to suit various applications is reviewed. Finally, a section is included that summarizes the history of the development of the NJOY system over the last 37 years.

Contents

I. Introduction 2741
A. The Modules of NJOY 2741
B. Data Flow in NJOY 2742

II. RECONR 2744
A. ENDF Cross Section Representations 2744
B. Unionization and Linearization Strategy 2744
C. Linearization and Reconstruction Methods 2745
D. Resonance Representations 2746
E. Running RECONR 2752

III. BROADR 2752
A. Doppler-Broadening Theory 2752
B. Thermal Quantities 2754
C. Energy Range for Broadening 2754
D. Running BROADR 2755

IV. HEATR 2755
A. Theory of Nuclear Heating 2755
B. Theory of Damage Energy 2757
C. Computation of KERMA Factors By Energy Balance 2757
D. Kinematic Limits 2758
E. Computation of Damage Energy 2759
F. Heating and Damage from File 6 2761
G. Running HEATR 2762
H. Reading HEATR Output 2763
I. Diagnosing Energy-Balance Problems 2765

V. THERMR 2770
A. Coherent Elastic Scattering 2771
B. Incoherent Inelastic Scattering 2772
C. Incoherent Elastic Scattering 2774
D. Using the ENDF/B-VII Thermal Data Files 2774
E. Running THERMR 2775

VI. PURR 2776
A. Sampling from Ladders 2776

*Electronic address: ryxm@lanl.gov
†Electronic address: akahler@lanl.gov

0090-3752/$ – see front matter © 2010 Published by Elsevier Inc.
doi:10.1016/j.nds.2010.11.001

B. Temperature Correlations 2778
C. Self-Shielded Heating Values 2778
D. Running PURR 2778

VII. GASPR 2779
A. Gas Production 2779
B. Running GASPR 2779

VIII. ACER 2780
A. ACER and ACE Data Classes 2780
B. Continuous-Energy Neutron Data 2780
C. Energy Grids and Cross Sections 2780
D. Two-Body Scattering Distributions 2781
E. Secondary-Energy Distributions 2781
F. Energy-Angle Distributions 2781
G. Photon Production 2783
H. Probability Tables for the Unresolved Region 2784
I. Charged-Particle Production 2784
J. Gas Production 2784
K. Consistency Checks and Plotting 2784
L. Thermal Cross Sections 2785
M. Dosimetry Cross Sections 2787
N. Photoatomic Data 2787
O. Photonuclear Data 2787
P. Type 1 and Type 2 ACE Files 2788
Q. Running ACER 2788

IX. GROUPR 2789
A. Multigroup Constants 2789
B. Group Ordering 2790
C. Basic ENDF Cross Sections 2791
D. Weighting Flux 2791
E. Flux Calculator 2793
F. Fission Source 2795
G. Photon Production and Coupled Sets 2796
H. Thermal Data 2796
I. Generalized Group Integrals 2797
J. Two-Body Scattering 2797
K. Charged-Particle Elastic Scattering 2799
L. Continuum Scattering and Fission 2799
M. File 6 Energy-Angle Distributions 2799
N. GENDF Output 2801
O. Running GROUPR 2802

X. GAMINR 2806
A. Description of the Photon Interaction Files 2807
B. Calculational Method 2808
C. Integrals Involving Form Factors 2808
D. Running GAMINR 2809

XI. ERRORR 2810
A. Definitions of Covariance-Related Quantities 2811
B. Structure of ENDF Files 31, 33, and 40: Energy-Dependent Data 2812
C. Resonance-Parameter Formats—File 32 2813
D. Secondary Particle Angular Distribution Covariances—File 34 2814
E. Secondary Particle Energy Distribution Covariances—File 35 2815
F. Radioactive Nuclide Production Covariances–File 40 2816
G. Calculation of Multigroup Fluxes, Cross Sections, and Covariances on the Union Grid 2816
H. Basic Strategy for Collapse to the User Grid 2818
I. Group-Collapse Strategy for Data Derived by Summation 2818
J. Processing of Data Derived from Ratio Measurements 2819
K. Multigroup Processing of Resonance-Parameter Uncertainties 2821
L. Processing of Lumped-Partial Covariances 2822
M. Running ERRORR 2822
N. ERRORR Output File Specification 2827

XII. COVR 2828
A. Production of Boxer-Format Libraries 2828
B. Generation of Plots 2829
C. Running COVR 2829

XIII. LEAPR 2831
A. Theory 2831
B. Running LEAPR 2838

XIV. PLOTR 2841
A. Simple 2-D Plots 2842
B. Multicurve and Multigroup Plots 2844
C. Right-Hand Axes 2845
D. Plotting Input Data 2846
E. Three-D Plots of Angular Distributions 2847
F. Three-D Plots of Energy Distributions 2847
G. Two-D Spectra Plots from Files 5 and 6 2848

XV. VIEWR 2848
A. Running VIEWR 2849

XVI. MODER 2852
A. Running MODER 2852

XVII. DTFR 2852
A. Transport Tables 2852
B. Data Representations 2853
C. Plotting 2856
D. Running DTFR 2858

XVIII. CCCCR 2859
A. Introduction 2859
B. The Standard Interface Files 2860
C. ISOTXS 2860
D. BRKOXS 2863
E. DLAYXS 2865
F. Running CCCCR 2865

XIX. MATXSR 2867
A. Background 2867
B. The MATXS Format 2868
C. Historical Notes 2873
D. MATXS Libraries 2873
E. Running MATXSR 2876

XX. WIMSR 2877
A. Resonance Integrals 2877
B. Cross Sections 2879
C. Burn Data 2880
D. Running WIMSR 2880
E. WIMS Data File Format 2881
F. WIMSR Auxiliary Codes 2883

XXI. History and Acknowledgments 2883

References 2886

I. INTRODUCTION

The nuclear data evaluations of ENDF/B-VII [1] are physics representations of the data encoded in a computer-readable format called ENDF-6 [2]. Although they can be useful in their own right, they are often converted into forms more suitable for applications, such as neutron transport calculations using multigroup or Monte Carlo techniques. The NJOY Nuclear Data Processing System [3, 4] is widely used for this purpose. This paper will summarize the capabilities of this system and the methods used, with emphasis on preparing nuclear data from ENDF/B-VII evaluations. For more complete details on the methods and the coding in NJOY, please consult the latest code manual [5] and the NJOY web sites [6].

NJOY currently exists in two forms, one using Fortran-77 style (NJOY99), and one using Fortran-95 style (NJOY10). For the purposes of this paper, they work alike for most features, but there are some new features in the Fortran-95 version.

A. The Modules of NJOY

The NJOY code consists of a set of main modules, each performing a well-defined processing task.

NJOY directs the flow of data through the other modules. Subsidiary modules for locale, ENDF formats, physics constants, utility routines, and math routines are grouped with the NJOY module for descriptive purposes.

RECONR reconstructs pointwise (energy-dependent) cross sections from ENDF resonance parameters and interpolation schemes.

BROADR Doppler-broadens and thins pointwise cross sections.

UNRESR computes effective self-shielded pointwise cross sections in the unresolved energy range. For most purposes, UNRESR has been superseded by PURR, and it won't be described here.

HEATR generates pointwise heat production cross sections (KERMA factors) and radiation-damage-production cross sections.

THERMR produces cross sections and scattering distributions for free or bound scatterers in the thermal energy range.

GROUPR generates self-shielded multigroup cross sections, group-to-group scattering matrices, photon production matrices, and charged-particle cross sections from pointwise input.

GAMINR calculates multigroup photoatomic cross sections, KERMA factors, and group-to-group photon scattering matrices.

ERRORR computes multigroup covariance matrices from ENDF uncertainties.

COVR reads the output of ERRORR and performs covariance plotting and output formatting operations.

MODER converts ENDF files back and forth between formatted (that is, ASCII) and blocked binary modes.

DTFR formats multigroup data for transport codes that accept formats based on the DTF-IV code.

CCCCR formats multigroup data for the CCCC standard interface files ISOTXS, BRKOXS, and DLAYXS.

MATXSR formats multigroup data for the newer MATXS material cross-section interface file, which works with the TRANSX code to make libraries for many particle transport codes.

RESXSR prepares pointwise cross sections in a CCCC-like form for thermal flux calculators. It won't be described here.

ACER prepares libraries in ACE format for the Los Alamos continuous-energy Monte Carlo code MCNP. The ACER module is supported by several subsidiary modules for the different classes of the ACE format.

POWR prepares libraries for the EPRI-CELL and EPRI-CPM codes. This is obsolete coding and won't be described here.

WIMSR prepares libraries for the thermal reactor assembly codes WIMS-D and WIMS-E.

PLOTR makes plots of cross sections and perspective views of distributions for both pointwise and multigroup data by generating input for the VIEWR module.

VIEWR converts plotting files produced by the other modules into high-quality color Postscript plots.

MIXR is used to combine cross sections into elements or other mixtures, mainly for plotting. MIXR won't be described here.

PURR is used to prepare unresolved-region probability tables for the MCNP continuous-energy Monte Carlo code.

LEAPR produces thermal scattering data in ENDF-6 File 7 format that can be processed using the THERMR module.

GASPR generates gas-production cross sections in the pointwise PENDF format from basic ENDF cross sections.

B. Data Flow in NJOY

The modules of NJOY can be linked in a number of different ways to prepare libraries for various nuclear applications. The various modules are requested by giving their name in the NJOY input file, and the data flow between the modules is controlled by the Fortran unit numbers for the input and output files of each module. The following brief summary illustrates the typical flow of data in the code.

The first step is normally to generate a "point-ENDF" (PENDF) file. RECONR reads an ENDF file and produces a common energy grid for all reactions (the union grid) such that all cross sections can be obtained to within a specified tolerance by linear interpolation. Resonance cross sections are constructed to the desired accuracy. Summation cross sections (for example, total, inelastic) are reconstructed from their parts. The resulting pointwise cross sections are written onto a pointwise PENDF file for future use. BROADR reads a PENDF file and Doppler-broadens the data using the accurate point-kernel method. The union grid allows all resonance reactions to be broadened simultaneously, resulting in a saving of processing time. After broadening and thinning, the summation cross sections are again reconstructed from their parts. The results are written out on a new PENDF file. HEATR reads the PENDF file and adds energy-balance heating, KERMA, and damage energy using reaction kinematics, applying conservation of energy, or processing the charged-particle spectra given in the ENDF File 6 when available. The energy-balance heating results are added to the PENDF file using ENDF reaction numbers in the 300 series; kinematic KERMA uses the special identifier 443, and damage results use the special identifier 444. PURR reads this PENDF file and writes self-shielded cross sections and probability tables for Monte Carlo methods using special MT numbers in File 2 (152, 153) onto a new PENDF file. This module should be run after HEATR to allow the calculation of self-shielding tables for KERMA. THERMR reads this PENDF file and produces pointwise cross sections in the thermal range. Energy-to-energy incoherent inelastic scattering matrices can be computed for free-gas scattering or for bound scattering using a precomputed scattering law S(α,β) in ENDF format. Coherent-elastic scattering from crystalline materials can be computed using internal lattice information, or for ENDF-6 format files, using data from the evaluation directly. Incoherent-elastic scattering for hydrogenous materials is computed analytically or using ENDF-6 parameters. The results for all the processes are added to the PENDF file using special reaction numbers between 221 and 250. Finally, GASPR reads in the PENDF file, generates additional reactions describing the production of the gases ^{1}H, ^{2}H, ^{3}H, ^{3}He, and ^{4}He, and adds them to the PENDF file with MT=203–207.

```
reconr
20 21
...
broadr
20 21 22
...
heatr
20 22 23/
...
purr
20 23 24
...
thermr
0 24 25
...
gaspr
20 25 26
...
```

FIG. 1: PENDF processing sequence.

At this point in the data flow, the PENDF file is complete and it is useful to save it away for subsequent uses. Fig. 1 shows a skeleton of the NJOY input for this stage of the processing. Start with an ENDF file on unit 20. Save the file on unit 26 as the PENDF file. Note how the intermediate steps of the development of the PENDF file are linked using units 21, 22, 23, *etc.* The details of the input for each of the modules will be discussed below.

One possible application of the PENDF file is to use ACER to prepare cross sections and scattering laws in ACE format (A Compact ENDF) for the MCNP code [7]. All the cross sections are represented on a union grid for linear interpolation by taking advantage of the representation used in RECONR and BROADR. "Laws" for describing scattering and photon production are very detailed, providing a faithful representation of the ENDF-format evaluation with few approximations. The data are organized for random access for purposes of efficiency. MCNP handles self-shielding in the unresolved-

energy range using the probability tables added to the PENDF file by PURR. Formats are provided for incident neutrons, thermal scattering, incident charged particles, photonuclear data, and photoatomic data. ACER produces an ACE file that can be inserted in an MCNP data library, a one-line file with information for the XSDIR index file of an MCNP data library, and optionally, a plotting file that can be passed to VIEWR to make a complete set of color Postscript plots of the Monte Carlo data.

Another common application of the PENDF file is to use GROUPR to process the pointwise cross sections into into multigroup form. The weighting function for group averaging can be taken to be the Bondarenko form, or it can be computed from the slowing-down equation for a heavy absorber in a light moderator. Self-shielded cross sections, scattering matrices, photon production matrices, charged-particle matrices, and photonuclear matrices are all averaged in a unified way. Special features are included for delayed neutrons, coupled energy-angle distributions (either from THERMR or from ENDF-6 evaluations using File 6), discrete scattering angles arising from thermal coherent reactions, and charged-particle elastic scattering. Prompt fission is treated with a full group-to-group matrix. The results are written in a special "groupwise-ENDF" format (GENDF) for use by the output formatting modules. GAMINR uses a specialized version of GROUPR to compute photoatomic cross sections and group-to-group matrices. Coherent and incoherent atomic form factors are processed in order to extend the useful range of the results to lower energies. Photon heat production cross sections are also generated. The results are saved on a GENDF file.

These GENDF files can then be linked to one of several different output formatting modules to prepare multigroup libraries for various applications codes. DTFR is a simple reformatting code that produces cross-section tables acceptable to many discrete-ordinates transport codes. It also converts the GROUPR fission matrix to χ and $\bar{\nu}\sigma_f$ and prepares a photon-production matrix, if desired. The user can define edit cross sections that are any linear combination of the cross sections on the GENDF file. This makes complex edits such as gas production possible. DTFR also prepares plotting files that can be linked to VIEWR to make plots for the cross sections, P_0 scattering matrix, and photon production matrix. This module has become somewhat obsolete with the advent of the MATXS/TRANSX system.

A number of other interface file formats are available from NJOY. The CCCCR module is a straightforward reformatting code that supports all the options of the CCCC-IV [8] file specification. In the cross-section file (ISOTXS), the user can choose either isotope χ matrices or isotope χ vectors collapsed using any specified flux. The BRKOXS file includes the normal self-shielding factors plus self-shielding factors for elastic removal. The DLAYXS file provides delayed-neutron data for reactor kinetics codes. Note that some of the cross sections producible with NJOY are not defined in the CCCC files. For that reason, we have introduced the new CCCC-type material cross section file MATXS. The MATXSR module reformats GENDF data for neutrons, photons, and charged particles into the MATXS format, which is suitable for input to the TRANSX (transport cross section) code [9]. TRANSX can produce libraries for a variety of particle transport codes, such as ANISN [10], ONEDANT [11], DIF3D [12], or the most recent Los Alamos S_N code, PARTISN [13]. The MATXS format uses efficient packing techniques and flexible naming conventions that allow it to store all NJOY data types.

As a third possible use of the full PENDF file, the covariance module ERRORR can either produce its own multigroup cross sections using the methods of GROUPR or start from a precomputed set. The cross sections and ENDF covariance data are combined in a way that includes the effects of deriving one cross section from several others. Special features are included to process covariances for data given as resonance parameters (MF32) or ratios (for example, fission $\bar{\nu}$). ERRORR can also produce covariance matrices for angular distributions (MF34), energy distributions (MF35), and radioactive nuclide production (MF40). The COVR module uses the output from ERRORR together with the VIEWR module to make publication-quality plots of covariance data; it also provides output in the efficient BOXER format, and it provides a site for user-supplied routines to prepare covariance libraries for various sensitivity systems.

MODER is often used at the beginning of an NJOY job to convert ENDF library files into binary mode for calculational efficiency, or at the end of a job to obtain a printable version of a result from ENDF, PENDF, or GENDF input. It can also be used to extract desired materials from a multimaterial library, or to combine several materials into a new ENDF, PENDF, or GENDF file.

At the end of any NJOY run, the PLOTR and VIEWR modules can be used to view the results or the original ENDF data. PLOTR can prepare 2-D plots with the normal combinations of linear and log axes, including legend blocks or curve tags, titles, and so on. Several curves can be compared on one plot (for example, pointwise data can be compared with multigroup results), and experimental data points with error bars can be superimposed, if desired. PLOTR can also produce 3-D perspective plots of ENDF and GENDF angle or energy distributions. The output of PLOTR is passed to VIEWR, which renders the plotting commands into high-quality color Postscript graphics for printing or for viewing on the screen. The HEATR, COVR, DTFR, and ACER modules also produce plot files in VIEWR format that are useful for quality reviews of data-processing results. The MIXR module can be used to combine isotopes into elements for plotting and other purposes. It only works for simple cross sections at the present time.

The following sections describe the methods and the user input for these modules in more detail.

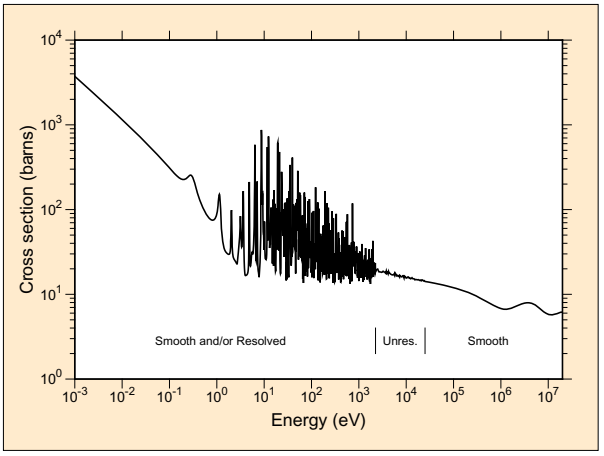

FIG. 2: A typical cross section reconstructed from an ENDF evaluation using RECONR. The smooth, resolved, and unresolved energy regions use different representations of the cross sections. This is the total cross section for ^{235}U from ENDF/B-VII.0.

II. RECONR

The RECONR module is used to reconstruct resonance cross sections from resonance parameters and to reconstruct cross sections from ENDF nonlinear interpolation schemes. The output is written as a PENDF file with all cross sections on a unionized energy grid suitable for linear interpolation to within a user specified tolerance. Redundant reactions (for example, total or inelastic) are reconstructed to be exactly equal to the sum of their reconstructed and linearized parts at all energies. The resonance parameters are removed from File 2, and the material directory is corrected to reflect all changes.

A. ENDF Cross Section Representations

A typical cross section derived from an ENDF evaluation is shown in Fig. 2. The low-energy cross sections are sometimes represented as "smooth". In these cases, they are described in "File 3" of an ENDF evaluation using cross-section values given on an energy grid with a specified law for interpolation between the points. In the resolved resonance range, resonance parameters are given in File 2, and the cross sections for resonance reactions have to be obtained by adding the contributions of all the resonances to "backgrounds" from File 3. At still higher energies comes the unresolved region where explicit resonances are no longer defined. Instead, the cross section is computed from statistical distributions of the resonance parameters given in File 2 and backgrounds from File 3 (or optionally taken directly from File 3 as for smooth cross sections). Finally, at the highest energies, the smooth File 3 representation is used again.

For light and medium-mass isotopes, the unresolved range is usually omitted. For modern evaluations using Reich-Moore resonance representations, the low-energy smooth range is normally omitted and the cross sections there are generated directly from the resonance parameters. For the lightest isotopes, the resolved range is also omitted, the resonance cross sections being given directly in the "smooth" format. In addition, several different resonance representations are allowed (for example, Single-Level Breit-Wigner, Multilevel Breit-Wigner, Reich-Moore, and so on). It is the purpose of RECONR to take all of these separate representations and produce a simple cross section versus energy representation like the one shown in Fig. 2.

B. Unionization and Linearization Strategy

Several of the cross sections found in ENDF evaluations are summation cross sections (for example, total, inelastic, sometimes (n,2n) or fission, and sometimes charged-particle absorption reactions), and it is important that each summation cross section be equal to the sum of its parts. However, if the partial cross sections are represented with nonlinear interpolation schemes, the sum cannot be represented by any simple interpolation law. A typical case is the sum of elastic scattering (MT=2 interpolated linearly to represent a constant) and radiative capture (MT=102 interpolated log-log to represent $1/v$). The total cross section cannot be represented accurately by either scheme unless the grid points are very close together. This effect leads to significant balance errors in multigroup transport codes and to splitting problems in continuous-energy Monte Carlo codes.

Furthermore, the use of linear-linear interpolation (that is, σ linear in E) can be advantageous in several ways. The data can be plotted easily, they can be integrated easily, cross sections can be Doppler-broadened efficiently (see BROADR below), and, finally, linear data can be retrieved efficiently in continuous-energy Monte Carlo codes.

Therefore, RECONR puts all cross sections on a single unionized grid suitable for linear interpolation. The summation cross sections are then obtained by adding up the partial cross sections at each grid point.

If discontinuities are found in the cross sections (typically at the boundaries of the resonance regions), RECONR moves the lower point down by one digit in the last significant figure, and it moves the upper point up by one digit. Therefore, the discontinuity is replaced by a very steep sloping segment, and no duplicate energies will be found in the energy grid. This is done to conform with the requirements of MCNP.

While RECONR is going through the reactions given in the ENDF evaluation, it also checks the reaction thresholds against the Q value and atomic weight ratio to the neutron A (AWR in the file) given for the reaction.

If the condition

$$\text{threshold} \geq \frac{A+1}{A}\,Q$$

is not satisfied, the threshold energy is moved up to satisfy the condition and an informative message is printed. This condition was also introduced by match the requirements of MCNP.

If desired, the unionized grid developed from the ENDF file can be supplemented with "user grid points" given in the input data. The code automatically adds the conventional thermal point of 0.0253 eV and the 1, 2, and 5 points in each decade to the grid if they are not already present. These simple energy grid points help when comparing materials, and they provide well-controlled starting points for further subdivision of the energy grid.

There are special problems with choosing the energy grid in the unresolved range. In some cases, the unresolved cross section is represented using resonance parameters that are independent of energy. The cross sections are not constant, however, but have a shape determined by the energy variation of neutron wave number, penetrability factors, and so on. RECONR handles this case by choosing a set of energies (about 13 per decade) to be used to calculate the cross sections; the set of energies gives a reasonable approximation to the result intended. For evaluations that use energy-dependent resonance parameters, it is supposed to be sufficient to compute the unresolved cross sections at the given energies and to use interpolation on the cross sections to obtain the appropriate values at other energies. However, some evaluations carried over from earlier versions of ENDF/B were not evaluated using this convention, and cross sections computed using cross-section interpolation have unreasonable shapes. Even some modern evaluations have this problem. RECONR detects such cases by looking for large steps between the points of the given energy grid. It then adds additional energy grid points using the same 13-per-decade rule used for energy-independent parameters.

C. Linearization and Reconstruction Methods

Linearization and resonance reconstruction both function by inserting new energy grid points between the points of an original grid until a desired level of accuracy has been obtained. The convergence criterion used for linearization is that the linearized cross section at the intermediate point is within the fractional tolerance `err` (or a small absolute value `errlim`) of the actual cross section specified by the ENDF law. More complicated criteria are used for resonance reconstruction.

There are two basic problems that arise if a simple fractional tolerance test is used to control resonance reconstruction. First, as points are added to the energy grid, adjacent energy values may become so close that they will be rounded to the same number when a formatted output file is produced. There can be serious problems if the code continues to add grid points after this limit is reached. Through the use of dynamic format reconstruction, the energy resolution available for formatted NJOY output (which uses ENDF 11-character fields) is 7 significant figures (that is, $\pm 1.234567 \pm n$) rather than the usual 5 or 6. Note that this requires that more than 32 bits be used to represent floating-point numbers; that is, 64-bit machines or double precision on 32-bit machines are necessary. Even this seven significant figure format is sometimes insufficient for very narrow resonances. If necessary, NJOY can go to nine significant figures by using a Fortran "F" format, *e.g.,* ± 1234.56789.

Significant-figure control is implemented as follows: each intermediate energy is first truncated to 7 significant figures before the corresponding cross sections are computed. If the resulting number is equal to either of the adjacent values and convergence has not been obtained, subdivision continues using energies truncated to 9 significant figures. If an energy on this finer grid is equal to either of the adjacent values, the interval is declared to be converged even though convergence has not been achieved. Thus, no identical energies are produced, but an unpredictable loss in accuracy results. The error in the area of this interval is certainly less than $0.5 \times \Delta\sigma \times \Delta E$, so this value is added to an error estimate and a count of panels truncated by the significant figure check is incremented for a later informative diagnostic message. This diagnostic is somewhat obsolete now that 64-bit precision is used for all energies in RECONR, and the "sig-fig" printout is now omitted.

The second basic problem alluded to above is that a very large number of resonance grid points arise from straightforward linear reconstruction of the resonance cross section of some isotopes. Many of these points come from narrow, weak, high-energy resonances, which do not need to be treated accurately in many applications. As an example, the capture and fission resonance integrals important for thermal reactors must be computed with a $1/E$ flux weighting. If the resonance reconstruction tolerance is set high (say 1%) to reduce the cost of processing, the resonance integrals will be computed to only 1% accuracy. However, if the reconstruction tolerance were set to a smaller value, like 0.1%, and if the high-energy resonances (whose importance is reduced by the $1/E$ weight and the $1/v$ trend of the capture and fission cross sections) were treated with less accuracy than the low-energy resonances, then it is likely that one could achieve an accuracy much better than 1% with an overall reduction in the number of points (hence computing cost). Since $1/E$ weighting is not realistic in all applications (for example, in fast reactors), user control of this "thinning" operation must be provided.

Based on these arguments, the following approach was chosen to control the problem of very large files. First, panels are subdivided until the elastic, capture, and fission cross sections are converged to within `errmax`, where

`errmax` $\geq$ `err`. These two tolerances are normally chosen to form a reasonable band, such as 2% and 0.1%, to ensure that all resonances are treated at least roughly (for example, for plotting). If the resonance integral ($1/E$ weight) in some panel is large, the panel is further subdivided to achieve an accuracy of `err` (say 0.1%). However, if the contribution to the resonance integral from any one interval gets small, the interval will be declared converged, and the local value of the cross section will end up with some intermediate accuracy. Once again, the contribution to the error in the resonance integral should be less than $0.5\times\Delta\sigma\times\Delta E$. This value is added into an accumulating estimate of the error, and a count of panels truncated by the resonance integral check is incremented.

The problem with this test is that RECONR does not know the value of the resonance integral in advance, so the tolerance parameter `errint` is not the actual allowed fractional error in the integral. Instead, it is more like the resonance integral error per grid point (barns/point). Thus, a choice of `errint=err/10000` with `err=0.001` would limit the integral error to about 0.001 barn if 10000 points resulted from reconstruction. Since important resonance integrals vary from a few barns to a few hundred barns, this is a reasonable choice. The integral check can be suppressed by setting `errint` very small or `errmax=err`.

When resonance reconstruction is complete, RECONR provides a summary of the possible resonance integral error due to the integral check over several coarse energy bands. An example is shown in Fig. 3. This is ENDF/B-VII U-235 run at a tolerance of 0.1%. The parameters `errmax` and `errint`, taken together, should be considered as adjustment "knobs" that can increase or decrease the errors in the "res-int" columns to get an appropriate balance between accuracy and economy for a particular application.

For energies in the thermal range (energies less than 0.5 eV), the user's reconstruction tolerance is divided by a factor of 5 in order to give better results for several important thermal integrals, especially after Doppler broadening, and to make the 0.0253-eV cross section behave well under Doppler broadening.

D. Resonance Representations

RECONR uses the resonance formulas as implemented in the original RESEND code [14] with several extensions. The coding handles Single-Level Breit-Wigner (SLBW), Multi-Level Breit Wigner (MLBW), Adler-Adler (AA), Reich-Moore (RM), Hybrid R-Function (HRF), Reich-Moore Limited (RML), energy-independent unresolved, and energy-dependent unresolved representations. The AA and HRF formats are not currently used in ENDF/B-VII and will not be discussed here. Most modern evaluations use RM or MLBW in the resolved range and energy-dependent parameters in the unresolved range.

a. Single-Level Breit-Wigner. The subroutine that computes Single-Level Breit-Wigner (SLBW) cross sections (`csslbw`) uses

$$\begin{aligned}\sigma_n &= \sigma_p \\ &\quad + \sum_\ell \sum_r \sigma_{mr} \Big\{ \Big[\cos 2\phi_\ell - (1 - \frac{\Gamma_{nr}}{\Gamma_r})\Big] \psi(\theta,x) \\ &\quad + \sin 2\phi_\ell \, \chi(\theta,x) \Big\} \ , \qquad (1)\end{aligned}$$

$$\sigma_f = \sum_\ell \sum_r \sigma_{mr} \frac{\Gamma_{fr}}{\Gamma_r} \psi(\theta,x) \ , \qquad (2)$$

$$\sigma_\gamma = \sum_\ell \sum_r \sigma_{mr} \frac{\Gamma_{\gamma r}}{\Gamma_r} \psi(\theta,x) \ , \text{ and} \qquad (3)$$

$$\sigma_p = \sum_\ell \frac{4\pi}{k^2} (2\ell+1) \sin^2\theta_\ell \ , \qquad (4)$$

where σ_n, σ_f, σ_γ, and σ_p are the neutron (elastic), fission, radiative capture, and potential scattering components of the cross section arising from the given resonances. There can be "background" cross sections in File 3 that must be added to these values to account for competitive reactions such as inelastic scattering or to correct for the inadequacies of the single-level representation with regard to multilevel effects or missed resonances. The sums extend over all the ℓ values and all the resolved resonances r with a particular value of ℓ. Each resonance is characterized by its total, neutron, fission, and capture widths $(\Gamma, \Gamma_n, \Gamma_f, \Gamma_\gamma)$, by its J value (AJ in the file), and by its maximum value (`smax`$=\sigma_{mr}/\Gamma_r$ in the code)

$$\sigma_{mr} = \frac{4\pi}{k^2} \, g_J \, \frac{\Gamma_{nr}}{\Gamma_r} \ , \qquad (5)$$

where g_J is the spin statistical factor

$$g_J = \frac{2J+1}{4I+2} \ , \qquad (6)$$

I is the total spin SPI given in File 2, and k is the neutron wave number, which depends on incident energy E and the atomic weight ratio to the neutron for the isotope A (AWRI in the file), as follows:

$$k = (2.196771\times10^{-3}) \frac{A}{A+1} \sqrt{E} \ . \qquad (7)$$

There are two different characteristic lengths that appear in the ENDF resonance formulas: first, there is the "scattering radius" $\hat{a}$, which is given directly in File 2 as AP; and second, there is the "channel radius" a, which is given by

$$a = 0.123 \, A^{1/3} + 0.08 \ . \qquad (8)$$

If the File 2 parameter NAPS is equal to one, a is set equal to $\hat{a}$ in calculating penetrabilities and shift factors

```
                 estimated maximum error due to
                 resonance integral check (errmax,errint)

   upper         elastic   percent   capture   percent   fission   percent
   energy        integral   error    integral   error    integral   error
 1.00E-05
 1.00E-04        3.50E+01   0.000    8.15E+03   0.000    4.29E+04   0.000
 1.00E-03        3.50E+01   0.000    2.57E+03   0.000    1.36E+04   0.000
 1.00E-02        3.50E+01   0.000    8.02E+02   0.000    4.24E+03   0.000
 1.00E-01        3.46E+01   0.000    2.15E+02   0.000    1.26E+03   0.000
 1.00E+00        3.25E+01   0.000    6.03E+01   0.000    3.26E+02   0.000
 2.00E+00        8.95E+00   0.000    7.31E+00   0.000    2.62E+01   0.000
 5.00E+00        1.08E+01   0.000    1.25E+01   0.000    1.56E+01   0.000
 1.00E+01        7.75E+00   0.000    2.40E+01   0.000    3.52E+01   0.000
 2.00E+01        8.18E+00   0.000    2.92E+01   0.000    3.31E+01   0.000
 5.00E+01        1.07E+01   0.000    2.57E+01   0.000    3.83E+01   0.000
 1.00E+02        8.30E+00   0.000    1.07E+01   0.000    2.34E+01   0.000
 2.00E+02        8.04E+00   0.000    8.17E+00   0.000    1.42E+01   0.000
 5.00E+02        1.10E+01   0.001    6.81E+00   0.008    1.51E+01   0.004
 1.00E+03        8.28E+00   0.008    3.44E+00   0.080    7.62E+00   0.038
 2.00E+03        8.27E+00   0.033    2.54E+00   0.261    5.06E+00   0.185

points added by resonance reconstruction    =  232418
points affected by resonance integral check =   80445
final number of resonance points            =  242170
number of points in final unionized grid    =  242594
```

FIG. 3: Example of reconstruction monitoring output for U-235 from ENDF/B-VII.

(see below). The ENDF-6 option to enter an energy-dependent scattering radius is not supported. The neutron width in the equations for the SLBW cross sections is energy dependent due to the penetration factors P_ℓ; that is,

$$\Gamma_{nr}(E) = \frac{P_\ell(E)\,\Gamma_{nr}}{P_\ell(|E_r|)} \ , \tag{9}$$

where

$$P_0 = \rho \ , \tag{10}$$

$$P_1 = \frac{\rho^3}{1+\rho^2} \ , \tag{11}$$

$$P_2 = \frac{\rho^5}{9+3\rho^2+\rho^4} \ , \tag{12}$$

$$P_3 = \frac{\rho^7}{225+45\rho^2+6\rho^4+\rho^6} \ , \text{ and} \tag{13}$$

$$P_4 = \frac{\rho^9}{11025+1575\rho^2+135\rho^4+10\rho^6+\rho^8} \ , \tag{14}$$

where E_r is the resonance energy and $\rho{=}ka$ depends on the channel radius or the scattering radius as specified by NAPS. The phase shifts are given by

$$\phi_0 = \hat{\rho} \ , \tag{15}$$

$$\phi_1 = \hat{\rho} - \tan^{-1}\hat{\rho} \ , \tag{16}$$

$$\phi_2 = \hat{\rho} - \tan^{-1}\frac{3\hat{\rho}}{3-\hat{\rho}^2} \ , \tag{17}$$

$$\phi_3 = \hat{\rho} - \tan^{-1}\frac{15\hat{\rho}-\hat{\rho}^2}{15-6\hat{\rho}^2} \ , \text{ and} \tag{18}$$

$$\phi_4 = \hat{\rho} - \tan^{-1}\frac{105\hat{\rho}-10\hat{\rho}^3}{105-45\hat{\rho}^2+\hat{\rho}^4} \ , \tag{19}$$

where $\hat{\rho}{=}k\hat{a}$ depends on the scattering radius. The final components of the cross section are the actual line shape functions ψ and χ. At zero temperature,

$$\psi = \frac{1}{1+x^2} \ , \tag{20}$$

$$\chi = \frac{x}{1+x^2} \ , \tag{21}$$

$$x = \frac{2(E-E_r')}{\Gamma_r} \ , \tag{22}$$

and

$$E_r' = E_r + \frac{S_\ell(|E_r|)-S_\ell(E)}{2(P_\ell(|E_r|)}\,\Gamma_{nr}(|E_r|) \ , \tag{23}$$

in terms of the shift factors

$$S_0 = 0 , \tag{24}$$

$$S_1 = -\frac{1}{1+\rho^2} , \tag{25}$$

$$S_2 = -\frac{18+3\rho^2}{9+3\rho^2+\rho^4} , \tag{26}$$

$$S_3 = -\frac{675+90\rho^2+6\rho^4}{225+45\rho^2+6\rho^4+\rho^6} , \text{ and} \tag{27}$$

$$S_4 = -\frac{44100+4725\rho^2+270\rho^4+10\rho^6}{11025+1575\rho^2+135\rho^4+10\rho^6+\rho^8} . \tag{28}$$

To go to higher temperatures, define

$$\theta = \frac{\Gamma_r}{\sqrt{\frac{4kTE}{A}}} , \tag{29}$$

where k is the Boltzmann constant and T is the absolute temperature. The line shapes ψ and χ are now given by

$$\psi = \frac{\sqrt{\pi}}{2}\,\theta\,\mathrm{Re}W\big(\frac{\theta x}{2},\frac{\theta}{2}\big) , \tag{30}$$

and

$$\chi = \frac{\sqrt{\pi}}{2}\,\theta\,\mathrm{Im}W\big(\frac{\theta x}{2},\frac{\theta}{2}\big) , \tag{31}$$

in terms of the complex probability function (see `quickw`, `wtab`, and `w`, which came from the MC^2 code [15])

$$W(x,y) = \mathrm{e}^{-z^2}\,\mathrm{erfc}(-iz) = \frac{i}{\pi}\int_{-\infty}^{\infty}\frac{\mathrm{e}^{-t^2}}{z-t}\,dt , \tag{32}$$

where $z{=}x{+}iy$. The $\psi\chi$ method is not as accurate as kernel broadening (see BROADR) because the backgrounds (which are sometimes quite complex) are not broadened, and terms important for energies less than about $16kT/A$ are neglected. In addition, resonances given using pointwise data will not be broadened by $\psi\chi$. However, the $\psi\chi$ method is less expensive than BROADR. The current version of RECONR includes $\psi\chi$ Doppler broadening for the single-level Breit-Wigner, multi-level Breit-Wigner, and Adler-Adler representations only.

The SLBW formalism can produce negative elastic cross sections. If seen, they are changed to a small positive value, and a count is kept for a diagnostic message in the output file.

b. Multi-Level Breit-Wigner. The Lubitz-Rose method used for calculating Multi-Level Breit-Wigner (MLBW) cross sections (`csmlbw`) is formulated as follows:

$$\sigma_n(E) = \frac{\pi}{k^2}\sum_{\ell}\sum_{s=|I-\frac{1}{2}|}^{I+\frac{1}{2}}\sum_{J=|l-s|}^{l+s} g_J\,|1-U_{nn}^{\ell sJ}(E)|^2 , \tag{33}$$

with

$$U_{nn}^{\ell J}(E) = \mathrm{e}^{2i\phi_\ell} - \sum_r \frac{i\Gamma_{nr}}{E_r' - E - i\Gamma_r/2} , \tag{34}$$

where the other symbols are the same as those used above. Expanding the complex operations gives

$$\begin{aligned}\sigma_n(E) &= \frac{\pi}{k^2}\sum_{\ell}\sum_{s=|I-\frac{1}{2}|}^{I+\frac{1}{2}}\sum_{J=|l-s|}^{l+s} g_J\Big\{\Big(1-\cos 2\phi_\ell \\ &- \sum_r \frac{\Gamma_{nr}}{\Gamma_r}\frac{2}{1+x_r^2}\Big)^2 \\ &+ \Big(\sin 2\phi_\ell + \sum_r \frac{\Gamma_{nr}}{\Gamma_r}\frac{2x_r}{1+x_r^2}\Big)^2\Big\} ,\end{aligned} \tag{35}$$

where the sums over r are limited to resonances in spin sequence ℓ that have the specified value of s and J. Unfortunately, the s dependence of Γ is not known. The file contains only $\Gamma_J{=}\Gamma_{s_1J}{+}\Gamma_{s_2J}$. It is assumed that the Γ_J can be used for one of the two values of s, and zero is used for the other. Of course, it is important to include both channel-spin terms in the potential scattering. Therefore, the equation is written in the following form:

$$\begin{aligned}\sigma_n(E) &= \frac{\pi}{k^2}\sum_{\ell}\Big[\sum_J g_J\Big\{\Big(1-\cos 2\phi_\ell \\ &- \sum_r \frac{\Gamma_{nr}}{\Gamma_r}\frac{2}{1+x_r^2}\Big)^2 \\ &+ \Big(\sin 2\phi_\ell + \sum_r \frac{\Gamma_{nr}}{\Gamma_r}\frac{2x_r}{1+x_r^2}\Big)^2\Big\} \\ &+ 2D_\ell(1-\cos 2\phi_\ell)\Big] ,\end{aligned} \tag{36}$$

where the summation over J now runs from

$$||I-\ell|-\frac{1}{2}| \to I+\ell+\frac{1}{2} , \tag{37}$$

and D_ℓ gives the additional contribution to the statistical weight resulting from duplicate J values not included in the new J sum; namely,

$$D_\ell = \sum_{s=|I-\frac{1}{2}|}^{I+\frac{1}{2}}\sum_{J=|l-s|}^{l+s} g_J - \sum_{J=||I-\ell|-\frac{1}{2}|}^{I+\ell+\frac{1}{2}} g_j \tag{38}$$

$$= (2\ell+1) - \sum_{J=||I-\ell|-\frac{1}{2}|}^{I+\ell+\frac{1}{2}} g_j . \tag{39}$$

A case where this correction would appear is the $\ell{=}1$ term for a spin-1 nuclide. There will be 5 J values: 1/2, 3/2, and 5/2 for channel spin 3/2; and 1/2 and 3/2 for channel spin 1/2. All five contribute to the potential scattering, but the file will only include resonances for the first three.

The fission and capture cross sections are the same as for the single-level option. The $\psi\chi$ Doppler-broadening cannot be used with this formulation of the MLBW representation. However, there is an alternate method implemented in `csmlbw2` based on the following equations:

$$\begin{aligned}\sigma_n &= \sigma_p \sum_\ell \sum_r \sigma_{mr} \\ &\Big\{ \Big[\cos 2\phi_\ell - (1 - \frac{\Gamma_{nr}}{\Gamma_r}) + \frac{G_{r\ell}}{\Gamma_{nr}}\Big] \psi(\theta, x) \\ &+ (\sin 2\phi_\ell + \frac{H_{r\ell}}{\Gamma_{nr}})\, \chi(\theta, x)\Big\} , \end{aligned} \tag{40}$$

where

$$\begin{aligned} G_{r\ell} &= \frac{1}{2} \sum_{\substack{r' \neq r \\ J_{r'} \neq J_r}} \Gamma_{nr}\Gamma_{nr'} \\ &\times \frac{\Gamma_r + \Gamma_{r'}}{(E_r - E_{r'})^2 + (\Gamma_r + \Gamma_{r'})^2/4} , \end{aligned} \tag{41}$$

and

$$\begin{aligned} H_{r\ell} &= \sum_{\substack{r' \neq r \\ J_{r'} \neq J_r}} \Gamma_{nr}\Gamma_{nr'} \\ &\times \frac{E_r - Er'}{(E_r - E_{r'})^2 + (\Gamma_r + \Gamma_{r'})^2/4} . \end{aligned} \tag{42}$$

Nominally, this method is slower than the previous one because it contains a double sum over resonances at each energy. However, it turns out that G and H are slowly varying functions of energy, and the calculation can be accelerated by computing them at just a subset of the energies and getting intermediate values by interpolation. It is important to use a large number of r' values on each side of r.

The GH method works well at higher energies when compared to the more accurate kernel broadening method (see BROADR below). However, in the eV range and below, it compares more poorly with kernel broadening. Since the accelerated GH method is only marginally faster than kernel broadening, it probably should not be used at the lower energies for cases where the details of elastic scattering are important. This still leaves it useful for materials like fission products where absorption is the most important factor.

c. Reich-Moore. The Reich-Moore (RM) representation is a multilevel formulation with two fission channels; hence, it is useful for both structural and fissionable materials. The cross sections are given by

$$\begin{aligned} \sigma_t &= \frac{2\pi}{k^2} \sum_\ell \sum_J g_J \Big\{ \big(1 - \mathrm{Re}\, U^{\ell J}_{nm}\big) \\ &+ 2d_{\ell J}\big[1 - \cos(2\phi_\ell)\big]\Big\} , \end{aligned} \tag{43}$$

$$\begin{aligned} \sigma_n &= \frac{\pi}{k^2} \sum_\ell \sum_J g_J \Big\{ |1 - U^{\ell J}_{nn}|^2 \\ &+ 2d_{\ell J}\big[1 - \cos(2\phi_\ell)\big]\Big\} , \end{aligned} \tag{44}$$

$$\sigma_f = \frac{4\pi}{k^2} \sum_\ell \sum_J g_J \sum_c |\mathcal{I}^{\ell J}_{nc}|^2 \text{ , and} \tag{45}$$

$$\sigma_\gamma = \sigma_t - \sigma_n - \sigma_f , \tag{46}$$

where $\mathcal{I}_{nc}$ is an element of the inverse of the complex R-matrix and

$$U^{\ell J}_{nn} = \mathrm{e}^{2i\phi_\ell} \Big[2\mathcal{I}_{nn} - 1 \Big] . \tag{47}$$

The elements of the R-matrix are given by

$$R^{\ell J}_{nc} = \delta_{nc} - \frac{i}{2} \sum_r \frac{\Gamma^{1/2}_{nr}\Gamma^{1/2}_{cr}}{E_r - E - \frac{i}{2}\Gamma_{\gamma r}} . \tag{48}$$

In these equations, "c" stands for the fission channel, "r" indexes the resonances belonging to spin sequence (ℓ, J), and the other symbols have the same meanings as for SLBW or MLBW. Of course, when fission is not present, σ_f can be ignored. The R-matrix reduces to an R-function, and the matrix inversion normally required to get $\mathcal{I}_{nn}$ reduces to a simple inversion of a complex number.

As in the MLBW case, the summation over J runs from

$$||I - \ell| - \frac{1}{2}| \rightarrow I + \ell + \frac{1}{2} . \tag{49}$$

The term $d_{\ell J}$ in the expressions for the total and elastic cross sections is used to account for the possibility of an additional contribution to the potential scattering cross section from the second channel spin. It is unity if there is a second J value equal to J, and zero otherwise. This is just a slightly different approach for making the correction discussed in connection with Eq. (39). Returning to the I=1, ℓ=1 example given above, d will be one for J=1/2 and J=3/2, and it will be zero for J=5/2.

ENDF-6 format Reich-Moore evaluations can contain a parameter LAD that indicates that these parameters can be used to compute an angular distribution for elastic scattering if desired (an approximate angular distribution is still given in File 4 for these cases). The Fortran-77 version of RECONR does not compute angular distributions. The Fortran-95 version has such a capability, and it can be used with RM evaluations. Because of channel-spin issues, it works best with RML evaluations. See the discussion of angular distributions below.

d. Reich-Moore Limited. The new Reich-Moore-Limited representation (RML) is a more general multi-level and multichannel formulation. In addition to the normal elastic, fission, and capture reactions, it allows for inelastic scattering and Coulomb reactions. Furthermore, it allows resonance angular distributions to be calculated. It is also capable of computing derivatives of cross sections with respect to resonance parameters. See ERRORR (Sect. XI). The RML processing in NJOY is based on the SAMMY code [16].

The quantities that are conserved during neutron scattering and reactions are the total angular momentum J and its associated parity π, and the RML format lumps all the channels with a given $J\pi$ into a "spin group." In each spin group, the reaction channels are defined by $c = (\alpha, \ell, s, J)$, where α stands for the particle pair (masses, charges, spins, parities, and Q-value), ℓ is the orbital angular momentum with associated parity $(-1)^\ell$, and s is the channel spin (the vector sum of the spins of the two particles of the pair). The ℓ and s values must vector sum to $J\pi$ for the spin group. The channels are divided into incident channels and exit channels. Here, the important input channel is defined by the particle pair neutron+target. There can be several such incident channels in a given spin group. The exit channel particle pair defines the reaction taking place. If the exit channel is the same as the incident channel, the reaction is elastic scattering. There can be several exit channels that contribute to a given reaction.

The R matrix in the Reich-Moore "eliminated width" approximation for a given spin group is given by

$$R_{cc'} = \sum_\lambda \frac{\gamma_{\lambda c}\gamma_{\lambda c'}}{E_\lambda - E - i\Gamma_{\lambda\gamma}/2} + R_c^b \delta_{cc'}\,, \qquad (50)$$

where c and c' are incident and exit channel indexes, λ is the resonance index for resonances in this spin group, E_λ is a resonance energy, $\gamma_{\lambda c}$ is a resonance amplitude, and $\Gamma_{\lambda\gamma}$ is the "eliminated width," which normally includes all of the radiation width (capture). The channel indexes runs over the "particle channels" only, which doesn't include capture.

The quantity R_c^b is the "background R-matrix." In order to calculate the contribution of this spin group to the cross sections, we first compute the following quantity:

$$X_{cc'} = P_c^{1/2}\, L_c^{-1} \sum_{c''} Y_{cc''}^{-1} R_{c''c'}\, P_{c'}^{1/2}\,, \qquad (51)$$

where

$$Y_{cc''} = L_c^{-1}\delta_{cc''} - R_{cc''}\,, \qquad (52)$$

and

$$L_c = S_c - B_c + iP_c\,. \qquad (53)$$

Here, the P_c and S_c are penetrability and shift factors, and the B_c are boundary constants. The cross sections can now be written down in terms of the $X_{cc'}$. For elastic scattering

$$\begin{aligned}\sigma_{\rm n} &= \frac{4\pi}{k_\alpha^2}\sum_{J\pi}\Big[\sin^2\phi_c(1-2X_{cc}^i) \\ &- X_{cc}^r\sin(2\phi_c) + \sum_{c'}|X_{cc'}|^2\Big]\,, \qquad (54)\end{aligned}$$

where $X_{cc'}^r$ is the real part of $X_{cc'}$, $X_{cc'}^i$ is the imaginary part, ϕ_c is the phase shift, the sum over $J\pi$ is a sum over spin groups, the sum over c is limited to incident channels in the spin group with particle pair α equal to neutron+target, and the sum over c' is limited to exit channels in the spin group with particle pair α. Similarly, the capture cross section becomes

$$\sigma_\gamma = \frac{4\pi}{k_\alpha^2}\sum_{J\pi}\sum_c g_{J\alpha}\sum_c\Big[X_{cc}^i - \sum_{c'}|X_{cc'}|^2\Big]\,, \qquad (55)$$

where the sum over $J\pi$ is a sum over spin groups, the sum over c is a sum over incident channels in the spin group with particle pair α equal to neutron+target, and the sum over c' includes all channels in the spin group. The cross sections for other reactions (if present) are given by

$$\sigma_{reaction} = \frac{4\pi}{k_\alpha^2}\sum_{J\pi} g_{J\alpha}\sum_c\Big[X_{cc}^i - \sum_{c'}|X_{cc'}|^2\Big]\,, \qquad (56)$$

where the sum over c is limited to channels in the spin group $J\pi$ with particle pair α equal to neutron+target, and the sum over c' is limited to channels in the spin group with particle pair α'. The reaction is defined by $\alpha \to \alpha'$. This is one of the strengths of the RML representation. The reaction cross sections can include multiple inelastic levels with full resonance behavior. They can also include cross sections for outgoing charged particles, such as (n,α) cross sections, with full resonance behavior. The total cross section can be computed by summing up its parts.

For non-Coulomb channels, the penetrabilities P, shift factors S, and phase shift ϕ are the same as those given for the SLBW representation, except if a Q value is present, ρ must be modified. They are a little more complicated for Coulomb channels. See the SAMMY reference [16] for more details.

The RML representation is new to the ENDF format and isn't represented by any cases in the initial release of ENDF/B-VII. There are experimental evaluations for F-19 and Cl-37 available. However, the RML approach provides a very faithful representation of resonance physics, and it should see increasing use in the future. NJOY10 is currently able to process RML evaluations using coding adapted from SAMMY.

e. RML Angular Distributions. One of the physics advances available when using the RML format is the calculation of angular distributions from the resonance parameters. A Legendre representation is used:

$$\frac{d\sigma_{\alpha\alpha'}}{d\Omega_{\rm CM}} = \sum_L B_{L\alpha\alpha'}(E)P_L(\cos\beta)\,, \qquad (57)$$

where the subscript $\alpha\alpha'$ indicates the cross section as defined by the particle pairs, P_L is the Legendre polynomial of order L, and β is the angle of the outgoing particle with respect to the incoming neutron in the CM system. The coefficients $B_{L\alpha\alpha'}(E)$ are given by a complicated six level summation that we will omit from this text. Fig. 4 shows the first few Legendre coefficients for the elastic scattering cross sections as computed by NJOY from the experimental evaluation for F-19.

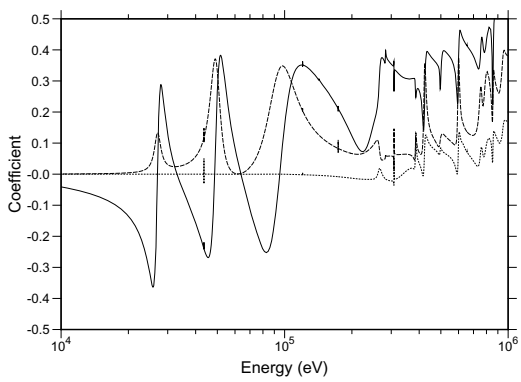

FIG. 4: Legendre coefficients of the angular distribution for elastic scattering in F-19 using the RML resonance representation (P_1 solid, P_2 dashed, P_3 dotted).

Because computing resonance angular distributions is a new feature, it is not enabled by default. To activate it, change `Want_Angular_Dist` to true. The Legendre coefficients are written into a section of File 4 on the RECONR PENDF file. Because the normal ENDF File 4 sections are not copied to the PENDF File 4, the presence of File 4 on a PENDF file can be detected by subsequent modules, such a ACER or GROUPR, and the resonance angular distributions can be used to replace the ENDF File 4 values over the resonance energy range. The default in NJOY10 is to use the conventional RM processing path for RM parameters. However, there is an option to convert the RM parameters into RML format and process them with the RML methods. If this is done, resonance angular distributions can be computed for RM evaluation. Change `Want_SAMRML_RM` to true.

f. Unresolved Resonance Range. Infinitely dilute cross sections in the unresolved-energy range are computed in `csunr1` or `csunr2` using average resonance parameters and probability distributions from File 2. With the approximations used, these cross sections are not temperature dependent; therefore, the results are a good match to resolved resonance data generated using `tempr`>0. The formulas used are based on the single-level approximation with interference.

$$\sigma_n(E) = \sigma_p + \frac{2\pi^2}{k^2}\sum_{\ell,J}\frac{g_J}{\overline{D}}\left[\overline{\Gamma}_n^2 R_n - 2\overline{\Gamma}_n \sin^2\phi_\ell\right] , \quad (58)$$

$$\sigma_x(E) = \frac{2\pi^2}{k^2}\sum_{\ell,J}\frac{g_J}{\overline{D}}\,\overline{\Gamma}_n\overline{\Gamma}_x R_x \text{ , and} \quad (59)$$

$$\sigma_p = \frac{4\pi}{k^2}\sum_{\ell}(2\ell+1)\sin^2\phi_\ell , \quad (60)$$

where x stands for either fission or capture, $\overline{\Gamma}_i$ and $\overline{D}$ are the appropriate average widths and spacing for the ℓ,J spin sequence, and R_i is the fluctuation integral for the reaction and sequence (see `gnrl`). These integrals are simply the averages taken over the chi-square distributions specified in the file; for example,

$$\overline{\Gamma}_n\overline{\Gamma}_f R_i = \left\langle\frac{\Gamma_n\Gamma_f}{\Gamma}\right\rangle = \int dx_n P_\mu(x_n)\int dx_f P_\nu(x_f)\int dx_c P_\lambda(x_c) \times \frac{\Gamma_n(x_n)\,\Gamma_f(x_f)}{\Gamma_n(x_n)+\Gamma_f(x_f)+\Gamma_\gamma+\Gamma_c(x_c)} , \quad (61)$$

where $P_\mu(x)$ is the chi-square distribution for μ degrees of freedom. The integrals are evaluated with the quadrature scheme developed by R. Hwang for the MC^2-2 code [15] giving

$$R_f = \sum_i W_i^\mu \sum_j W_j^\nu \sum_k W_k^\lambda \frac{Q_i^\mu\, Q_j^\nu}{\overline{\Gamma}_n Q_i^\mu + \overline{\Gamma}_f Q_j^\nu + \Gamma_\gamma + \overline{\Gamma}_c Q_k^\lambda} . \quad (62)$$

The W_i^μ and Q_i^μ are the appropriate quadrature weights and values for μ degrees of freedom, and Γ_γ is assumed to be constant (many degrees of freedom). The competitive width $\overline{\Gamma}_c$ is assumed to affect the fluctuations, but a corresponding cross section is not computed. The entire competitive cross section is supposed to be in the File 3 total cross section as a smooth background.

It should be noted that the reduced average neutron width (AMUN) is given in the file, and

$$\overline{\Gamma}_n = \Gamma_n^0\sqrt{E}\,V_\ell(E) , \quad (63)$$

where the penetrabilities for the unresolved region are defined as

$$V_0 = 1 , \quad (64)$$

$$V_1 = \frac{\rho^2}{1+\rho^2} \text{ , and} \quad (65)$$

$$V_2 = \frac{\rho^4}{\rho+3\rho^2+\rho^4} . \quad (66)$$

Other parameters are defined as for SLBW.

Unresolved parameters can be given as independent of energy, with only fission widths dependent on energy, or as fully energy dependent. The first two options are processed in `csunr1`, and the last one is processed in `csunr2`.

E. Running RECONR

The current input instructions for each module are given as comment cards at the beginning of the source code for each module. The versions in the code manuals [4, 5] may not be quite current. Another option is to use the web [6].

A sample input for processing U-235 from ENDF/B-VII follows (the line numbers are for reference only and are not part of the input). NJOY puts the most important parameters at the beginning of each line. Additional parameters can be given their default values by terminating the line with a "/" character.

```
1.  reconr
2.  20 21
3.  'U-235 PENDF from ENDF/B-VII'/
4.  9228 2/
5.  .001/
6.  '92-U-235 from ENDF/B-VII'/
7.  'processed with NJOY'/
8.  0/
```

Card 2 tells RECONR that the input ENDF file containing U-235 will be on unit 20, and that the output PENDF file will be on unit 21. Because the unit numbers are positive, both files will be in ASCII mode. Card 3 is a new title card for the output PENDF file. The MAT number (9228 for U-235) is given on Card 4. The reconstruction tolerance is 0.1% and the default integral thinning parameters are taken. Card 4 also contains a "2" indicating that two comment cards will be provided to be copied onto the output PENDF file, and those appear as lines 6 and 7. The "0/" in line 8 terminates the RECONR run. Input for a second material could have been started here, but this option is rarely used for neutron files. It is useful for photoatomic data.

To reconstruct the cross sections at a temperature other than zero, add a Kelvin temperature to Card 5 in the second field. This feature is not normally used and should be treated with care. Doppler broadening is better when handled with BROADR (see below). If ultimate accuracy is required, turn off the integral thinning by using non-default values for `errmax` and `errint`. Here is an example of Card 5 for running at full 0.1% accuracy and zero temperature:

```
5.  .001  0. .001  1.e-12/
```

III. BROADR

BROADR generates Doppler-broadened cross sections in PENDF format starting from piecewise linear cross sections in PENDF format. The input cross sections can be from RECONR or from a previous BROADR run. The code is based on SIGMA1 [17] by D. E. Cullen. The method is often called "kernel broadening" because it uses a detailed integration of the integral equation defining the effective cross section. It is a fully accurate method, treating all resonance and non-resonance cross sections including multilevel effects. BROADR generates its cross sections on a unionized and linearized grid with reconstruction of redundant cross sections as in RECONR.

A. Doppler-Broadening Theory

The effective cross section for a material at temperature T is defined to be that cross section that gives the same reaction rate for stationary target nuclei as the real cross section gives for moving nuclei. Therefore,

$$\rho v \overline{\sigma}(v,T) = \int d\mathbf{v}' \rho \, |\mathbf{v}-\mathbf{v}'| \, \sigma(|\mathbf{v}-\mathbf{v}'|) \, P(\mathbf{v}',T) \ , \quad (67)$$

where $\mathbf{v}$ is the velocity of the incident particles, $\mathbf{v}'$ is the velocity of the target, ρ is the density of target nuclei, σ is the cross section for stationary nuclei, and $P(\mathbf{v}',T)$ is the distribution of target velocities in the laboratory system. For many cases of interest, the target motion is isotropic and the distribution of velocities can be described by the Maxwell-Boltzmann function

$$P(\mathbf{v}',T)\, d\mathbf{v}' = \frac{\alpha^{3/2}}{\pi^{3/2}} \exp(-\alpha v'^2)\, d\mathbf{v}' \ , \quad (68)$$

where $\alpha = M/(2kT)$, k is Boltzmann's constant, and M is the target mass.

Equation 67 can be partially integrated in terms of the relative speed $V = |\mathbf{v}-\mathbf{v}'|$ to give the standard form of the Doppler-broadened cross section:

$$\overline{\sigma}(v) = \frac{\alpha^{1/2}}{\phi^{1/2}\, v^2} \int_0^\infty dV\, \sigma(V)\, V^2 \left\{ \mathrm{e}^{-\alpha(V-v)^2} - \mathrm{e}^{-\alpha(V+v)^2} \right\} . \quad (69)$$

It is instructive to break this up into two parts:

$$\overline{\sigma}(v) = \sigma^*(v) - \sigma^*(-v) \ , \quad (70)$$

where

$$\sigma^*(v) = \frac{\alpha^{1/2}}{\pi^{1/2} v^2} \int_0^\infty dV\, \sigma(V)\, V^2\, \mathrm{e}^{-\alpha(V-v)^2} \ . \quad (71)$$

The exponential function in Eq. (71) limits the significant part of the integral to the range

$$v - \frac{4}{\sqrt{\alpha}} < V < v + \frac{4}{\sqrt{\alpha}}.$$

TABLE I: Energy parameter for effective Doppler broadening.

Target	Temperature	Energy Parameter (E_m)
H_2	300 K	0.2 eV
U-235	300 K	0.0017 eV
U-235	1.0 keV	69 eV

For $\sigma^*(-v)$, the integral depends only on velocities satisfying

$$0 \le V < \frac{4}{\sqrt{\alpha}}.$$

These results can be converted to energy units using

$$E_m = \frac{1}{2}m\left(\frac{4}{\sqrt{\alpha}}\right)^2 = \frac{16kT}{A}.$$

Some examples are given in Table I. Doppler-broadening effects will be important below this energy and for any features such as resonances, thresholds, or artificial discontinuities in evaluations that are not slowly varying with respect to $2\sqrt{E_m E}$. As an example, for ^{235}U at 100 eV, Doppler effects are important for features smaller than about 0.8 eV.

The numerical evaluation of Eq. (71) developed for SIGMA1 assumes that the cross section can be represented by a piecewise linear function of energy to acceptable accuracy. This is just the form of the NJOY PENDF files (see RECONR). Defining the reduced variables $y = \sqrt{\alpha x}$ and $x = \sqrt{\alpha V}$, the cross section becomes

$$\sigma(x) = \sigma_i + s_i(x^2 - x_i^2) \ , \tag{72}$$

with slope $s_i = (\sigma_{i+1} - \sigma_i)/(x_{i+1}^2 - x_i^2)$. Eq. (71) can now be written as

$$\begin{aligned} \sigma^*(y) &= \frac{1}{\pi^{1/2}y^2}\sum_{i=0}^{N}\int_{x_i}^{x_{i+1}} \sigma(x)\, x^2 \mathrm{e}^{-(x-y)^2} dx \\ &= \sum_i \left\{ A_i\left[\sigma_i - s_i x_i^2\right] + B_i s_i \right\} \ , \end{aligned} \tag{73}$$

where

$$\begin{aligned} x_0 &= 0 \ , \\ x_{N+1} &= \infty \ , \\ A_i &= \frac{1}{y^2}H_2 + \frac{2}{y}H_1 + H_0 \ , \text{ and} \\ B_i &= \frac{1}{y^2}H_4 + \frac{4}{y}H_3 + 6H_2 + 4yH_1 + y^2 H_0 \ , \end{aligned}$$

and where H_n is shorthand for $H_n(x_i - y, x_{i+1} - y)$. The extrapolations to infinity assume a constant cross section ($s_0 = s_N = 0$). The extrapolation to zero uses a log-log shape. The H functions are the incomplete probability integrals defined by

$$H_n(a,b) = \frac{1}{\sqrt{\pi}}\int_a^b z^n \, \mathrm{e}^{-z^2} \, dz \ . \tag{74}$$

These functions can be computed in two ways. First,

$$H_n(a,b) = F_n(a) - F_n(b) \ , \tag{75}$$

where

$$F_n(a) = \frac{1}{\sqrt{\pi}}\int_a^\infty z^n \, \mathrm{e}^{-z^2} \, dz \ . \tag{76}$$

These functions satisfy a recursion relation that can be used to obtain

$$F_0(a) = \frac{1}{2}\mathrm{erfc}(a) \ , \tag{77}$$

$$F_1(a) = \frac{1}{2\sqrt{\pi}}\exp(-a^2) \ , \text{ and} \tag{78}$$

$$F_n(a) = \frac{n-1}{2}F_{n-2}(a) + a^{n-1}F_1(a) \ , \tag{79}$$

where $\mathrm{erfc}(a)$ denotes the complementary error function

$$\mathrm{erfc}(a) = \frac{2}{\sqrt{\pi}}\int_a^\infty \mathrm{e}^{-z^2} dz \ . \tag{80}$$

However, when $F_n(a) \approx F_n(b)$, the difference in Eq. (74) may lose significance. In such cases, $H_n(a,b)$ can be computed by a second method based on a direct Taylor expansion of the defining integral. Write

$$\begin{aligned} H_n(a,b) &= \frac{1}{\sqrt{\pi}}\int_0^b z^n \mathrm{e}^{-z} dz - \frac{1}{\sqrt{\pi}}\int_0^a z^n \mathrm{e}^{-z^2} dz \\ &= G_n(b) - G_n(a) \ . \end{aligned} \tag{81}$$

But by Taylor's Theorem,

$$\begin{aligned} G_n(b) - G_n(a) &= \frac{b-a}{1!}G_n'(a) + \ldots \\ &+ \frac{(b-a)^m}{m!}G_n^{(m)}(a) + \ldots . \end{aligned} \tag{82}$$

Also,

$$G_n^{(m)}(x) = \frac{d^{m-1}}{dx^{m-1}}\left[x^n \mathrm{e}^{-x^2}\right] = \mathrm{e}^{-x^2}P_n^m(x) \, , \tag{83}$$

where $P_n^m(x)$ is a polynomial with recursion relation

$$P_n^m(x) = \frac{d}{dx}P_n^{m-1}(x) - 2xP_n^{m-1}(x) \ , \tag{84}$$

with $P_n^1 = x^n$. From this point, it is straightforward to generate terms until the desired number of significant figures is obtained.

Earlier versions of BROADR and SIGMA1 (before 1984) assumed that the input energy grid from RECONR could also be used to represent the Doppler-broadened cross section before thinning. The grid was then thinned to take advantage of the smoothing effect of Doppler broadening. Unfortunately, this assumption is inadequate. The reconstruction process in RECONR places many points near the center of a resonance to represent its sharp sides. After broadening, the cross section

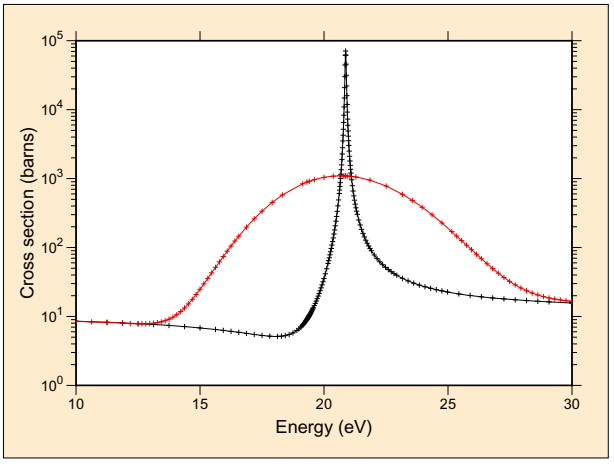

FIG. 5: An expanded plot of the 20 eV resonance for ^{238}U from ENDF/B-VII.0 showing both thinning and "thickening" of the energy grid produced adaptively by BROADR. The two curves show the capture cross section at 0 K and 300 000 K. Note that the high-temperature curve has fewer points than the 0 K curve near the peak at 20 eV and more points in the wings near 15 eV and 25 eV. Clearly, using the 0 K grid to represent the broadened cross section in the wings of this resonance would give poor results. This high temperature was used to exaggerate the effect for the plot.

in this energy region becomes rather smooth; the sharp sides are moved out to energies where RECONR provides few points. At still higher energies, the resonance line shape returns to its asymptotic value, and the RECONR grid is adequate once more. The more recent versions of BROADR check the cross section between points of the incoming energy grid, and add additional grid points if they are necessary to represent the broadened line shape to the desired accuracy. This effect is illustrated in Fig. 5.

It is useful to have some idea of the effects caused by Doppler broadening. First, $1/v$ cross sections are invariant under broadening. Constant elastic cross sections, such as those seen at low energies, develop a $1/v$ tail under broadening. For resonances at energies much greater than kT, they get shorter and broader, preserving their area. At lower energies, however, this is not true, and each resonance generates an additional $1/v$ tail.

It might be surprising that ENDF-format nuclear data is given at zero temperature when this temperature is not accessible by experiment. Evaluators work their way back to zero temperature by including Doppler broadening in the fitting process; thus, the underlying theory parameters can be fairly taken to represent zero temperature. In addition, theory suggests that elastic scattering at low energies is usually constant, and this allows some of the temperature dependence seen in experiments at low energies to be removed reliably. As we have seen above, $1/v$ cross sections do not change with temperature, so experimental values that show that dependence are useful directly.

```
thermal quantities at 293.6 K = 0.0253 eV
-----------------------------------------
       thermal fission xsec:  5.8490E+02
      thermal fission nubar:  2.4367E+00
       thermal capture xsec:  9.8665E+01
   thermal capture integral:  8.6639E+01
   thermal capture g-factor:  9.9086E-01
 capture resonance integral:  1.4046E+02
   thermal fission integral:  5.0605E+02
   thermal fission g-factor:  9.7628E-01
     thermal alpha integral:  1.6828E-01
       thermal eta integral:  2.0859E+00
        thermal k1 integral:  6.4040E+02
              equivalent k1:  7.2262E+02
 fission resonance integral:  2.7596E+02
-----------------------------------------
```

FIG. 6: Example of thermal quantities for U-235 from ENDF/B-VII as computed by the BROADR module.

B. Thermal Quantities

In thermal-reactor work, people make very effective use of a few standard thermal constants to characterize nuclear systems. These parameters include the cross sections at the standard thermal value of 0.0253 eV (2200 m/s), the integrals of the cross sections against a Maxwellian distribution for 0.0253 eV, the g-factors (which express the ratio between a Maxwellian integral and the corresponding thermal cross section), η, α, and K1. Here, η is the Maxwellian-weighted average of $(\bar{\nu}\sigma_f)/(\sigma_f + \sigma_c)$, α is the average of σ_c/σ_f, and K1 is the average of $(\bar{\nu} - 1)\sigma_f - \sigma_c$. If BROADR is run for a temperature close to 293.6 kelvin, which is equivalent to 0.0253 eV), these thermal quantities are automatically calculated and displayed. A sample output for U-235 from ENDF/B-VII is shown in Figure 6.

C. Energy Range for Broadening

There are many different reaction cross sections for each material. However, the cross sections for high velocities are normally smooth with respect to $32kT/A$ for any temperatures outside of stellar photospheres; therefore, they do not show significant Doppler effects. The code uses the input value `thnmax`, or the upper limit of the resolved-resonance energy range, or the lowest threshold (typically>100 keV) as a breakpoint. No Doppler-broadening or energy-grid reconstruction is performed above that energy. No broadening of thresholds is normally done, because we don't have methods to calculate the scattering distributions from broadened thresholds. There is an option to override this for applications like astrophysics that might desire to compute reaction rates for broadened thresholds.

D. Running BROADR

As for RECONR, the input instructions for BROADR can be found in the source code, in the NJOY manuals, or on the web. The following example continues the one from RECONR by broadening the cross sections for U-235 from ENDF/B-VII. The line numbers are for reference only; they are not part of the input.

```
1.  broadr
2.  20 21 22
3.  9228 2/
4.  .001/
5.  293.6 1000.
6.  0/
```

On line 2, unit 20 should contain the ENDF file and unit 21 should a RECONR-generated ASCII PENDF file of 0 K cross sections for U-235 (MAT9228). The "2" on Card 3 indicates that two materials will be generated on unit 22. The two temperatures are given on Card 5. The reconstruction accuracy is given as 0.1% on Card 4. As with RECONR, the optional integral-thinning controls `errmax` and `errint` on Card 4 were defaulted. The best results are obtained when the parameters on Card 4 are the same as those used for the RECONR run.

There are three additional parameters allowed on Card 3: `istart`, `istrap`, and `temp1`. Note that `temp1` need not occur on `nout` if `istart=0`. The restart option (`istart=1`) enables the user to add new temperatures to the end of an existing PENDF tape. This option is also useful if a job runs out of time while processing, for example, the fifth temperature. The job can be restarted from the `nout`. The first four temperatures will be copied to the new `nout` and broadening will continue for temperature five. The bootstrap option speeds up the code by using the broadened result for `temp2(i-1)` as the starting point to obtain `temp2(i)`. However, errors can accumulate using bootstrap, and with modern computers, its use is discouraged. The `thnmax` parameters can be used to speed up a calculation or to prevent the broadening of inappropriate data such as sharp steps or triangles in an evaluated cross section, but it is rarely needed for modern evaluations.

IV. HEATR

The HEATR module generates pointwise heat production cross sections and radiation damage energy production for specified reactions and adds them to an existing PENDF file. The heating and damage numbers can then be easily group averaged, plotted, or reformatted for other purposes. An option of use to evaluators checks ENDF files for neutron/photon energy-balance consistency.

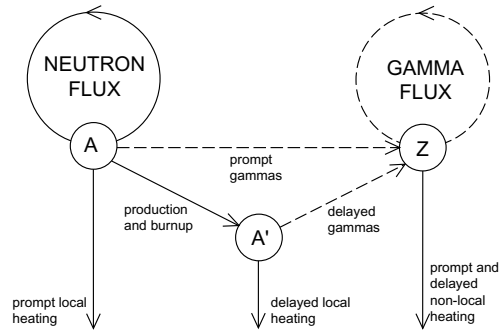

FIG. 7: Components of nuclear heating. HEATR treats the prompt local neutron heating only. Gamma heating is computed in GAMINR. Delayed local heating and photon production are not treated by NJOY, and they must be added at a later stage.

A. Theory of Nuclear Heating

Heating is an important parameter of any nuclear system. It may represent the product being sold—as in a power reactor—or it may affect the design of peripheral systems such as shields and structural components.

Nuclear heating can be conveniently divided into neutron heating and photon heating (see Fig. 7). The neutron heating at a given location is proportional to the local neutron flux and arises from the kinetic energy of the charged products of a neutron induced reaction (including both charged secondary particles and the recoil nucleus itself). Similarly, the photon heating is proportional to the flux of secondary photons transported from the site of previous neutron reactions. It is also traceable to the kinetic energy of charged particles (for example, electron–positron pairs and recoil induced by photoelectric capture).

Heating, therefore, is often described by the KERMA [18] (Kinetic Energy Release in Materials) coefficients $k_{ij}(E)$ defined such that the heating rate in a mixture is given by

$$H(E) = \sum_i \sum_j \rho_i k_{ij}(E)\phi(E) \ , \tag{85}$$

where ρ_i is the number density of material i, $k_{ij}(E)$ is the KERMA coefficient for material i and reaction j at incident energy E, and $\phi(E)$ is the neutron or photon scalar flux at E. KERMA is used just like a microscopic reaction cross section except that its units are energy × cross section (eV-barns for HEATR). When multiplied by a flux and number density, the result would give heating in eV/s.

The "direct method" for computing the KERMA coefficient is

$$k_{ij}(E) = \sum_\ell \overline{E}_{ij\ell}(E)\sigma_{ij}(E) \ , \tag{86}$$

where the sum is carried out over all charged products of the reaction including the recoil nucleus, and $\overline{E}_{ij\ell}$ is the total kinetic energy carried away by the ℓth species of secondary particle. These kinds of data are now becoming available for many materials with the advent of ENDF-6 format, but earlier ENDF/B versions did not include the detailed spectral information needed to evaluate Eq. (86).

For this reason, NJOY computes KERMA factors for many materials by the "energy-balance method." [19] The energy allocated to neutrons and photons is simply subtracted from the available energy to obtain the energy carried away by charged particles:

$$k_{ij}(E) = \left(E + Q_{ij} - \overline{E}_{ijn} - \overline{E}_{ij\gamma}\right)\sigma_{ij}(E) \ , \qquad (87)$$

where Q_{ij} is the mass-difference Q value for material i and reaction j, $\overline{E}_{ijn}$ is the total energy of secondary neutrons including multiplicity, and $\overline{E}_{ij\gamma}$ is the energy of secondary photons including photon yields.

The disadvantage of this method is that the KERMA factor sometimes depends on a difference between large numbers. In order to obtain accurate results, extreme care must be taken with the evaluation to ensure that photon and neutron yields and average energies are consistent. In fact, the lack of consistency in ENDF/B-V and VI often reveals itself as negative KERMA coefficients [20].

However, a negative KERMA coefficient is not always the defect it seems to be. It must be remembered that heating has both neutron and photon components. A negative KERMA might indicate that too much energy has been included with the photon production in the evaluation. This will result in excessive photon heating if most of the photons stay in the system. However, the negative KERMA will have just the right magnitude to cancel this excess heating. The energy-balance method guarantees conservation of total energy in large homogeneous systems.

In this context, large and homogeneous means that most neutrons and photons stay in their source regions. It is clear that energy-balance errors in the evaluation affect the spatial distribution of heat and not the total system heating when the energy-balance method is employed.

A final problem with the energy-balance method occurs for the elemental evaluations in earlier ENDF/B versions (newer evaluated libraries, such as ENDF/B-VII use mostly isotopic evaluations). Isotopic Q values and cross sections are not available in the files. It will usually be possible to define adequate cross sections, yields, and spectra for the element. However, it is clear that the available energy should be computed with an effective Q given by

$$\overline{Q} = \frac{\sum_i \rho_i \sigma_i Q_i}{\sum_i \rho_i \sigma_i} \ , \qquad (88)$$

where ρ_i is the atomic fraction of isotope i in the element. This number is energy dependent and can be represented only approximately by the single constant Q allowed in ENDF/B. HEATR allows the user to input an auxiliary energy-dependent Q for elements.

For elastic and discrete-level inelastic scattering, the neutron KERMA coefficient can be evaluated directly without reference to photon data. For other reactions, conservation of momentum and energy can be used to estimate the KERMA or to compute minimum and maximum limits for the heating. HEATR includes an option that tests the energy-balance KERMA factors against these kinematic limits, thereby providing a valuable test of the neutron–photon consistency of the evaluation. If the energy-balance heating numbers for a particular isotope should fail these tests, and if the isotope is important for a "small" system, an improved evaluation is probably required. The alternative of making *ad hoc* fixes to improve the local heat production is dangerous because the faults in the neutron and/or photon data revealed by the tests may lead to significant errors in neutron transport and/or photon dose and nonlocal energy deposition.

In practice, an exception to this conclusion must be made for the radiative capture reaction (n,γ). The difference between the available energy E+Q and the total energy of the emitted photons is such a small fraction of E+Q that it is difficult to hold enough precision to get reasonable recoil energies. Moreover, the emitted photons cause a component of recoil whose effect is not normally included in evaluated capture spectra. Finally, the "element problem" cited above is especially troublesome for capture because the available energy may change by several MeV between energies dominated by resonances in different isotopes of the element, giving rise to many negative or absurdly large heating numbers. These problems are more important for damage calculations (see below) where the entire effect comes from recoil and the compensation provided by later deposition of the photon energy is absent.

For these reasons, HEATR estimates the recoil due to radiative capture using conservation of momentum. The recoil is the vector sum of the "kick" caused by the incident neutron and the kicks due to the emission of all subsequent photons. Assuming that all photon emission is isotropic and that the directions of photon emission are uncorrelated, the photon component of recoil depends on the average of E_γ^2 over the entire photon spectrum

$$E_R = \frac{E}{A+1} + \frac{\overline{E_\gamma^2}}{2(A+1)mc^2} \ , \qquad (89)$$

where mc^2 is the neutron mass energy. The second term is important below 25–100 keV. This formula gives an estimate that works for both isotopes and elements and has no precision problems. However, it does not explicitly conserve energy, and isotopes with bad capture photon data can still cause problems.

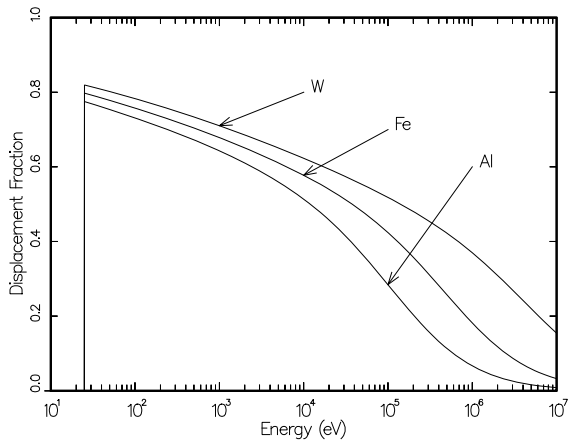

FIG. 8: Examples of the portion of the primary recoil energy that is available to cause lattice displacements in metallic lattices. The remaining energy leads to electronic excitation.

B. Theory of Damage Energy

Damage to materials caused by neutron irradiation is an important design consideration in fission reactors and is expected to be an even more important problem in fusion power systems. There are many radiation effects that may cause damage, for example, direct heating, gas production (*e.g.*, helium embrittlement), and the production of lattice defects.

A large cluster of lattice defects can be produced by the primary recoil nucleus of a nuclear reaction as it slows down in a lattice. It has been shown that there is an empirical correlation between the number of displaced atoms (DPA, displacements per atom) and various properties of metals, such as ductility. The number of displaced atoms depends on the total available energy E_a and the energy required to displace an atom from its lattice position E_d. Since the available energy is used up by producing pairs,

$$\mathrm{DPA} = \frac{E_a}{2E_d} \ . \tag{90}$$

The values of E_d used in practice are chosen to represent the empirical correlations, and a wide range of values is found in the literature [21–23] (see Table II for some examples). The energy available to cause displacements is what HEATR calculates. It depends on the recoil spectrum and the partition of recoil energy between electronic excitations and atomic motion. The partition function used was given by Robinson [24] based on the electronic screening theory of Lindhard [25] (see Fig. 8).

The damage output from HEATR is the damage energy production cross section (eV-barns). As in Eq. (85), multiplying by the density and flux gives eV/s. Dividing by $2E_d$ gives displacements/s. This result is often reduced by an efficiency factor (say 80%) to improve the fit to the empirical correlations.

TABLE II: Typical values for the atomic displacement energy needed to compute DPA [23]. E_d is 25 eV for all other elements.

Element	E_d in eV	Element	E_d in eV
Be	31	Co	40
C	31	Ni	40
Mg	25	Cu	40
Al	27	Zr	40
Si	25	Nb	40
Ca	40	Mo	60
Ti	40	Ag	60
V	40	Ta	90
Cr	40	W	90
Mn	40	Au	30
Fe	40	Pb	25

C. Computation of KERMA Factors By Energy Balance

The ENDF/B files do not usually give photon production data for all partial reactions. Summation reactions such as nonelastic (MT=3) and inelastic (MT=4) are often used. It is still possible to compute partial KERMA factors for those summation reactions by reordering Eq. (87) as follows:

$$k_{ij} = \sum_{j\in J} k^n_{ij}(E) - \sum_{\ell\in J} \overline{E}_{i\ell\gamma}\sigma_{i\ell}(E) \ , \tag{91}$$

where j runs over all neutron partials contained in J, and ℓ runs over all photon partials in J. The total KERMA is well defined, but partial KERMAS should be used only with caution.

HEATR loops through all the neutron reactions on the ENDF/B file. If energy balance is to be used, it computes the neutron contributions needed for the first term. These are

$$k^n_{ij}(E) = \Big[E + Q_{ij} - \overline{E}_{ijn}(E)\Big]\sigma_{ij}(E) \ . \tag{92}$$

The Q value is zero for elastic and inelastic scattering. For (n,n′) particle reactions represented by scattering with an LR flag set, Q is the ENDF "C1" field from MF=3. For most other reactions, Q is the "C2" field from MF=3. HEATR allows users to override any Q value with their own numbers.

There are some special problems with obtaining the Q value for fission. First, the fission Q value given in the C1 field of MF=3 includes delayed contributions, and second, the Q value for fission is energy dependent. The energy dependence can be seen by writing the mass balance equation for the reaction. In energy units,

$$Q = m_A + m_n - m_{fp} - \overline{\nu} m_n \ , \tag{93}$$

where m_A is the mass energy of the target, m_n is the mass energy of the neutron, m_{fp} is the total mass energy of the fission products, and $\overline{\nu}$ is the fission neutron yield. Both m_{fp} and $\overline{\nu}$ are energy dependent. It is also necessary to account for the energy dependence of neutrino emission. Values for some of these terms and estimates for the energy dependence are given in MF=1/MT=458 and the discussion of that section in ENDF-102. We have chosen to ignore the energy dependence of delayed beta and gamma emission because we don't yet treat it in subsequent codes. Therefore, the following corrections are made to get an effective Q for fission in HEATR:

$$\begin{aligned} Q' &= Q - 8.07E6(\overline{\nu}(E) - \overline{\nu}(0)) \\ &+ .057E + .100E - E_{\beta D} - E_{\gamma D} \ , \end{aligned} \tag{94}$$

where $E_{\beta D}$ and $E_{\gamma D}$ are the delayed beta and gamma energies from MT=458. The term $.057E$ is an estimate of the energy dependence of the term m_{fp}, and the term $.100E$ is an estimate of the energy dependence of neutrino emission.

The $\overline{E}_n$ value as used in Eq. (92) is defined to include multiplicity. The multiplicity is either implicit—for example, 2 for (n,2n)—or is retrieved from the ENDF file (fission $\overline{\nu}$). The average energy per neutron is computed differently for discrete two-body reactions and continuum reactions.

For elastic and discrete inelastic scattering (MT=2, 51–90),

$$\overline{E}_n = \frac{E}{(A+1)^2}\Big(1 + 2Rf_1 + R^2\Big) \ , \tag{95}$$

where f_1 is the center-of-mass (CM) average scattering cosine from MF=4 and R is the effective mass ratio. For elastic scattering, R=A, but for threshold scattering,

$$R = A\sqrt{1 - \frac{(A+1)S}{AE}} \ , \tag{96}$$

where S is the negative of the C2 field from MF=3.

For continuum scattering, the average energy per neutron is computed from the secondary neutron spectrum g in MF=5 using

$$\overline{E}_n(E) = \int_0^U E' g(E, E')\, dE' \ , \tag{97}$$

where U is defined in MF=5. If g is tabulated (LAW=1 or LAW=5), the integral is carried out analytically for each panel by making use of the ENDF interpolation laws. For the simple analytic representations (LAW=7, 9, or 11), the average energies are known [2].

The neutron cross sections required by Eq. (92) are obtained from an existing PENDF file (see RECONR and BROADR).

When the neutron sum in Eq. (91) is complete, the code processes the photon production files. If the evaluation does not include photon data, HEATR returns only the first sum. This is equivalent to assuming that all photon energy is deposited locally, consistent with the fact that there will be no contribution to the photon transport source from this material. The same result can be forced by using the `local` parameter (see "Running HEATR" below).

Discrete photon yields and energies are obtained from MF=12 or 13. Continuum photon data are obtained from MF=15, and the average photon energy and $\overline{E_\gamma^2}$ are computed. For radiative capture, the photon term becomes

$$E_\gamma \sigma_\gamma = \left(E + Q - \frac{E}{A+1} + \frac{\overline{E_\gamma^2}}{2(A+1)mc^2} Y_\gamma\right)\sigma_\gamma \ , \tag{98}$$

where Y_γ is the capture photon yield from MF=12. This corrects the capture contribution from Eq. (92) by conservation of momentum. For other reactions, Eq. (92) is sufficient, and the product of $\overline{E}_\gamma$, Y_γ, and σ_γ is subtracted from the neutron contribution.

D. Kinematic Limits

As an option provided mainly as an aid to evaluators, HEATR will compute the kinematic maximum and minimum KERMA coefficients and compare them with the energy-balance results. The formulas are as follows. For elastic scattering (MT=2), the expected recoil energy is

$$\overline{E}_R = \frac{2AE}{(A+1)^2}(1 - f_1) \ . \tag{99}$$

For discrete-inelastic scattering (MT=51–90), the photon momentum is neglected to obtain

$$\begin{aligned} \overline{E}_R &= \frac{2AE}{(A+1)^2}\left[1 - f_1\sqrt{1 - \frac{(A+1)E_\gamma}{AE}}\right] \\ &- \frac{E_\gamma}{A+1} \ , \end{aligned} \tag{100}$$

where E_γ=−C2 from MF=3. For continuum inelastic scattering (MT=91), secondary neutrons are assumed to be isotropic in the laboratory system (LAB) giving

$$\overline{E}_R = \frac{E - E_n}{A} \ , \tag{101}$$

and

$$\overline{E}_\gamma = \frac{(A-1)E - (A+1)\overline{E}_n}{A} \ , \tag{102}$$

where $\overline{E}_\gamma$ is the average photon energy expected for this representation. For radiative capture (MT=102),

$$\overline{E}_R = \frac{E}{A+1} + E_K \tag{103}$$

and

$$\overline{E}_\gamma = Q + \frac{AE}{A+1} - E_K \ , \tag{104}$$

where

$$E_K = \frac{1}{2M_Rc^2}\left[\frac{AE}{A+1}+Q\right]^2 \left\{1 - \frac{1}{M_Rc^2}\left[\frac{AE}{A+1}+Q\right]\right\} \ , \tag{105}$$

with

$$M_Rc^2 = (939.512\times 10^6)(A+1) - Q \tag{106}$$

being the mass energy in eV.

For two-body scattering followed by particle emission (MT=51–91, LR flag set), a minimum and maximum can be defined:

$$(E'_R + E_x)_{\rm min} = \overline{E}_R \ , \text{ and} \tag{107}$$
$$(E'_R + E_x)_{\rm max} = \overline{E}_R + Q + (E_\gamma)_{\rm max} \ , \tag{108}$$

where $\overline{E}_R$ is the value from Eq. (100) or (101), Q is the C2 field from File 3, and $(E_\gamma)_{\rm max}$ is the negative of the C2 field from File 3. In these equations, E'_R is the recoil energy and E_x is the energy of the charged product. For absorption followed by particle emission (MT=103–120),

$$(E_R + E_x)_{\rm min} = \frac{E}{A+1-x} \ , \tag{109}$$
$$(E_\gamma)_{\rm max} = Q + \frac{A-x}{A+1-x}E \ , \text{ and} \tag{110}$$
$$(E_R + E_x)_{\rm max} = E + Q \ , \tag{111}$$

where Q is the C2 field from MF=3 and x is the particle mass ratio (x=1 gives a minimum for all reactions). For (n,2n) reactions,

$$(E_R)_{\rm min} = 0 \ , \text{ and} \tag{112}$$
$$(E_R)_{\rm max} = \frac{E+\overline{E}_n}{A-1} \ , \tag{113}$$

and for (n,3n) reactions,

$$(E_R)_{\rm min} = 0 \ , \text{ and} \tag{114}$$
$$(E_R)_{\rm max} = \frac{E+2\overline{E}_n}{A-2} \ . \tag{115}$$

For both (n,2n) and (n,3n), if $(E_R)_{\rm max}$ is greater than E_R, it is set equal to E_R. In addition, these formulas are not used for A<10; the following case is used instead. For other neutron continuum scattering reactions (MT=22–37),

$$(E_R + E_x)_{\rm min} = 0 \ , \text{ and} \tag{116}$$
$$(E_R + E_x)_{\rm max} = E + Q - \overline{E}_n \ , \tag{117}$$

where Q is the C2 field from File 3. Finally, for fission (MT=18–21, 38), the limits are

$$(E_R)_{\rm min} = E + Q - \frac{1}{2}\overline{E}_n - 15\times 10^6\,{\rm eV} \ , \text{ and} \tag{118}$$
$$(E_R)_{\rm max} = E + Q - \overline{E}_n \ , \tag{119}$$

where Q is the prompt fission Q-value less neutrinos. It is determined by taking the total (less neutrinos) value from File 3 and subtracting the delayed energy obtained from MF=1/MT=458.

These values are intended to be very conservative. Note that E_K is only significant at very low neutron energy. In order to reduce unimportant error messages, a tolerance band is applied to the limits. If all checks are satisfied, the resulting KERMA coefficients should give good local heating results even when 99.8% of the photons escape the local region. More information on using the kinematic checks to diagnose energy-balance problems in evaluations will be found below.

The upper kinematic limit can also be written out to the output file as MT=443 if desired. It is similar to the KERMA factors generated by the MACK code [18], and it is sometimes preferable to the energy-balance KERMA for calculating local heating for evaluations with severe energy-balance problems. The kinematic value in MT=443 is useful for plots (see the examples in this report).

E. Computation of Damage Energy

The formulas used for calculating damage energy are derived from the same sources as the heating formulas given above, except in this case, the effects of scattering angle do not result in simple factors like f_1 because the Robinson partition function is not linear. Instead, it is calculated as follows:

$$P(E) = \frac{E_R}{1 + F_L(3.4008\epsilon^{1/6} + 0.40244\epsilon^{3/4} + \epsilon)} \ , \tag{120}$$

if $E_R \geq 25.0$ eV, and zero otherwise. In Eq. (120), E_R is the primary recoil energy,

$$\epsilon = \frac{E_R}{E_L} \ , \tag{121}$$
$$E_L = 30.724 Z_R Z_L \left(Z_R^{2/3} + Z_L^{2/3}\right)^{1/2} (A_R + A_L)/A_L \ , \text{ and} \tag{122}$$
$$F_L = \frac{0.0793 Z_R^{2/3} Z_L^{1/2} (A_R + A_L)^{2/3}}{\left(Z_R^{2/3} + Z_L^{2/3}\right)^{3/4} A_R^{3/2} A_L^{1/2}} \ , \tag{123}$$

and Z_i and A_i refer to the charge and atomic number of the lattice nuclei (L) and the recoil nuclei (R). The function behaves like E_R at low recoil energies and then levels out at higher energies. Therefore, the damage-energy production cross section is always less than the heat production cross section. See Fig. 8 for examples.

For elastic and two-body discrete-level inelastic scattering,

$$E_R(E,\mu) = \frac{AE}{(A+1)^2}\Big(1 - 2R\mu + R^2\Big) \ , \qquad (124)$$

where the "effective mass" is given by

$$R = \sqrt{1 - \frac{(A+1)(-Q)}{AE}} \ , \qquad (125)$$

and μ is the CM scattering cosine. The damage energy production cross section is then obtained from

$$D(E) = \sigma(E) \int_{-1}^{1} f(E,\mu)\, P(E_R[E,\mu])\, d\mu \ , \qquad (126)$$

where f is the angular distribution from the ENDF File 4. This integration is performed with a 20-point Gauss-Legendre quadrature. Discrete-level reactions with LR flags to indicate, for example, (n,n′)α reactions, are treated in the same way at present. The additional emitted particles are ignored.

Continuum reaction like (n,n′) give a recoil spectrum

$$E_R(E,E',\mu) = \frac{1}{A}\Big(E - 2\sqrt{EE'}\mu + E'\Big) \ , \qquad (127)$$

where E' is the secondary neutron energy, μ is the laboratory cosine, and the photon momentum has been neglected. The damage becomes

$$D(E) = \sigma(E) \int_0^\infty dE' \int_{-1}^{1} d\mu \\ f(E,\mu)\, g(E,E')\, P(E_R[E,E',\mu]) \ , \qquad (128)$$

where g is the secondary energy distribution from File 5. In the code, the angular distribution is defaulted to isotropic, and a 4-point Gaussian quadrature is used for the angular integration. For analytic representations of g, an adaptive integration to 5% accuracy is used for E'; for tabulated File 5 data, a trapezoidal integration is performed using the energy grid of the file. The same procedure is used for (n,2n), (n,3n), etc., but it is not realistic for reactions like (n,n′p) or (n,n′α). The neutron in these types of reactions can get out of the nucleus quite easily; thus, much of the energy available to secondary particles is typically carried away by the charged particles [26]. HEATR treats these reactions in the same way as (n,p) or (n,α).

The recoil for radiative capture must include the momentum of the emitted photons below 25–100 keV giving

$$E_R = \frac{E}{A+1} - 2\sqrt{\frac{E}{A+1}}\sqrt{\frac{E_\gamma^2}{2(A+1)mc^2}}\cos\phi \\ + \frac{\overline{E_\gamma^2}}{2(A+1)mc^2} \ , \qquad (129)$$

where ϕ is the angle between the incident neutron direction and emitted photon direction. If subsequent photons are emitted in a cascade, each one will add an additional term of $\overline{E_\gamma^2}$ and an additional angle. A complete averaging of Eq. (129) with respect to $P(E_R)$ would be very difficult and would require angular correlations not present in ENDF. However, damage calculations are still fairly crude, and an estimate for the damage obtained by treating the neutron "kick" and all the photon kicks independently should give a reasonable upper limit because

$$\int_{-1}^{1} D(E_R)\, d\cos\phi \ \le \ D\left(\frac{E}{A+1}\right) \\ + \sum_\gamma D\left(\frac{\overline{E_\gamma^2}}{2M_R c^2}\right) \ . \qquad (130)$$

The actual formula used in the code is

$$D(E) = D\left(\frac{E}{A+1}\right) + D\left(\frac{1}{2M_R c^2}\left[\frac{AE}{A+1} + Q\right]^2\right) \\ + \sum_\gamma D\left(\frac{\overline{E_\gamma^2}}{2M_R c^2}\right) \\ - D\left(\frac{1}{2M_R c^2}\left[\frac{AE}{A+1} + Q\right]^2\right) \ , \qquad (131)$$

where the first line is computed in the neutron section, and the second line is computed in the photon section. This form also provides a reasonable default when no photons are given.

Finally, for the (n,particle) reactions, the primary recoil is given by

$$E_R = \frac{1}{A+1}\Big(E^* - 2\sqrt{aE^*E_a}\cos\phi + aE_a\Big) \ , \qquad (132)$$

where a is the mass ratio of the emitted particle to the neutron, E^* is given by

$$E^* = \frac{A+1-a}{A+1}\, E \ , \qquad (133)$$

and the particle energy E_a is approximated as being equal to the smaller of the available energy

$$Q + \frac{AE}{A+1} \ , \qquad (134)$$

or the Coulomb barrier energy

$$\frac{1.029 \times 10^6\, zZ}{a^{1/3} + A^{1/3}} \text{ in eV} \ , \qquad (135)$$

where z is the charge of the emitted particle and Z is the charge of the target. A more reasonable distribution would be desirable [26], but this one has the advantage of eliminating an integration, and most results are dominated by the kick imparted by the incident neutron anyway. The angular distribution for the emitted particle is taken as isotropic in the lab. At high incident energies,

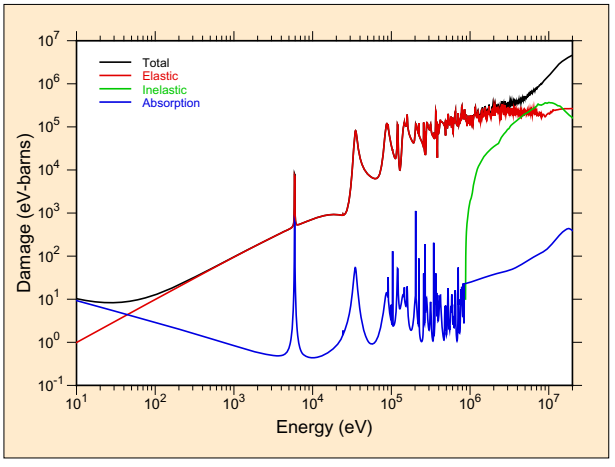

FIG. 9: Components of radiation damage energy production for ^{27}Al from ENDF/B-VII. Note that capture dominates at very low energies, then elastic dominates, and finally inelastic begins to contribute at very high energies.

direct interaction processes would be expected to give rise to a forward-peaked distribution, thereby reducing the damage. However, the importance of this effect is also reduced by the dominance of the neutron kick.

Figure 9 gives a typical result for a damage energy production calculation, showing the separate contributions of elastic, inelastic, and absorption processes.

F. Heating and Damage from File 6

A number of the evaluations in ENDF/B-VII include complete energy-angle distributions for all of the particles produced by a reaction, including the residual nucleus. In these cases, HEATR can compute the contributions to KERMA by calculating the average energy in the spectrum of each outgoing charged particle or residual nucleus and using Eq. (86).

A fully-defined section of File 6 contains subsections for all of the particles and photons produced by the reaction, including the recoil nucleus. There are a number of different schemes used to represent the energy-angle distributions for these outgoing particles. The normal procedure is to loop through all of these subsections. The subsections producing neutrons are processed to be used in a total energy check, but they contribute nothing to the heating or to the damage. Subsections describing charged particles and residual nuclei are processed into heating and damage contributions. Finally, the photon subsection is processed for the photon energy check and the total energy check, even though it does not affect either heating or damage. Any remaining difference between the eV-barns available for the reaction and the eV-barns carried away by the neutrons, photons, particles, and recoil is added into the heating to help preserve the total energy deposition in the spirit of the energy-balance method.

For "two-body" sections, the emitted particle energy is given by

$$E' = \frac{A'E}{A+1}\left(1 + 2R\mu + R^2\right) , \tag{136}$$

where

$$R = \sqrt{\frac{A(A+1-A')}{A'}} , \tag{137}$$

and A' is the ratio of the mass of the outgoing particle to that of the incident particle. The heating is obtained by doing a simple integral over μ, and the damage is computed using the integral over μ given in Eq. (126). In both cases, the integrals are performed using either a 20-point Gauss-Legendre quadrature (for Legendre representations) or a trapezoidal integration (for tabulated data).

For laboratory distributions that use the E, E', μ ordering, the angular part can be ignored, and the heating and damage become

$$K(E) = \int g(E \rightarrow E')E' \, dE' , \tag{138}$$

and

$$D(E) = \int g(E \rightarrow E')P(E') \, dE' , \tag{139}$$

where $g(E \rightarrow E')$ is the angle-integrated energy distribution from File 6, and $P(E')$ is the damage partition function. Trapezoidal integration is used for the continuum, and the integrand is simply added into the sum for the delta functions (if any).

Heating for subsections that use the ordering E, μ, E' is computed using the formula

$$K(E) = \int \left\{ \int g(E \rightarrow E', \mu)E' \, dE' \right\} d\mu , \tag{140}$$

where an inner integral is performed using trapezoidal integration for each value in the μ grid. The results are then used in a second trapezoidal integration over μ. The damage integral is performed at the same time in a parallel manner.

The problem is somewhat more difficult for subsections represented in the center-of-mass frame. The definitions for $K(E)$ and $D(E)$ are the same as those given above, except that the quantity $g(E \rightarrow E')$ has to be generated in the lab system. The methods used to do the transformation are basically the same in HEATR and GROUPR. The first step is to set up an adaptive integral over E'. The first value needed to prime the stack is obtained by calling `h6cm` with E'=0. It returns the corresponding value of g in the lab system and a value for `epnext`. The second value for the stack is computed for E=`epnext`. The routine then subdivides this interval until 2% convergence is achieved, accumulating the contributions to

the heating and damage integrals as it goes. It then moves up to a new panel. This process continues until the entire range of E' has been covered.

The key to this process is `h6cm`. As described in more detail in the GROUPR section of this paper, it performs integrals of the form

$$g_L(E \rightarrow E'_L) = \int_{\mu_{\min}}^{+1} g_C(E \rightarrow E'_C, \mu_C)\, J\, d\mu_L \ , \qquad (141)$$

where L and C denote the laboratory and center-of-mass systems, respectively, and J is the Jacobian for the transformation. The contours in the E'_C,μ_C frame that are used for these integrals have constant E'_L. The limiting cosine $\mu_{\min}$ depends on kinematic factors and the maximum possible value for E'_C in the File 6 tabulation.

The ENDF/B-VII library contains a few abbreviated versions of File 6 that contain an energy-angle distribution for neutron emission, but no recoil or photon data. In order to get semi-reasonable results for both heating and damage for such cases, HEATR applies a "one-particle recoil approximation," where the first particle emitted is assumed to induce all the recoil. There are also some cases where capture photons are described in MF=6/MT=102 with no corresponding recoil data. Here, the recoil can be added using the same logic described above for capture represented using File 15. The difference between the eV-barns available for the reaction and the energy accounted for by the emitted neutrons, photons, particles, and the approximated recoil is added into the heating in order to preserve the total heating in the spirit of the energy-balance method.

G. Running HEATR

The best places to find the formal input instructions for HEATR are in the source code or on the web [6]. Here is an example continuing the processing of U-235 that produces HEATR results and also performs kinematic checks.

```
1. heatr
2. 20 22 23/
3. 9228 5/
4. 302 318 402 443 444/
5. heatr
6. 20 23 24 30/
7. 9228 7 0 1 0 2/
8. 302 303 304 318 402 443 444
```

Card 2 specifies the input and output units for HEATR. They are all ENDF-type files. The input PENDF file has normally been through RECONR and BROADR, but it is possible to run HEATR directly on an ENDF file in order to do kinematic checks. In this case, the results in the resonance range should be ignored. Card 6 has an additional entry for `nplot` which will produce a file of input for the VIEWR module containing detailed energy-balance test results.

Card 3 gives the U-235 MAT number and requests that five additional HEATR quantities be generated. Those quantities are defined on Card 4. The total heating value (MT301) is automatically produced. In this example, additional heating values are requested for elastic (302), fission (318), capture (402), and the total kinematic KERMA. Total damage (444) is also requested. The kinematic KERMA computed when MT=443 is requested is very useful for judging the energy-balance consistency of an evaluation (see the subsection on "Diagnosing Energy-Balance Problems" below). It can also be used instead of the energy-balance value in MT=301 when local heating effects are important and the evaluation scores poorly in an energy-balance check. Damage energy production cross sections (MT=444) should be computed for important structural materials; this expensive calculation can be omitted for other materials. The number of temperatures to be run was defaulted on Card 3, which causes all temperatures on the input PENDF file to be processed.

The second HEATR run in the example produces kinematic checks. Card 7 requests that 7 additional values be produced. The additional parameters are `nqa` for user Q values, `ntemp=1`, which means process only one temperature on the input PENDF file, `local=1`, which means that gamma rays are expected to be transported and not deposited locally, and finally `iprint=2` to indicate that check output is to be printed. This final option should always be used for kinematic checking runs. Card 8 specifies the 7 additional values to be printed out for the kinematic checks.

The additional MT values for kinematic checks can be determined by checking the evaluation to see what partial KERMA factors are well defined. For old-style evaluations that do not use File 6, look for the MT values used in Files 12 and 13. Many evaluations use only MT=3 and MT=102 (or 3, 18, and 102 for fissionable materials); in these cases, the only `mtk` values that make sense are 302, 303, and 402 (or 302, 303, 318, and 402 for fissionable materials). Caution: in some older evaluations, MT=102 is used at low energies and taken to zero at some breakpoint. MT=3 is used at higher energies. In these evaluations, the partial KERMA MT=402 does not make sense above the breakpoint, and MT=3 does not make sense below it.

More complicated photon-production evaluations may include MT=4 and/or discrete-photon data in MT=51–90. In these cases, the user can request `mtk=304`. The same kind of energy-range restriction discussed for MT=102 can occur for the inelastic contributions. Other evaluations give additional partial reactions that can be used to check the photon production and energy-balance consistency of an evaluation in detail. Unfortunately, HEATR can only handle 6 additional reactions at a time. Multiple runs may be necessary in complex cases.

Note that several special `mtk` values are provided for the components of the damage-energy production cross section. They were used to prepare Fig. 9, and they may

be of some interest to specialists, but they are not needed for normal data libraries.

In a few cases in the past, it has been necessary to change the Q values that are normally retrieved from the ENDF file. In addition, it is sometimes necessary to replace the single Q value supplied in MF=3 with an energy-dependent Q function for an element. One example of the former occurred for ^{16}O for ENDF/B-V. The first inelastic level (MT=51) decays by pair production rather than the more normal mode of photon emission. In order to get the correct heating, it was necessary to change the Q value by giving Card 4 and Card 5 as follows:

```
51
-5.0294e6
```

That is, the Q value is increased by twice the electron energy of 0.511 MeV. Another example is the sequential (n,2n) reaction for ^{9}Be in ENDF/B-V. It is necessary to include 4 changes to the Q values:

```
46 47 48 49/
-1.6651e6 -1.6651e6 -1.6651e6 -1.6651e6/
```

The next example illustrates using energy-dependent Q values for elemental titanium. Set `nqa` equal to 3 and provide the following additional cards:

```
16 103 107/
99e6 99e6 99e6/
0. 0. 0 0 1 8
8 2
8.0e6 -8.14e6 9.0e6 -8.14e6 1.1e7 -8.38e6
1.2e7 -8.74e6 1.3e7 -1.03e7 1.4e7 -1.091e7
1.5e7 -1.11e7 2.0e7 -1.125e7/
0. 0. 0 0 1 9
9 2
1.0e-5 1.82e5 4.0e6 1.82e5 5.0e6 -1.19e6
6.0e6 -2.01e6 7.0e6 -2.20e6 8.0e6 -2.27e6
1.4e7 -2.35e6 1.7e6 -2.43e6 2.0e7 -2.37e6/
0. 0. 0 0 1 9
9 2
1.0e-5 2.182e6 6.0e6 2.182e6 7.0e6 2.10e6
8.0e6 -3.11e5 9.0e6 -9.90e% 1.0e7 -1.20e6
1.1e7 -1.27e6 1.4e6 -1.32e6 2.0e7 -1.48e6/
```

Elemental evaluations are becoming rare in modern evaluated nuclear data libraries.

H. Reading HEATR Output

When full output and/or kinematic checks have been requested, HEATR loops through the reactions found in Files 3, 12, and 13. For each reaction, it prints out information about the energies, yields, cross sections, and contributions to heating. The energy grid used is a subset of the PENDF grid. At present, decade steps are used below 1 eV, factor-of-two steps are used from 1 eV to 100 keV, quarter-lethargy steps are used above 100 keV, and approximately 1 MeV steps are used above 2 MeV. An example of this printout for elastic scattering in ENDF/B-VI ^{27}Al is shown in Fig. 10. Note the identification and Q information printed on the first line; `q` is the ENDF Q value from File 3, and `q0` is the corresponding mass-difference Q value needed for Eq. (87). The `ebar`, `yield`, and `xsec` columns contain $\overline{E}_n$, Y, and σ, respectively. The `heating` column is just $(E+Q-Y\overline{E}_n)\sigma$. The results are similar for discrete inelastic levels represented using File 4. The heating due to the associated photons will be subtracted later while MF=12 or MF=13 is being processed. However, if an LR flag is set, the residual nucleus from the (n,n′) reaction breaks up by emitting additional particles. This extra breakup energy changes the `q0` value. An example of such a section for ^{27}Al(n,n$_{25}$)p is shown in Fig. 11.

Starting with ENDF/B-VI, discrete-inelastic sections may also be given in File 6. Such sections contain their own photon production data, and the `heating` column will represent the entire recoil energy as in Eq. (100). (See below for detailed discussion of ENDF-6 output.)

For continuum reactions that use MF=4 and MF=5, such as (n,n′) or (n,2n), the neutron part of the display looks like the lines shown in Fig. 12. Once again, the photon effects will be subtracted later.

Absorption reactions such as (n,γ) or (n,p), lead to similar displays, but the particle `ebar` columns will always be set to zero (no emitted neutrons). An example from ^{27}Al is shown in Fig. 13.

If File 6 is present (which happens for evaluations in ENDF-6 format only), each reaction will be divided into subsections, one for each emitted particle. The neutron subsections are displayed as part of the energy-balance checks, but they do not contribute to KERMA or damage. The subsection for each charged particle or residual nucleus will give the incident energy, average energy for the emitted particle, cross section, heating contribution, and damage contribution as shown in Fig. 14. Note that the last subsection in this example was for emitted photons. Photons do not contribute to the KERMA or damage, but this information is used to check the total energy conservation for this reaction. The `ebal` lines show the difference between the available energy and the sum over all the outgoing particles. The values should be a small percentage of the total heating. If the `ebal` values are too large, there may be an error in the evaluation, or it may be necessary to refine the energy grids in the distributions. In addition, this photon production information is needed later for the photon energy check.

After all the sections corresponding to MT numbers in File 3 have been processed (using the File 4, File 4/5, or File 6 method as appropriate), the photon production sections in Files 12 and 13 are processed, if present. File 12 data are usually present for radiative capture (MT=102), at least at low energies. Simple files normally give a tabulated photon spectrum. The display gives the average energy for this spectrum in the `ebar` column and the negative contribution to the heating computed with Eq. (98) in the `heating` column. The `edam` column contains the $\overline{E_\gamma^2}$ term needed to compute the photon con-

```
neutron heating for mt  2   q0 =  0.0000e+00     q =  0.0000e+00
             e          ebar         yield          xsec       heating
    1.0000e-05    9.3053e-06    1.0000e+00    1.5004e+01    1.0424e-05
    1.1406e-04    1.0614e-04    1.0000e+00    4.5996e+00    3.6449e-05
    1.1406e-03    1.0614e-03    1.0000e+00    1.8953e+00    1.5019e-04
    1.1912e-02    1.1085e-02    1.0000e+00    1.4027e+00    1.1609e-03
    1.2812e-01    1.1922e-01    1.0000e+00    1.3521e+00    1.2036e-02
       ...
    4.3750e+02    4.0710e+02    1.0000e+00    1.3473e+00    4.0950e+01
    8.8750e+02    8.2584e+02    1.0000e+00    1.3472e+00    8.3068e+01
    3.0000e+03    2.7916e+03    1.0000e+00    1.3443e+00    2.8018e+02
    6.0000e+03    5.5832e+03    1.0000e+00    1.6952e+00    7.0663e+02
    1.2000e+04    1.1166e+04    1.0000e+00    1.1130e+00    9.2791e+02
    2.4000e+04    2.2333e+04    1.0000e+00    6.0689e-01    1.0119e+03
       ...
```

FIG. 10: Example of elastic heating for Al-27.

```
neutron heating for mt 75   q0 = -8.2710e+06     q = -1.0750e+07
             e          ebar         yield          xsec       heating
    1.2000e+07    7.8653e+05    1.0000e+00    8.2242e-02    2.4199e+05
    1.3000e+07    1.7116e+06    1.0000e+00    8.0121e-02    2.4176e+05
    1.4000e+07    2.6427e+06    1.0000e+00    5.9282e-02    1.8296e+05
    1.5000e+07    3.5864e+06    1.0000e+00    4.1834e-02    1.3147e+05
    1.6000e+07    4.5096e+06    1.0000e+00    2.8880e-02    9.2977e+04
    1.7000e+07    5.4335e+06    1.0000e+00    1.9867e-02    6.5472e+04
    1.8000e+07    6.3848e+06    1.0000e+00    1.3677e-02    4.5739e+04
    1.9000e+07    7.2944e+06    1.0000e+00    9.4771e-03    3.2550e+04
    2.0000e+07    8.2479e+06    1.0000e+00    6.6142e-03    2.3025e+04
```

FIG. 11: Example of heating for MT75 in Al-27.

tribution to the damage, which is given in the `damage` column. See Eq. (131). The display also has an extra line for each incident energy containing the percent error "`--- pc`" between the total photon energy as computed from File 12 and the value $E+Q-E/(A+1)$ computed from File 3. As discussed above, HEATR does not guarantee energy balance in large systems if this error occurs. The example shown in Fig. 15 shows some large errors due to mistakes in the ENDF/B-V evaluation for ^{55}Mn. Two columns labeled `edam` and `xsec` have been removed to show the `heating` and `damage` columns. The text has also been shifted to the left of its normal position to fit better on the printed page.

Many sections MF=12, MT=102 give multiplicities for the production of discrete photons. In these cases, HEATR prints out data for all of the parts, and it provides a sum at the end. The balance error is printed with the sum. The example in Fig. 16 shows a case with discrete photons. The last two columns have been removed (`heating`, `damage`), and the text has been compacted and shifted to the left to fit on the printed page. In this case (^{27}Al), the capture energy production checks out perfectly for the sum of all 89 discrete photons.

Other sections using either File 12 or File 13 generate displays similar to the one shown in Fig. 17. Note that the photon $\overline{E}\sigma$ is simply subtracted from the `heating` column for each incident energy.

If the partial KERMA `mtk=443` was requested in the user's input, HEATR will print out a special section that tests the total photon energy production against the kinematic limits. An example is shown in Fig. 18. The low and high kinematic limits will be the same at low energies where only kinematics affect the calculations. They may be the same for all energies for ENDF/B-VI or VII evaluations that provide complete distributions for all outgoing charged particles and recoil nuclei. Normally, the limits diverge above the threshold for continuum reactions. Note that HEATR marks lines where the computed value goes more than a little way outside the limits with the symbols `++++` or `----`. It is often convenient to extract these numbers from the output listing and plot them (see Fig. 19). Although the energy grid is a little coarse, such plots can often be useful (see below).

The last part of a full HEATR output listing is a tabulation of the computed KERMA and damage coefficients on the normal coarse energy grid. Columns are provided for the total KERMA and for each of the partial KERMA results requested with `mtk` values in the user's input. If kinematic checks were requested, the check values are written just above and below the corresponding partial KERMA values. In addition, `low` and `high` messages are written just above or just below the kinematic limits in every column where a significant violation of the limits occurs. Caution: if summation reactions (MT=3, MT=4) were used to define the photon production over some parts of the energy range, the partial KERMA results may not make sense at some energies. For example, consider the common pattern in ENDF/B-V where

```
neutron heating for mt 16   q0 = -1.3057e+07     q = -1.3057e+07
             e          ebar         yield          xsec       heating
    1.4000e+07    1.9960e+05    2.0000e+00    2.4000e-02    1.3051e+04
    1.5000e+07    6.6850e+05    2.0000e+00    1.2320e-01    7.4659e+04
    1.6000e+07    1.0855e+06    2.0000e+00    2.0710e-01    1.5987e+05
    1.7000e+07    1.4308e+06    2.0000e+00    2.6510e-01    2.8667e+05
    1.8000e+07    1.6379e+06    2.0000e+00    3.0300e-01    5.0518e+05
    1.9000e+07    1.7659e+06    2.0000e+00    3.3000e-01    7.9567e+05
    2.0000e+07    1.8755e+06    2.0000e+00    3.5000e-01    1.1172e+06
```

FIG. 12: Example of heating for the (n,2n) reaction in Al-27.

```
neutron heating for mt103   q0 = -1.8278e+06     q = -1.8278e+06
             e          ebar         yield          xsec       heating
    2.5000E+06    0.0000E+00    1.0000E+00    3.2800E-05    2.2048E+01
    3.0000E+06    0.0000E+00    1.0000E+00    1.3300E-03    1.5590E+03
    3.5000E+06    0.0000E+00    1.0000E+00    1.0100E-02    1.6889E+04
    4.0000E+06    0.0000E+00    1.0000E+00    6.9667E-03    1.5133E+04
    4.5000E+06    0.0000E+00    1.0000E+00    1.7000E-02    4.5427E+04
    5.0000E+06    0.0000E+00    1.0000E+00    2.3300E-02    7.3912E+04
       ...
    1.7000E+07    0.0000E+00    1.0000E+00    5.5200E-02    8.3751E+05
    1.8000E+07    0.0000E+00    1.0000E+00    4.7800E-02    7.7303E+05
    1.9000E+07    0.0000E+00    1.0000E+00    4.0200E-02    6.9032E+05
    2.0000E+07    0.0000E+00    1.0000E+00    3.2200E-02    5.8514E+05
```

FIG. 13: Example of heating for the (n,p) reaction in Al-27.

MT=102 is used for capture at low energies, but at higher energies, it is set to zero, and the capture contribution is included in MT=3 (nonelastic). Clearly, the partial KERMA MT=402 doesn't make sense above this breakpoint. Figure 20 shows part of the final KERMA listing for ENDF/B-V ^{55}Mn. The damage column was removed and the columns compressed to fit on the printed page.

The following subsection discusses how to analyze the "check" output of HEATR in order to diagnose energy-balance errors in ENDF-format evaluations. The examples are drawn from ENDF/B-V testing [20] because the energy-balance problems are more dramatic than in latter ENDF/B evaluations. Many of the types of balance errors illustrated here can still occur in newer evaluations, so these examples may still be helpful to current evaluators.

I. Diagnosing Energy-Balance Problems

The analysis should start with MT=102, because if it is wrong, the guarantee of energy conservation for large systems breaks down. If the display for MF=12,MT=102 shows messages of the form "`--- pc`", there may be a problem. If these messages only show up at the higher energies, and if the size of the error increases with energy, it is probable that the evaluator has used a thermal spectrum over the entire energy range (this is very common). Of course, the total photon energy production from radiative capture should equal

$$\frac{A}{A+1}E + Q \ , \tag{142}$$

where the rest of the total energy $E+Q$ is carried away by recoil. If only a thermal spectrum is given, the E term is being neglected, and errors will normally appear above about 1 MeV. The E term can be included in evaluations that use tabulated data by giving E-dependent spectra in File 15; and it can be included for evaluations that use discrete photons by setting the "primary photon" flags in File 12 properly. In practice, the capture cross sections above 1 MeV are often comparatively small due to the $1/v$ tendency of capture, and the errors introduced by neglecting the E term can be ignored.

If the MT=102 errors show up at low energies, there is probably an error in the average photon yield from File 12, in the average energy computed from File 15, or both. In the ^{55}Mn case shown above, the yield had been incorrectly entered. In addition, the spectrum didn't agree with the experimental data because the bin boundaries were shifted. Each case must be inspected in detail to find the problems.

The next common source of energy-balance errors in ENDF files arises from the representation used for inelastic scattering. Typically, the neutron scattering is described in detail using up to 40 levels for the (n,n′) reaction. However, the photon production is often described using MF=13/MT=3 or MF=13/MT=4 and rather coarse energy resolution. As a result, it is possible to find photons for (n,n$_1$) being produced for incident

```
file six heating for mt 28, particle =     1     q =  -8.2721E+06
             e          ebar         yield          xsec       heating
    9.0000E+06    1.2303E+05    1.0000E+00    1.0385E-03    0.0000E+00
    1.0000E+07    4.6746E+05    1.0000E+00    1.3526E-02    0.0000E+00
    ...
    1.8000E+07    3.1862E+06    1.0000E+00    3.7721E-01    0.0000E+00
    1.9000E+07    3.4535E+06    1.0000E+00    3.7577E-01    0.0000E+00
    2.0000E+07    3.7207E+06    1.0000E+00    3.7434E-01    0.0000E+00

file six heating for mt 28, particle =  1001     q =  -8.2721E+06
             e          ebar         yield          xsec       heating
    9.0000E+06    4.5909E+05    1.0000E+00    1.0385E-03    4.7677E+02
    1.0000E+07    8.9616E+05    1.0000E+00    1.3526E-02    1.2121E+04
    ...
    1.8000E+07    3.5193E+06    1.0000E+00    3.7721E-01    1.3275E+06
    1.9000E+07    3.8257E+06    1.0000E+00    3.7577E-01    1.4376E+06
    2.0000E+07    4.1321E+06    1.0000E+00    3.7434E-01    1.5468E+06

file six heating for mt 28, particle = 12026     q =  -8.2721E+06
             e          ebar         yield          xsec       heating
    9.0000E+06    3.2104E+05    1.0000E+00    1.0385E-03    3.3340E+02
    1.0000E+07    3.8540E+05    1.0000E+00    1.3526E-02    5.2128E+03
    ...
    1.8000E+07    8.0820E+05    1.0000E+00    3.7721E-01    3.0486E+05
    1.9000E+07    8.6147E+05    1.0000E+00    3.7577E-01    3.2372E+05
    2.0000E+07    9.1475E+05    1.0000E+00    3.7434E-01    3.4243E+05

file six heating for mt 28, particle =     0     q =  -8.2721E+06
             e          ebar         yield          xsec       heating
    9.0000E+06    1.8347E+05    2.8104E-06    1.0385E-03    0.0000E+00
                                                     ebal   -1.8204E+02
    1.0000E+07    4.4913E+05    3.0441E-05    1.3526E-02    0.0000E+00
                                                     ebal   -2.8626E+02
    ...
    1.8000E+07    1.8309E+06    1.2028E+00    3.7721E-01    0.0000E+00
                                                     ebal    4.5397E+03
    1.9000E+07    1.8856E+06    1.3695E+00    3.7577E-01    0.0000E+00
                                                     ebal    1.8336E+03
    2.0000E+07    1.9403E+06    1.5363E+00    3.7434E-01    0.0000E+00
                                                     ebal   -7.6798E+03
```

FIG. 14: Example of a reaction using the ENDF File 6 format with all the emitted particles described, including a calculation of any energy-balance errors.

```
photon energy (from yields) mf12, mt102
         e     ebar/err         egam ...      yield      heating       damage
1  continuum gammas
1.0000e-05   4.5088e+06   2.4237e+02 ... 2.4791e+00  -4.8477e+09   5.4975e+04
1.0000e-05     53.7 pc
1.0703e-04   4.5088e+06   2.4237e+02 ... 2.4791e+00  -1.4819e+09   1.6806e+04
1.0703e-04     53.7 pc
1.2520e-03   4.5088e+06   2.4237e+02 ... 2.4791e+00  -4.3347e+08   4.9159e+03
1.2520e-03     53.7 pc
1.4292e-02   4.5088e+06   2.4237e+02 ... 2.4791e+00  -1.2830e+08   1.4551e+03
1.4292e-02     53.7 pc
  ...
1.3571e+04   4.4522e+06   2.3887e+02 ... 2.4836e+00  -7.3869e+03  -1.2406e-01
1.3571e+04     51.8 pc
2.7142e+04   4.3957e+06   2.3536e+02 ... 2.4881e+00  -3.7037e+05  -1.6487e+01
2.7142e+04     49.9 pc
5.4287e+04   4.2826e+06   2.2834e+02 ... 2.4972e+00  -2.5604e+04  -2.5340e+00
5.4287e+04     46.0 pc
1.0858e+05   4.0563e+06   2.1430e+02 ... 2.5154e+00  -1.3327e+04  -2.7289e+00
1.0858e+05     38.3 pc
```

FIG. 15: Capture photon print out for an evaluation with an energy-balance problem.

```
photon energy (from yields) mf12, mt102
         e     ebar/err         egam         edam         xsec        yield
1   7.7260E+06 ev gamma
1.0000e-05   7.7260e+06   1.1448e+03   8.9780e+02   1.1677e+01   3.0000e-01
1.1406e-04   7.7260e+06   1.1448e+03   8.9780e+02   3.4574e+00   3.0000e-01
1.1406e-03   7.7260e+06   1.1448e+03   8.9780e+02   1.0934e+00   3.0000e-01
  ...
2.0000e+07   2.7005e+07   1.1448e+03   8.9780e+02   1.0000e-03   3.0000e-01
2   7.6950e+06 ev gamma
1.0000e-05   7.6950e+06   1.1356e+03   8.9091e+02   1.1677e+01   5.0000e-02
1.1406e-04   7.6950e+06   1.1356e+03   8.9091e+02   3.4574e+00   5.0000e-02
  ...
2.0000e+07   2.6974e+07   1.1356e+03   8.9091e+02   1.0000e-03   5.0000e-02
3   6.8630e+06 ev gamma
1.0000e-05   6.8630e+06   9.0330e+02   7.1515e+02   1.1677e+01   1.2000e-03
1.1406e-04   6.8630e+06   9.0330e+02   7.1515e+02   3.4574e+00   1.2000e-03
1.1406e-03   6.8630e+06   9.0330e+02   7.1515e+02   1.0934e+00   1.2000e-03
  ...
89   3.1000e+04 eV gamma
1.0000e-05   3.1000e+04   1.8430e-02   0.0000e+00   1.1677e+01   2.8884e-01
1.0000e-05      0.0 pc
1.1406e-04   3.1000e+04   1.8430e-02   0.0000e+00   3.4574e+00   2.8884e-01
1.1406e-04      0.0 pc
1.1406e-03   3.1000e+04   1.8430e-02   0.0000e+00   1.0934e+00   2.8884e-01
1.1406e-03      0.0 pc
1.1912e-02   3.1000e+04   1.8430e-02   0.0000e+00   3.3832e-01   2.8884e-01
1.1912e-02      0.0 pc
  ...
2.0000e+07   3.1000e+04   1.8430e-02   0.0000e+00   1.0000e-03   2.8884e-01
2.0000e+07      0.0 pc
```

FIG. 16: Example of a capture reaction with both discrete and continuum parts, including a test of the sum over all the parts.

```
photon energy (from xsecs) mf13, mt  3
             e          ebar          xsec        energy       heating
    1  continuum gammas
    2.0000e+05    3.6753e+06    4.2076e-03    1.5464e+04   -1.5464e+04
    4.0500e+05    3.3863e+06    5.2873e-03    1.7904e+04   -1.7904e+04
    6.0031e+05    3.1097e+06    6.4478e-03    2.0051e+04   -2.0051e+04
    8.0182e+05    2.0089e+06    9.3236e-02    1.8730e+05   -1.8730e+05
    1.0000e+06    9.2622e+05    1.7859e-01    1.6541e+05   -1.6541e+05
    1.2000e+06    9.6151e+05    2.8329e-01    2.7239e+05   -2.7239e+05
  ...
```

FIG. 17: Example of a section of photon production.

```
photon energy production check
             e      ev-barns          min          max
    1.0000e-05    9.0215e+07    9.0187e+07    9.0200e+07
    1.1406e-04    2.6712e+07    2.6704e+07    2.6708e+07
    1.1406e-03    8.4479e+06    8.4453e+06    8.4466e+06
    1.1912e-02    2.6138e+06    2.6130e+06    2.6134e+06
    1.2812e-01    7.9895e+05    7.9871e+05    7.9883e+05
    1.2812e+00    2.5211e+05    2.5203e+05    2.5207e+05
    2.6875e+00    1.7420e+05    1.7415e+05    1.7417e+05
    5.5000e+00    1.2186e+05    1.2182e+05    1.2184e+05
    1.1406e+01    8.4662e+04    8.4636e+04    8.4648e+04
    2.4062e+01    5.8231e+04    5.8213e+04    5.8222e+04
    4.9375e+01    4.0614e+04    4.0601e+04    4.0607e+04
    1.0000e+02    2.8522e+04    2.8514e+04    2.8518e+04
  ...
    8.0000e+06    3.8972e+06    3.7964e+06    4.4880e+06
    9.0000e+06    4.4782e+06    4.4401e+06    5.4986e+06
    1.0000e+07    4.9645e+06    5.2078e+06    6.6176e+06
    1.1000e+07    5.3712e+06    6.1302e+06    7.8374e+06    ----
    1.2000e+07    5.3212e+06    5.8699e+06    8.0618e+06
    1.3000e+07    5.0984e+06    5.9333e+06    8.5312e+06    ----
    1.4000e+07    4.7415e+06    5.7172e+06    8.6605e+06    ----
    1.5000e+07    4.0795e+06    4.7419e+06    8.0409e+06    ----
    1.6000e+07    3.2521e+06    3.6806e+06    7.2734e+06    ----
    1.7000e+07    2.8079e+06    2.8418e+06    6.6927e+06
    1.8000e+07    2.7492e+06    2.2915e+06    6.2850e+06
    1.9000e+07    2.9626e+06    1.9674e+06    6.2330e+06
    2.0000e+07    3.4419e+06    1.7938e+06    6.1994e+06
```

FIG. 18: Example of the photon energy-production with its kinematic limits.

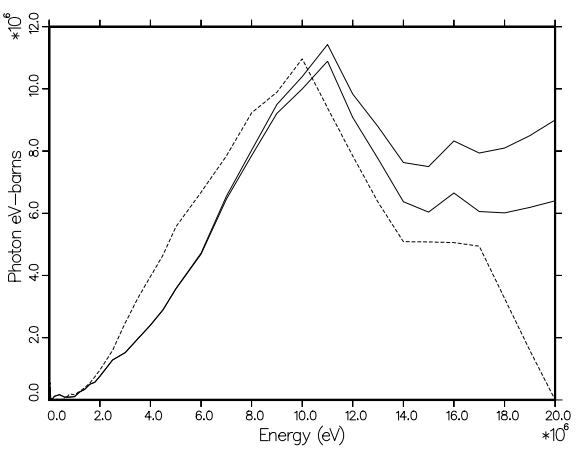

FIG. 19: Example of a plot comparing the total photon energy production for ^{55}Mn from ENDF/B-V.1 (dashed) with the kinematic limits (solid).

neutron energies slightly below the MT=51 threshold! These photons would lead to a spike of negative KERMA factors. A more common effect of the coarse grid used for photon production is to lead to an underestimate or overestimate of the photon production by not following the detailed shape of the inelastic cross section. The HEATR "kinematic KERMA" is correct in this range since only two-body reactions are active. Therefore, a plot of MT=301 and MT=443 on the same frame normally shows these effects in detail. Figure 21 is an example of such a plot. Figure 22 shows both the inelastic cross section from File 3 and the photon production cross section from File 13 to demonstrate the mismatch in the energy grids that contributes to the energy-balance errors. These kinds of errors are best removed by changing to a representation that uses File 12 to give photon production yields for the separate reactions MT=51, MT=52, etc. This representation makes full use of the File 3 cross sections, and as long as each section of File 12 conserves energy, the total inelastic reaction is guaranteed to conserve energy, even at the finest energy resolution.

A method that is frequently used by evaluators of photon production files is to select a number of nonelastic photon spectra on a fairly coarse incident-energy grid using theory or experiment, and then to readjust the photon yield on this energy grid so as to conserve energy at each grid point. However, the results do not, in general, conserve energy at intermediate points. If a very coarse energy grid is used for File 13, quite large deviations between MT=301 and MT=443 can result. Figure 23 shows such a case. The solution to this kind of violation of energy balance is to add intermediate points in Files 13 and 15 until the magnitude of the deviations is small enough for practical calculations.

Especially large energy-balance errors of this type are caused by interpolating across the minimum formed by the decreasing capture heating and the increasing inelastic heating. Figure 24 shows a dramatic example using a photon energy production comparison.

For energies above the threshold for continuum reac-

```
final kerma factors
          e          301          302          303          402          443          444

        min   1.2400e-04   3.7775e-06   1.2022e-04   1.2022e-04
 1.0000e-05   4.0068e+05   3.7775e-06   4.0068e+05   4.0068e+05   4.0068e+05
        max   3.3820e+05   3.7775e-06   3.3820e+05   3.3820e+05
                high                       high         high

        min   4.0650e-04   1.3174e-05   3.9333e-04   3.9333e-04
 1.0703e-04   1.2248e+05   1.3174e-05   1.2248e+05   1.2248e+05   1.2248e+05
        max   1.0338e+05   1.3174e-05   1.0338e+05   1.0338e+05
                high                       high         high

 ...

        min   3.2373e+04   2.3497e+04   8.8759e+03   4.1114e+01
 6.0031e+05   1.0017e+05   2.3497e+04   7.6673e+04   2.9899e+04   3.2375e+04
        max   3.2375e+04   2.3497e+04   8.8781e+03   4.3366e+01
                high                       high         high

                 low                       low
        min   7.3041e+04   5.9403e+04   1.3638e+04   4.4748e+01
 8.0182e+05  -2.4355e+04   5.9403e+04  -8.3758e+04   2.4987e+04   7.3043e+04
        max   7.3043e+04   5.9403e+04   1.3640e+04   4.6676e+01
                                                       high

                low                       low
        min   9.8973e+04   7.7075e+04   2.1898e+04   4.7777e+01
 1.0000e+06   3.7682e+04   7.7075e+04  -3.9393e+04   2.1917e+04   9.8974e+04
        max   9.8974e+04   7.7075e+04   2.1900e+04   4.9509e+01
                                                       high

                low                       low
        min   1.1397e+05   7.7800e+04   3.6168e+04   4.9760e+01
 1.2000e+06   9.5321e+04   7.7800e+04   1.7521e+04   1.9482e+04   1.1397e+05
        max   1.1397e+05   7.7800e+04   3.6169e+04   5.1335e+01
                                                       high

                low                       low
        min   1.4632e+05   1.0192e+05   4.4402e+04   5.3005e+01
 1.4000e+06   9.8251e+04   1.0192e+05  -3.6667e+03   1.8208e+04   1.4632e+05
        max   1.4632e+05   1.0192e+05   4.4403e+04   5.4511e+01
                                                       high

 ...

        min   2.3650e+05   1.6907e+05   6.7426e+04   1.7607e+02
 1.9000e+07   7.1384e+06   1.6907e+05   6.9693e+06   1.3503e+04   2.5435e+06
        max   2.5435e+06   1.6907e+05   2.3744e+06   1.7939e+02
                high                       high         high

        min   2.4001e+05   1.8261e+05   5.7406e+04   1.4423e+02
 2.0000e+07   9.2369e+06   1.8261e+05   9.0543e+06   1.0908e+04   2.8385e+06
        max   2.8385e+06   1.8261e+05   2.6559e+06   1.4701e+02
                high                       high         high
```

FIG. 20: Example of print out of partial kermas and kinematic limits.

tions like (n,n′) or (n,2n), it is difficult to use the results of the kinematic checks to fix evaluations. The representation of Eqs. (101) and (102) for continuum inelastic scattering is very rough. Comparison to other more accurate methods suggests that a CM formula would be better here [27], even though the ENDF file says "lab." Most other reactions give very wide low and high limits. Two exceptions are (n,2n) and (n,3n). If they dominate the cross section, the kinematic limits will be fairly close together. In the 14 MeV range, energy errors could be in the photon data, the neutron data, or both. The best way to eliminate balance errors is to construct a new evaluation based on up-to-date nuclear model codes.

Many of the energy-balance problems cited here only occur in older evaluations (*e.g.* ENDF/B-V and ENDF/B-VI), having been corrected in the more mod-

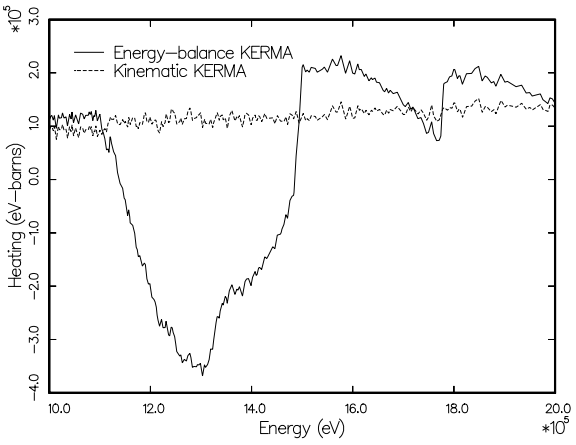

FIG. 21: Comparison of MT=301 with MT=443 for the region of the discrete-inelastic thresholds for ^{59}Co from ENDF/B-V.2. Note the large region of negative KERMA. The best way to remove this kind of problem is by using yields in File 12, MT=51, 52, 53, ... to represent the photon production.

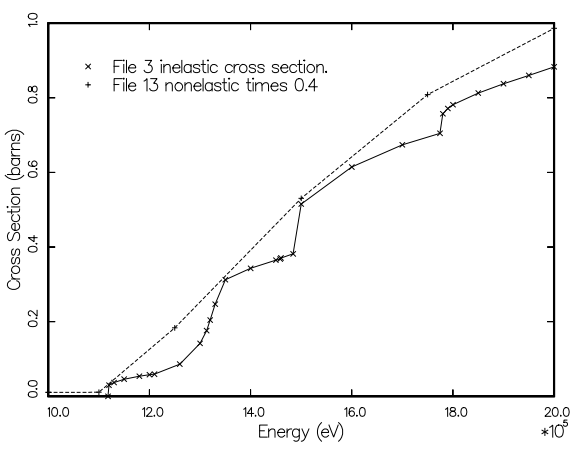

FIG. 22: Plot showing the mismatch between the energy grids used for File 3 and File 13 in the region of the thresholds for discrete-inelastic scattering levels for the case shown in Fig. 21. The cross and ex symbols show the actual grid energies in the evaluation.

ern ENDF/B-VII files. However, problems still remain. For the current status, see the ENDF/B-VII Energy Balance web page [28].

V. THERMR

The THERMR module generates pointwise neutron scattering cross sections in the thermal energy range and adds them to an existing PENDF file. The cross sections can then be group-averaged, plotted, or reformatted in subsequent modules. THERMR works with either the

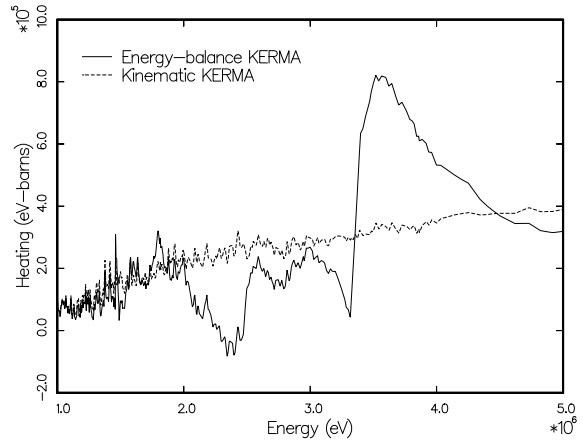

FIG. 23: Typical energy-balance problems between points where balance is satisfied. Discrete photons were used below about 2 MeV, and energy balance is reasonably good there. The energy points in MF=13 for the continuum part are at 2, 3, and 5 MeV, and the balance is also good at those energies. Clearly, a grid in File 13 that used steps of about 0.25 MeV between 2 and 4 MeV would reduce the size of the deviations substantially and remove the negative KERMA factors.

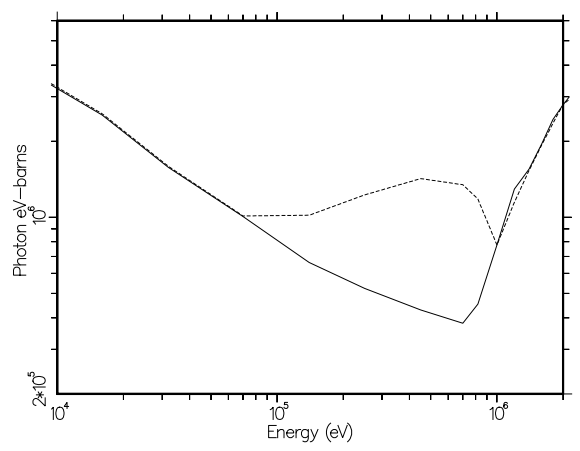

FIG. 24: Computed photon energy production (dashed) compared with the kinematic value (solid) for ^{93}Nb from ENDF/B-V. The original File 13 has grid points at 100 keV and 1 MeV. Interpolating across that wide bin gives a photon production rate that is much too large for energies in the vicinity of a few hundred keV. This will result in a large region of negative heating numbers. Since this is just the region of the peak flux in a fast reactor, niobium-clad regions could be cooled instead of heated!

original ENDF/B-III thermal format and data files [29] (which were also used for ENDF/B-IV and -V), or the newer ENDF-6 format. Coherent elastic cross sections are generated for crystalline materials using either parameters given in an ENDF-6 format evaluation or an extended version of the method of HEXSCAT [30]. Inco-

herent elastic cross sections for non-crystalline materials such as polyethylene and ZrH can be generated either from parameters in an ENDF-6 format file or by direct evaluation using parameters included in the THERMR coding. Inelastic cross sections and energy-to-energy transfer matrices can be produced for a gas of free atoms, or for bound scatterers when ENDF $S(\alpha,\beta)$ scattering functions are available. An earlier code that performed this function was FLANGE-II [31].

A. Coherent Elastic Scattering

In crystalline solids consisting of coherent scatters—for example, graphite—the so-called "zero-phonon term" leads to interference scattering from the various planes of atoms of the crystals making up the solid. There is no energy loss in such scattering, and the ENDF term for the reaction is coherent elastic scattering. The cross section may be represented as follows:

$$\sigma^{\rm coh}(E,E',\mu) = \frac{\sigma_c}{E}\sum_{E_i<E} f_i\,{\rm e}^{-2WE_i}\,\delta(\mu-\mu_0)\,\delta(E-E')\ , \quad (143)$$

where

$$\mu_0 = 1-\frac{E_i}{E}\ , \quad (144)$$

and the integrated cross section is given by

$$\sigma^{\rm coh} = \frac{\sigma_c}{E}\sum_{E_i>E} f_i\,{\rm e}^{-2WE_i}\ . \quad (145)$$

In these equations, E is the incident neutron energy, E' is the secondary neutron energy, μ is the scattering cosine in the laboratory (LAB) reference system, σ_c is the characteristic coherent cross section for the material, W is the effective Debye-Waller coefficient (which is a function of temperature), the E_i are the so-called "Bragg edges", and the f_i are related to the crystallographic structure factors.

It can be seen from Eq. (145) and the example in Fig. 25 that the coherent elastic cross section is zero before the first Bragg edge, E_1 (typically about 2 to 5 meV). It then jumps sharply to a value determined by f_1 and the Debye-Waller term. At higher energies, the cross section drops off as $1/E$ until E=E_2. It then takes another jump and resumes its $1/E$ drop-off. The sizes of the steps in the cross section gradually get smaller, and at high energies there is nothing left but an asymptotic $1/E$ decrease (typically above 1 to 2 eV).

For evaluations in the ENDF-6 format, the section MF=7/MT=2 contains the quantity $E\sigma^{\rm coh}(E)$ as a function of energy and temperature. The energy dependence is given as a histogram with breaks at the Bragg energies. The cross section is easily recovered from this representation by dividing by E. The E_i are easily found as the

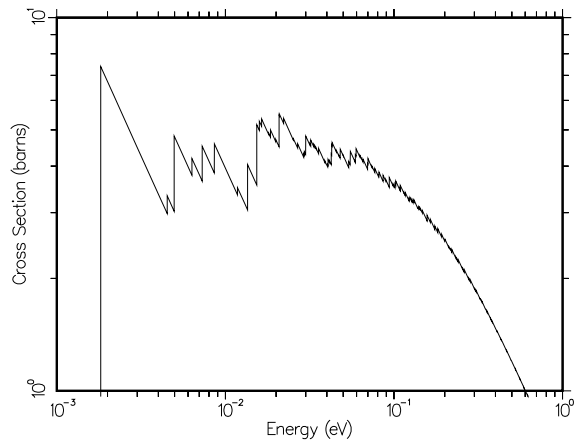

FIG. 25: Typical behavior of the coherent elastic scattering cross section for a crystalline material as computed by THERMR. This cross section is for graphite at 300 K.

tabulation points of the function, and the f_i for a point can be obtained by subtracting the value at the previous point.

For evaluations using the older ENDF/B-III thermal format, it is necessary to compute the E_i and f_i in THERMR. The methods used are based on HEXSCAT and work only for the hexagonal materials graphite, Be, and BeO. The Bragg edges are given by

$$E_i = \frac{\hbar^2\tau_i^2}{8m}\ , \quad (146)$$

where τ_i is the length of the vectors of one particular "shell" of the reciprocal lattice, and m is the neutron mass. The f_i factors are given by

$$f_i = \frac{\pi^2\hbar^2}{2mNV}\sum_{\rm shell}|F(\tau)|^2\ , \quad (147)$$

where the shell sum extends over all reciprocal lattice vectors of the given length, N is the number of atoms in the unit cell, and F is the crystallographic structure factor. The calculation works by preparing a sorted list of precomputed τ_i and f_i values. As τ_i gets large, the values of τ_i get more and more closely spaced. In order to save storage and run time, a range of τ values can be lumped together to give a single effective τ_i and f_i. This device washes out the Bragg edges at high energies while preserving the proper average cross section and angular dependence. The current grouping factor is 5% (see `eps` in `sigcoh`).

Lattice constants (given in `sigcoh` for graphite, Be, and BeO), form factor formulas (see `form`), Debye-Waller coefficients, and methods for computing reciprocal lattice vectors were borrowed directly from HEXSCAT.

The energy grid for E is obtained adaptively (see `coh`). A panel extending from just above one Bragg edge to

just below the next higher edge is subdivided by successive halving until linear interpolation is within a specified fractional tolerance (`tol`) of the exact cross section at every point. This procedure is repeated for every panel from the first Bragg edge to the specified maximum energy for the thermal treatment (`emax`).

The code usually computes and writes out the cross section of Eq. (145), the average over μ of Eq. (143), which is sometimes called the P_0 cross section. Subsequent modules can deduce the correct discrete scattering angles μ_0 from the location of the Bragg edges E_i and the factors f_i from the cross-section steps at the Bragg edges (see the explanation of this in the GROUPR section).

B. Incoherent Inelastic Scattering

In ENDF/B notation, the thermal incoherent scattering cross section is given by

$$\sigma^{\rm inc}(E,E',\mu) = \frac{\sigma_b}{2kT}\sqrt{\frac{E'}{E}}\,{\rm e}^{-\beta/2}S(\alpha,\beta)\ , \qquad (148)$$

where E is the initial neutron energy, E' is the energy of the scattered neutron, μ is the scattering cosine in the laboratory system, σ_b is the characteristic bound incoherent cross section for the nuclide, T is the Kelvin temperature, β is the dimensionless energy transfer,

$$\beta = \frac{E'-E}{kT}\ , \qquad (149)$$

α is the dimensionless momentum transfer,

$$\alpha = \frac{E'+E-2\mu\sqrt{EE'}}{AkT}\ , \qquad (150)$$

k is Boltzmann's constant, and A is the ratio of the scatter mass to the neutron mass. The bound scattering cross section is usually given in terms of the characteristic free cross section, σ_f,

$$\sigma_b = \sigma_f\frac{(A+1)^2}{A^2}\ . \qquad (151)$$

The scattering law $S(\alpha,\beta)$ describes the binding of the scattering atom in a material. For a free gas of scatterers with no internal structure

$$S(\alpha,\beta) = \frac{1}{\sqrt{4\pi\alpha}}\exp\left\{-\frac{\alpha^2+\beta^2}{4\alpha}\right\}\ . \qquad (152)$$

For binding in solids and liquids, $S(\alpha,\beta)$ for a number of important moderator materials is available in ENDF File 7 format. The scattering law is given as tables of S versus α for various values of β. Values of S for other values of α and β can be obtained by interpolation. The scattering law is normally symmetric in β and only has to be tabulated for positive values, but for materials like orthohydrogen and parahydrogen of interest for cold moderators at neutron scattering facilities, this is not true. These kinds of materials are identified by the ENDF-6 `LASYM` option, and THERMR assumes that the scattering law is given explicitly for both positive and negative values of β.

If the α or β required is outside the range of the table in File 7, the differential scattering cross section can be computed using the short collision time (SCT) approximation

$$\sigma^{\rm SCT}(E,E',\mu) = \frac{\sigma_b}{2kT}\frac{\sqrt{E'/E}}{\sqrt{4\pi\,\alpha\,T_{\rm eff}/T}} \times\exp\left\{-\frac{(\alpha-|\beta|)^2}{4\,\alpha}\frac{T}{T_{\rm eff}}-\frac{\beta+|\beta|}{2}\right\}\ , \qquad (153)$$

where $T_{\rm eff}$ is the effective temperature for the SCT approximation. These temperatures are available [29] for the older ENDF/B-III evaluations; they are usually somewhat larger than the corresponding Maxwellian temperature T. For the convenience of the user, the values of $T_{\rm eff}$ for the common moderators are included as defaults (see input instructions). For the ENDF-6 format, the effective temperatures are included in the data file. However, there is a complication. Some evaluations give $S(\alpha,\beta)$ for a molecule or compound (in the ENDF/B-III files, these cases are BeO and C_6H_6). The corresponding SCT approximation must contain terms for both atoms. The two sets of bound cross sections and effective temperatures are included in the data statements in THERMR, and they can be given in the new ENDF-6 format if desired.

THERMR expects the requested temperature T to be one of the temperatures included on the ENDF/B thermal file, or within a few degrees of that value (296 K is used if 300 K is requested). Intermediate temperatures should be obtained by interpolating between the resulting cross sections and not by interpolating $S(\alpha,\beta)$.

The normal output format for THERMR orders its results as E, E', μ (`iform=0`). For this format, the secondary energy grid for incoherent scattering is obtained adaptively. A stack is first primed with the point at zero and the first point above zero that can be derived from the positive and negative values of β from the evaluator's β grid using Eq. (149). (For free-gas scattering, the β grid is taken to have nine starting points ranging from 0 to 25). This interval is then subdivided by successive halving until the cross section obtained by linear interpolation is within the specified tolerance of the correct cross section. The next highest energy derived from the β grid is then calculated, and the subdivision process is repeated for this new interval. This process is continued until the β grid has been exhausted. Excess points with zero cross section are removed before writing the spectrum into File 6. This procedure is sure to pick up all the structure in the evaluation; giving points related to the β grid avoids excessive work in trying to fit sharp corners introduced by breaks in the interpolation of $S(\alpha,\beta)$. Figure 26 shows how the procedure picks up features resulting from the sharp excitation features in the graphite

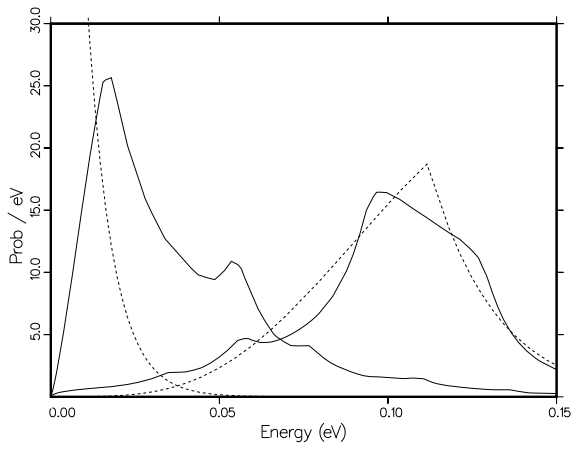

FIG. 26: Adaptive reconstruction of two of the emission curves for graphite at 300 K (E=.00016 eV to the left, and E=.1116 eV to the right). Note the presence of excitation features from the phonon frequency spectrum for both upscatter and downscatter. The breaks in the curves are due to β interpolation in $S(\alpha, \beta)$ and not to the tolerances in the reconstruction process. The dashed curves are the corresponding free gas results.

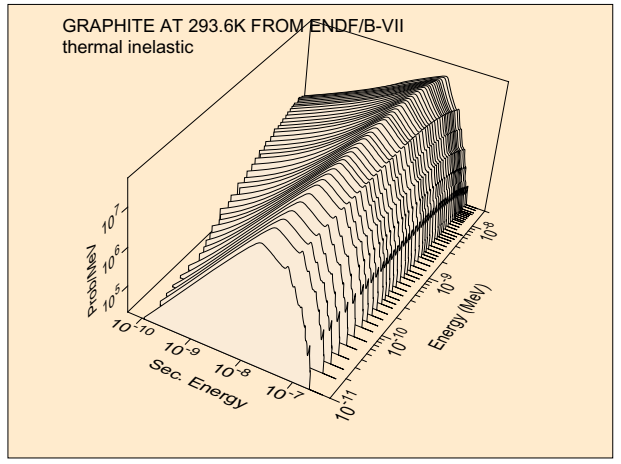

FIG. 27: Neutron distribution for incoherent inelastic scattering from graphite (T = 293.6 K).

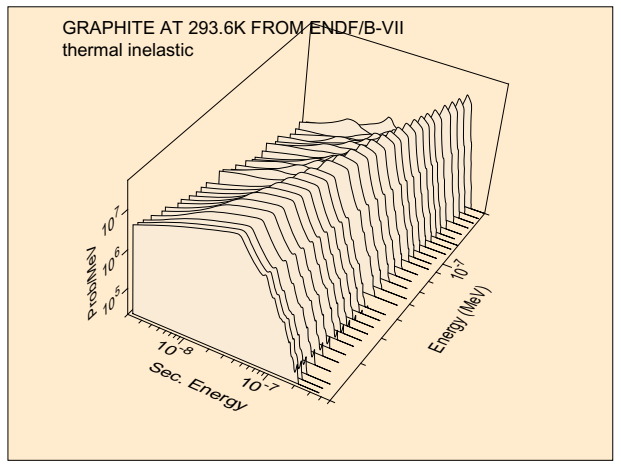

FIG. 28: Expanded view of the high-energy region of the graphite incoherent inelastic distribution.

phonon frequency distribution. The result of this adaptive reconstruction is easily integrated by the trapezoid rule to find the incoherent cross section at energy E.

The cross section for one particular $E \rightarrow E'$ is the integral over the angular variable of Eq. (148). The angular dependence is obtained by adaptively subdividing the cosine range until the actual angular function is represented by linear interpolation to within a specified tolerance. The integral under this curve is used in calculating the secondary-energy dependence as described above. Rather than providing the traditional Legendre coefficients, THERMR divides the angular range into equally probable cosine bins and then selects the single cosine in each bin that preserves the average cosine in the bin. These equally probable cosines can be converted to Legendre coefficients easily when producing group constants, and they are suitable for direct use in Monte Carlo codes. For strongly peaked functions, such as scattering for $E \gg kT$ when the result begins to look "elastic", all the discrete angles will be bunched together near the scattering angle defined by ordinary kinematics. This behavior cannot be obtained with ordinary P_3 Legendre coefficients. Conversely, if such angles are converted to Legendre form, very high orders can be used.

The incident energy grid is currently stored directly in the code. The choice of grid for $\sigma^{\rm inc}(E)$ is not critical since the cross section is a slowly varying function of E. However, the energy grid would seem to be important for the emission spectra. In order to demonstrate the problem, two perspective plots of the full energy distribution of incoherent inelastic scattering from graphite at 293.6 K are shown in Figs. 27 and 28. The second plot is simply an expansion of the high-energy region of the distribution.

It is clear that $\sigma^{\rm inc}(E, E')$ for one value of E' is a very strongly energy-dependent function for the higher incident energies. However, as shown in Fig. 28, the shape of the secondary energy distribution changes more slowly, with the peak tending to follow the line $E'=E$. This behavior implies that a relatively coarse incident energy grid might prove adequate if a suitable method is used to interpolate between the shapes at adjacent E values. One such interpolation scheme is implemented in GROUPR. The use of discrete angles is especially suitable for this interpolation scheme.

Strictly speaking, the scattering law for free-gas scattering given in Eq. (152) is only applicable to scatterers with no internal structure. However, many materials of interest in reactor physics have strong scattering resonances in the thermal range (for example, ^{240}Pu and ^{135}Xe). The Doppler broadened elastic cross section

produced by BROADR is formally correct for a gas of resonant scatterers, but the cross section resulting from Eq. (152) is not. In order to allow for resonance scattering in a way that at least provides the correct total cross section, THERMR renormalizes the free-atom scattering to the broadened elastic cross section. The secondary energy distribution will still be incorrect.

THERMR also supports a different output format (`iform=1`) using the ordering E, μ, E'. This ordering is more like the data reported from experiments, and this option can be used for comparisons to experimental data. The output is written in the MF6/Law7 format. The calculation starts by selecting some initial cosine values. For each value, the spectrum *vs* E' is generated adaptively as described above. The integral of the spectrum is the angular cross section. The cosine range is then adaptively subdivided to obtain the full μ dependence of the angular cross section with linear interpolation. There is a secondary energy distribution given for each of these cosine points. This entire process is repeated for each incident energy to obtain the final result.

C. Incoherent Elastic Scattering

In hydrogenous solids, there is an elastic (no energy loss) component of scattering arising from the zero-phonon term that can be treated in the incoherent approximation because of the large incoherent cross section of hydrogen. The ENDF term for this process is incoherent elastic scattering, and it is found in the materials polyethylene and zirconium hydride. The differential cross section is given by

$$\sigma^{\rm iel}(E, E', \mu) = \frac{\sigma_b}{2} {\rm e}^{-2WE(1-\mu)}\, \delta(E - E') \ , \qquad (154)$$

where σ_b is the characteristic bound cross section and W is the Debye-Waller coefficient. The energy grid of the elastic cross section is used for E, and the average cross section and equally probable angles are computed using

$$\sigma^{\rm iel}(E) = \frac{\sigma_b}{2} \left\{ \frac{1 - {\rm e}^{-4WE}}{2WE} \right\} \ , \qquad (155)$$

and

$$\begin{aligned} \bar{\mu}_i &= \frac{N}{2WE} \Big[{\rm e}^{-2WE(1-\mu_i)} (2WE\mu_i - 1) \\ &- {\rm e}^{-2WE(1-\mu_{i-1})} (2WE\mu_{i-1} - 1) \Big] \\ &/ \ (1 - {\rm e}^{-4WE}) \ , \end{aligned} \qquad (156)$$

where

$$\mu_i = 1 + \frac{1}{2WE} \ln \left[\frac{1 - {\rm e}^{-4WE}}{N} + {\rm e}^{-2WE(1-\mu_{i-1})} \right] \qquad (157)$$

is the upper limit of one equal probability bin and $\bar{\mu}_i$ is the selected discrete cosine in this bin. Here N is the number of bins and μ_0 is -1.

TABLE III: Moderator materials available for ENDF/B-VII thermal data files showing their MAT numbers and the special MT numbers used by NJOY. These MT numbers are used in MATXSR to define names for the reactions. If MATXSR is not being run, arbitrary values can be used for the MT numbers. Some standard MT numbers have not been defined as yet, but these materials have been used with MCNP.

Material	MAT	MTs	elastic	secondary
Al	45	243,244	coh	
Be	26	231,232	coh	
Be(BeO)	27	233,234	coh	
O(BeO)	28	237,238	coh	
C(graphite)	31	229,230	coh	
Fe	56	245,246	coh	
H(CH_2)	37	223	iel	free C
H(liquidCH_4)	33			
H(solidCH_4)	34			
C_6H_6	40	227		none
D(D_2O)	11	228		free O
D(paraD_2)	12			
D(orthoD_2)	13			
H(H_2O)	1	222		free O
H(paraH_2)	2			
H(orthoH_2)	3			
Zr(ZrH_n)	58	235,236	iel	
H(ZrH_n)	7	225,226	iel	
U(UO2)	76	241,242	coh	
O(UO2)	75	239,240	coh	

The characteristic bound cross sections and the Debye-Waller coefficients can be read from MF=7/MT=2 of an ENDF-6 format evaluation, or obtained directly from data statements in the code for the older format.

D. Using the ENDF/B-VII Thermal Data Files

The ENDF/B-VII thermal data files have been extended somewhat from those originally prepared for ENDF/B-III (which were also used for ENDF/B-IV and ENDF/B-V). Table III summarizes the thermal materials now available.

For most of these evaluations, THERMR will produce cross sections appropriate for the major scattering atom bound in a particular material, for example, hydrogen bound in water, or Zr bound in ZrH. In these cases, the cross sections are combined later (for example, hydrogen bound in water would be combined with free-gas oxygen, and Zr bound in ZrH would be combined with H bound in ZrH). The treatment to be used for the secondary scattering atoms for each evaluation is indicated in the table.

In one case—Benzine—the scattering laws $S(\alpha, \beta)$ for the two component atoms has been combined into a single scattering law normalized to be used with the cross section of the primary scattering atom. For these cases, THERMR produces a cross section for the molecule or compound directly. In making a macroscopic cross section for benzine, the user would multiply the thermal

benzine cross section from THERMR by the atomic density of H, taking care not to add any additional thermal contribution for the oxygen.

THERMR labels the thermal cross sections that it generates with specially defined MT numbers. The particular numbers shown in the table are recognized by the reaction naming logic in MATXSR (see section XVIII). Note that two numbers are defined for materials that have both inelastic and elastic components; the first number is for inelastic, and the second for elastic.

The plots in Fig. 27 and 28 were prepared during ACER processing of the ENDF/B-VII thermal data. The sets of plots include cross sections, average cosine, average energy, the perspective plots of scattering distributions, and perspective plots of angle-energy distributions at selected incident energies [32].

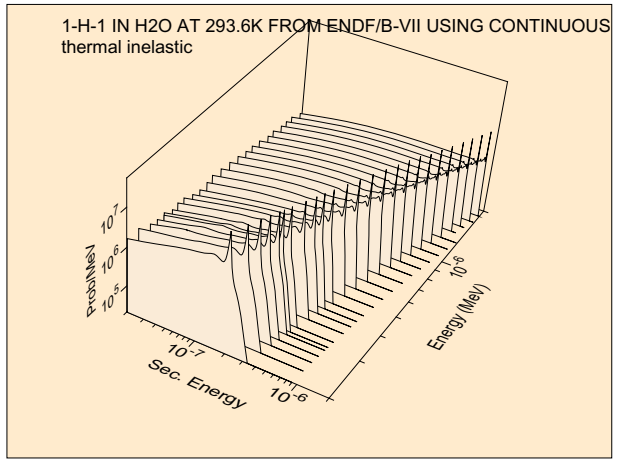

FIG. 29: Expanded view of the high-energy region of the incoherent inelastic distribution for hydrogen bound in water. Note the sharp quasi-elastic peak at E=E'.

E. Running THERMR

See the comments in the source code, the NJOY manuals, or the NJOY web areas for detailed user input instructions for THERMR. The following sample problem illustrates the production of thermal cross sections for hydrogen in water. The line numbers are for reference and aren't part of the input.

```
1. thermr
2. 20 21 22 /
3. 1 125 8 2 2 0 0 2 222 0
4. 293.6 500/
5. .01 4.6
6. stop
```

It is assumed that the thermal data for H in H_2O is mounted on unit 20, and that a previously prepared PENDF file for ^{1}H (MAT125) is mounted on unit 21. The thermal cross sections for hydrogen in water will be written on unit 22 using `mtref=222`. On Card 3, thermal material 1 is requested (H in H_2O), ENDF material 125 is requested (H-1), 8 angular bins are requested, two temperatures are requested, inelastic is set to read in $S(\alpha,\beta)$ data, the elastic option is set to none, output is to be in E, E', μ ordering, `natom` is set to 2 because the water molecule H_2O contains two hydrogen atoms, the thermal MT number is to be 222, and no additional printed information was requested. Card 4 gives the two temperatures desired. Card 5 says to use 1% reconstruction of the distributions and to limit the incident energies to 4.6 eV.

The higher-energy parts of the neutron emission curves for this example are shown in Fig. 29. The sharp peak at E=E' is quasi-elastic scattering broadened by diffusion.

A calculation of both free and graphite cross sections for ENDF/B-VII carbon would go as follows:

```
 1. reconr
 2. 20 22 / tape20 is ENDF/B-VII carbon
 3. /
 4. 600 1/
 5. .001/
 6. '6-C-nat from ENDF/B-VII'/
 7. 0/
 8. broadr
 9. 20 22 23
10. 600 1/
11. .001/
12. 293.6
13. 0/
14. thermr
15. 0 23 24
16. 0 600 8 1 1 0 0 1 221 0/
17. 293.6
18. .005 5/
19. thermr
20. 26 24 25 / tape26 is ENDF/B-VII graphite
21. 31 600 8 1 2 1 0 1 229 0
22. 293.6
23. .005 5/
24. stop
```

First, the ENDF/B-VII carbon file must be mounted on unit 20. Similarly, graphite from the ENDF/B-VII thermal sublibrary must be copied to `tape26`. Next, RECONR is run to linearize the evaluation, and BROADR is run to prepare 293.6K (.0253 eV) cross sections. The first THERMR run is for free-gas scattering (`mtref=221`), and the second run is for carbon bound in graphite (`mtref=229`). Note that `natom` is now 1, and that 8 discrete angles were requested in both cases. For graphite, `icoh` is set to 1 in order to request the calculation of coherent elastic scattering like that shown in Fig. 25. The coherent results will use MF=3 and MT=230. Distributions will be calculated to .5% accuracy for energies up to 5 eV. The final PENDF output will be `tape25`.

VI. PURR

As pointed out by Levitt [33], the natural approach for treating unresolved-resonance self-shielding for Monte Carlo codes is the "Probability Table" method. The trick is to develop tables of the probability that the total cross section will be less than some value σ_t for a number of incident energies. Then, when the Monte Carlo code needs the cross section at E, it selects a random number between 0 and 1 and looks up the corresponding σ_t in the appropriate probability table. The corresponding capture and fission cross sections are obtained from conditional probability tables that give σ_γ and σ_f versus σ_t. This approach allows geometry and mix effects on self-shielding to arise naturally during the Monte Carlo calculation, and it supplies reasonable variances for the tallies. The probability table method has been used successfully in a number of applications, notably the VIM continuous-energy Monte Carlo code [34], which was developed to solve fast-reactor problems where unresolved effects become very important.

The PURR module produces probability tables that can be used in versions of MCNP from 4B on to treat unresolved-resonance self-shielding.

A. Sampling from Ladders

In the unresolved range, we don't know the real center energy of any of the resonances, and we don't know the partial widths that determine the shape and strength of any particular resonance. However, the ENDF evaluation provides us with mean values for the resonance spacings, the probability distribution for the spacings (the Wigner distribution), the mean values for the resonance partial widths, and the distributions for the partial widths (chi-square distributions for various numbers of degrees of freedom). These quantities are given for several different spin sequences, which are statistically independent, and for a number of energies spaced through the unresolved energy range. A "narrow-resonance assumption" is always made in the unresolved range; that is, the energy loss in scattering in the system is assumed to be large with respect to the width of any of the resonances. Thus, neutrons arrive at random energies that are not correlated with the resonance structure. The effective cross sections at one of the energies in the ENDF unresolved grid then depend on a number of resonances in the vicinity of that energy, all of which are assumed to have resonance parameters characteristic of that grid energy value.

This allows us to define a plausible set of cross sections in the vicinity of one of the grid energies. We first define an energy range that will hold a specified number of resonances, and we randomly choose a set of sampling energies in this range, avoiding the ends of the range to reduce truncation effects. For each spin sequence, we start by choosing a center for the starting resonance from a uniform distribution (this provides a random offset between the various spin sequences in the ladder). We then choose a set of partial widths for this resonance drawn from the appropriate distributions. A second resonance is then chosen above the first using the distribution for resonance spacings, and partial widths are randomly chosen for it. Then a third resonance is chosen, and so on, until the energy range defined for the ladder has been filled. We can now compute the cross section contributions from this spin sequence at each of the sampling energies. We then step to the next spin sequence and repeat the process. After looping through all the spin sequences, the accumulated cross sections are one possible set of plausible cross sections that obey the defined statistics for the unresolved range.

We can now go through this set of sampled cross sections, determine how many values of the total cross section hit each bin, and compute the conditional average for the elastic, fission, and capture cross sections for samples where the total ends up in each bin. This is the contribution to the probability table from the particular resonance ladder. However, using only one ladder could result in average cross sections that differ dramatically from the expected infinitely dilute averages computed directly from the resonance parameters. Therefore, the whole sampling process is repeated again for a user-selected number of different ladders. When enough ladders have been processed, the average cross sections will begin to converge to the expected results. Fig. 30 shows piece of a PURR output listing with 16 ladders being processed for U-235 from ENDF/B-VII. Some steps have been eliminated for brevity. The "aver" values computed from the sampling have converged to be fairly close to the infinitely dilute values "infd" computed from the resonance parameters. Here `spot` is the potential scattering cross section, `dbar` is the average resonance spacing, and `sigx` is the competitive cross section.

The probability table can be used to generate a picture of the probability distribution for the total cross section as shown in Fig. 31. This example is for U-238. It demonstrates how the fluctuations get smaller as energy increases, which means that the self-shielding effect also gets smaller. The statements that generate the VIEWR input for this kind of plot are normally commented out.

During the sampling process, PURR also computes Bondarenko-style self-shielded cross sections. See GROUPR (Sect. IX) for a detailed explanation of these. It is easy to accumulate a flux estimate

$$\text{flux} = \sum \frac{1}{\sigma_0 + \sigma_t} \,, \tag{158}$$

and a set of reaction rate estimates

$$\text{rate} = \sum \frac{\sigma_x}{\sigma_0 + \sigma_t} \tag{159}$$

where x can indicate the total, elastic, capture, or fission reactions. Dividing the rates by the flux gives the desired self-shielded cross sections. In modern versions of NJOY,

```
e= 2.2500E+03   spot= 1.1700E+01   dbar= 1.6137E-01   sigx= 0.0000E+00
              total     elastic     fission     capture
     1  1.9471E+01  1.2074E+01  5.3460E+00  2.0507E+00
     2  1.9617E+01  1.2041E+01  5.6196E+00  1.9566E+00
     3  1.9559E+01  1.2075E+01  5.4597E+00  2.0245E+00
     4  1.9458E+01  1.2000E+01  5.5723E+00  1.8859E+00
   ...
    14  1.9456E+01  1.2102E+01  5.3632E+00  1.9905E+00
    15  1.9660E+01  1.2082E+01  5.4429E+00  2.1354E+00
    16  1.9363E+01  1.2019E+01  5.3736E+00  1.9710E+00
  bkgd  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
  infd  1.9778E+01  1.2105E+01  5.6364E+00  2.0363E+00
  aver  1.9567E+01  1.2068E+01  5.5030E+00  1.9964E+00
  pcsd        1.38        0.32        3.28        4.97
  nres        3664
```

FIG. 30: Example of a portion of a PURR output.

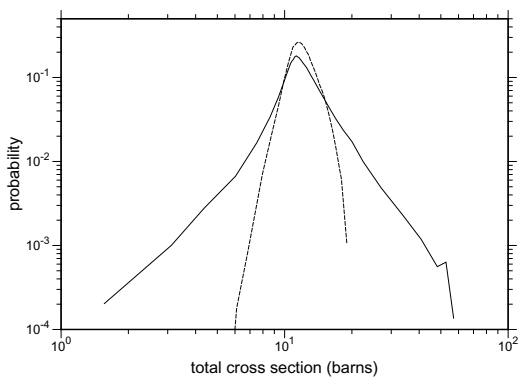

FIG. 31: The probability distribution for the total cross section at 20 keV (solid) and 140 keV (dashed) in the unresolved resonance range of U-238.

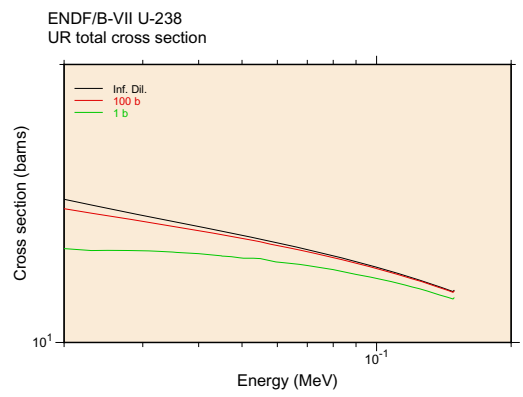

FIG. 32: Bondarenko-style self-shielded cross sections for the total cross section of U-238 in the unresolved resonance region at three different dilutions.

the PURR results for Bondarenko cross sections are used rather than the UNRESR results used in previous times. The are written onto the PENDF file using MF2/MT152.

The Bondarenko cross sections can also be computed directly from the probability table using

$$\sigma_x(E) = \frac{\displaystyle\sum_i \frac{P_i(E)\sigma_{xi}(E)}{\sigma_0 + \sigma_{ti}(E)}}{\displaystyle\sum_i \frac{P_i(E)}{\sigma_0 + \sigma_{ti}}}\,, \tag{160}$$

where x is t, n, f, or γ. PURR prints out these values. They can be compared to the directly computed values to help judge the accuracy of the probability table.

Fig. 32 shows Bondarenko-style self-shielded curves from PURR for the total cross section for U-238 in the unresolved region at three different values of the dilution. This kind of plot is included in the standard graphs available from *http://t2.lanl.gov/data/neutron7.html* Most of the self shielding comes from the elastic channel in this case. Fig. 33 shows how the total self-shielding factor for U-238 varies with temperature and dilution at an energy of 20 keV.

PURR uses the Single-Level Breit-Wigner (SLBW) approximation to calculate cross sections (as specified for ENDF unresolved data), and it uses the ψ-χ method to compute the Doppler broadened values. As is well known, this method doesn't always produce reasonable results for the small cross sections between resonances due to the neglect of interference and multi-channel effects. It is even possible to get negative elastic cross sections. When comparing Bondarenko results from PURR with those from UNRESR, several factors should be considered. The PURR results may be more reliable at low σ_0 values than UNRESR results because of the more complete treatment of resonance overlap effects, but the unrealistic cross sections in the dips between resonances will eventually make even the PURR results suspect at low values. This effect may be especially apparent for the current-weighted total cross section, which is especially sensitive to the low cross sections between resonances.

The piece of a PURR output listing in Fig. 34 shows the probability table for the U-235 example at an energy of 2.25 keV. The columns give the results for the 20 probability bins, but the rightmost 5 columns of numbers have been removed to make the lines fit this page.

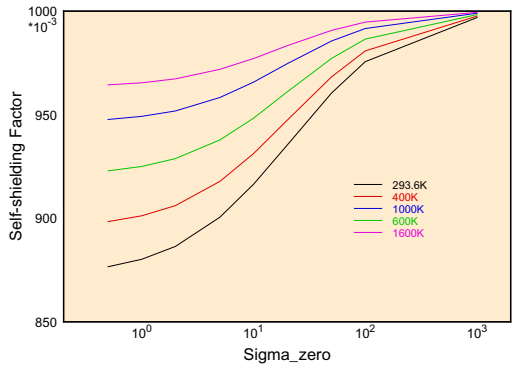

FIG. 33: Self-shielding factors for the total cross section of U-238 at 20 keV in the unresolved resonance range showing the variation with temperature and background cross section (dilution).

Thus, we are seeing bins 1–5 and 11–15. The rows for `tmax` give the upper bounds for the total cross section bins, and `prob` gives the probability that the total cross section lies in the bin. The `tot`, `els`, `fis`, and `cap` lines give the average cross section seen when the total lies in that bin, all for a temperature of 293.6K.

The ENDF format contains an option called LSSF. When LSSF=1, the resonance parameters are to be used to compute a fluctuation factor or self-shielding factor that is to be applied to the cross section given in File 3 of the ENDF tape. When LSSF=0, the parameters are used to compute cross sections that are to be added to any possible background corrections that may be given in File 3. The presence of this option doesn't effect the table printed out by PURR, but when LSSF=1, all the cross section values are divided by the corresponding infinitely dilute cross sections when the output file is written. The bins then contain dimensionless fluctuation factors instead of cross sections in barns.

A caution to the PURR user: the values printed out on the output listing have not been renormalized to match either the `infd` values (LSSF=0) or the File 3 unresolved-range cross sections (LSSF=1).

B. Temperature Correlations

One important feature of PURR is its ability to handle temperature correlations. If the Monte Carlo code is tracking a particle through a system, it periodically checks for a total cross section to calculate the range to the next collision. Consider a collision that takes place in a region at a particular temperature. The Monte Carlo code samples from the probability table, getting a low cross section. The mean free path then takes the particle into another region at a different temperature that contains the same material. The sampled total cross section there cannot be independent from the first; the resulting cross section must also be low. PURR handles this kind of correlation. When a particular ladder of resonances is sampled to obtain a probability table, all the different temperatures are sampled simultaneously at the same energies to preserve the correlations. In the Monte Carlo code, it is only necessary to use the same random number to sample for the cross section in the two different regions to preserve the proper correlations.

C. Self-Shielded Heating Values

PURR can also provide self-shielding effects for the heating if partial heating cross sections are provided to it from HEATR. Of course, there are no resonance parameters for heating, but the heating value does depend on the elastic, fission, and capture cross sections in the unresolved range. It may also have a contribution from competitive reactions, such as MT=51 discrete inelastic scattering or (n,p) absorption. The general idea is to extract the portion of the heating corresponding to elastic, fission, and heating, and to apply the fluctuations in the probability table to them in order to get an estimate for the possible fluctuations in the total heating. This requires that HEATR be run to generate the partial heating values MT=302 (elastic), MT=318 (fission), and MT=402 (capture), in addition to the normal MT=301 (total heating). With these present, each partial heating value (eV-barns) can be divided by the corresponding infinitely-dilute reaction cross section to get eV/reaction for that reaction channel. These numbers can then be multiplied by the corresponding conditional cross section in each bin of the probability table and added to the eV-barns from the competitive reaction, if any, to get a value for heating in eV-barns in each bin. Finally, these values can be divided by the average total for the bin to get a heating value in eV/reaction that MCNP can use with sampled values of the total cross section to produce contributions to heating tallies.

D. Running PURR

The detailed user input specifications for PURR can be found in the source code, the code manuals, or on the web. The following is an example of HEATR and PURR input for a full calculation of the probability tables for a fissionable material. This would be just a small part of a sequence for producing a PENDF file for U-235.

```
probability table
tmin            1.162E+01
tmax            1.352E+01  1.428E+01  1.509E+01  1.594E+01  1.683E+01 ...
                2.339E+01  2.471E+01  2.611E+01  2.758E+01  2.913E+01 ...
prob            2.084E-02  3.597E-02  5.978E-02  8.244E-02  9.788E-02 ...
                6.094E-02  5.269E-02  4.169E-02  3.112E-02  1.806E-02 ...
tot 2.936E+02   1.308E+01  1.393E+01  1.469E+01  1.552E+01  1.639E+01 ...
                2.275E+01  2.403E+01  2.537E+01  2.680E+01  2.832E+01 ...
els 2.936E+02   1.114E+01  1.132E+01  1.144E+01  1.151E+01  1.161E+01 ...
                1.220E+01  1.230E+01  1.262E+01  1.274E+01  1.287E+01 ...
fis 2.936E+02   1.486E+00  1.976E+00  2.444E+00  3.002E+00  3.582E+00 ...
                7.838E+00  8.607E+00  9.206E+00  1.023E+01  1.104E+01 ...
cap 2.936E+02   4.574E-01  6.334E-01  8.095E-01  1.005E+00  1.202E+00 ...
                2.711E+00  3.121E+00  3.543E+00  3.832E+00  4.417E+00 ...
```

FIG. 34: Example of probability table output.

```
 1. ...
 2. heatr
 3. 20 23 24/
 4. 9228 5/
 5. 302 318 402 443 444/
 6. purr
 7. 20 24 25
 8. 9228 8 7 20 32/
 9. 293.6 400 600 800 1000 1200 1600 2000/
10. 1e10 1e4 1e3 300 100 30 10/
11. 0/
12. ...
```

The HEATR run requests partial heating for elastic, fission, capture, kinematic total, and damage on Card 5. The total heating, MT=301, is always produced automatically. On Card 8, the PURR run requests 20 bins for the probability table, and 32 ladders are to be used. This run produces data for 8 temperatures and 7 σ_0 values. To get a more detailed listing of the PURR results, add the parameter `iprint=1` at the end of Card 8.

VII. GASPR

This module will add gas production reactions (MT=203–207) to the PENDF file. Any existing gas-production sections on the input PENDF final are removed, and the file directory is updated to show the new reactions. This module can be run anywhere in the PENDF preparation sequence, as long as it is somewhere after BROADR.

A. Gas Production

The light products of nuclear reactions—namely, protons, deuterons, tritons, He-3s, and alphas—can accumulate as gases in a nuclear system. The resulting hydrogen, deuterium, tritium, He-3, and He-4 can have important effects, such as causing embrittlement. In other cases, the gas may even be the desired product, such as in tritium production.

Keeping track of the total production of each gas species is complicated. Gases can sometimes be determined from the MT number for the reaction; for example, an (n,α) reaction produces one ^{4}He per reaction. The gas might be a residual nucleus as in ^{2}H(n,2n)^{1}H or ^{9}Be(n,2n)2α. Sometimes, the ENDF LR flags are used to indicate that the residual nucleus from a reaction breaks up by further particle emission. An example of this is ^{16}O(n,n$_6$)α represented using MT=56/LR=22. In other cases, the yield per reaction for the light product may be tabulated directly (and even be fractional) when File 6 is used for the reaction. This is common for the high-energy data (150 MeV) introduced for ENDF/B-VI Release 6. GASPR goes through all the reactions in the evaluation and adds up all these various contributions to get the net production of each of the light species.

The ENDF format assigns the MT values from 203 to 207 to represent the production of hydrogen, deuterium, tritium, He-3, and He-4, but only a few evaluators have supplied these data in the past. By using GASPR to generate these data at the PENDF stage, a number of possible inconsistencies are avoided. Therefore, when gas-production MT values are found, GASPR removes them in favor of the ones that it calculates.

When the code runs, it prints out a summary of which reactions contribute to the production of each gas. An example for aluminum from ENDF/B-VII is shown in Fig. 35. Simple multiplicities are shown for most of the reactions, and the "***" symbol indicates that energy-dependent particle yields were found in File 6 for the MT=5 reaction.

B. Running GASPR

The input for running GASPR is minimal—only the units to link the modules are required. Since the material number is not part of the GASPR input, the user's

```
mf6,mt5 found

the gas production threshold is 1.8969E+06 ev

found  1009 points

pendf mt  mt203  mt204  mt205  mt206  mt207
--------  -----  -----  -----  -----  -----
       5    ***    ***    ***    0.0    ***
      22    0.0    0.0    0.0    0.0    1.0
      28    1.0    0.0    0.0    0.0    0.0
      32    0.0    1.0    0.0    0.0    0.0
      33    0.0    0.0    1.0    0.0    0.0
      45    1.0    0.0    0.0    0.0    1.0
     103    1.0    0.0    0.0    0.0    0.0
     104    0.0    1.0    0.0    0.0    0.0
     105    0.0    0.0    1.0    0.0    0.0
     107    0.0    0.0    0.0    0.0    1.0
     108    0.0    0.0    0.0    0.0    2.0
     111    2.0    0.0    0.0    0.0    0.0
     112    1.0    0.0    0.0    0.0    1.0
     117    0.0    1.0    0.0    0.0    1.0

*** means that the yield is energy dependent

found 8 temperatures
```

FIG. 35: Example of GASPR listing showing the reactions that contribute to each gas production reaction.

input PENDF file should contain only the one material of interest. Here is an example of the single input card assuming that the output of the RECONR, BROADR, HEATR, THERMR sequence is on `tape24`:

```
1. 24 25
```

The leading line number is not part of the input. The new gas production reactions will be found in File 3 on `tape25`.

VIII. ACER

The ACER module prepares libraries in ACE format (A Compact ENDF) for the MCNP continuous-energy neutron-photon Monte Carlo code. One of the design goals for MCNP has been to use the most detailed representation of the physics of a problem that is practical. Therefore, the ACE format has evolved to include all the details of the ENDF representations for neutron and photon data. However, for the sake of efficiency, the representation of data in ACE is quite different from that in ENDF. The fundamental difference is the use of random access with pointers to the various parts of the data. Other key differences include the use of union energy grids, equal-probability bins, and cumulative probability distributions.

A. ACER and ACE Data Classes

The ACE format provides for several different "classes" of data, the most popular being the "continuous-energy neutron" class. Others include include photoatomic data, thermal data, and photonuclear data. Files for each class of data are distinguished by a code letter at the end of the ZAID identifier for each material. For example, a file with a ZAID identifier of "13027.00c" would contain continuous-energy neutron data. The data classes currently handled by ACER and the class suffixes are given in Table IV.

TABLE IV: ACE data classes and ZAID suffixes.

Suffix	ACE Data Class
c	continuous-energy neutron data
t	thermal $S(\alpha, \beta)$ data
y	dosimetry data
p	photoatomic data (incomplete)
u	photonuclear data
h	continuous-energy proton data
o	continuous-energy deuteron data
r	continuous-energy triton data
s	continuous-energy He-3 data
a	continuous-energy alpha data

B. Continuous-Energy Neutron Data

The next few sections will discuss the details of preparing data for this very important class of ACE data, including consistency checking and plotting.

C. Energy Grids and Cross Sections

MCNP requires that all the cross sections be given on a single union energy grid suitable for linear interpolation, and this is one of the reasons that the RECONR and BROADR modules of NJOY are also organized around union grids and linear interpolation.

The energy grid and cross section data on an NJOY PENDF file are basically consistent with the requirements of MCNP. The formats for storing energy grid and cross section data in an ACE library are completely described in Appendix F of the MCNP manual, but they will also be reviewed briefly here for the reader's convenience. The principal cross sections are given in the ESZ block. First, the NES energy values of the union grid are given, then the NES values of the total cross section. These are followed by the absorption cross section, elastic cross section, and average heating numbers. The cross sections for the other NTR reaction types are controlled by a set of blocks called MTR, LQR, TYR, and LSIG that contain the reaction ENDF MT numbers, the

Q values, the reaction types, and pointers to the cross section data for each reaction, respectively. The cross section segments addressed by the pointers in the LSIG block contain a count of values, the energy index from the main energy grid for the first value, and the actual cross sections for the reaction.

The energy and cross section values from the input PENDF file are copied onto the grid of the total cross section and stored into the ACE-format blocks. Note that all energy values in the ACE libraries are given in MeV. The ACE heating numbers are computed by dividing the heat production cross sections from MT=301 on the PENDF file by the corresponding total cross sections to obtain heating in MeV per reaction. Damage values from MT=444 are converted to MeV-barns. Sometimes additional cross sections, such as nonelastic or inelastic are needed, and they are added at the end of the reaction list. Note that there are two reaction counters used in the ACE format: NTR is the total number of reactions, and NR is the number of reactions that participate in the transport (*i.e.,* that add up to the total cross section). Reactions with indexes above NR and up to NRT can be used for tallies. This can include reactions like damage or gas production.

D. Two-Body Scattering Distributions

Reactions like elastic and discrete-level inelastic scattering are completely described by their reaction cross sections, Q values, and angular distributions in the center-of-mass (CM) system. The ACE locations for the cross sections and Q values were noted above. The angular distributions are stored in the AND block using a set of pointers stored in the LAND block. Two different representations for angular distributions are provided: equally probable cosine bins, and cumulative distributions. In the older format, which is supported by all versions of MCNP, the angular distributions are represented by 32 equally probable cosine bins for each incident energy (except for isotropic cases). The ENDF angular distributions are obtained from File 4 on the input ENDF tape.

The newer representation for angular distributions has been available in MCNP since version 4C. The ENDF data are converted into cumulative density functions (CDF) and the corresponding probability density functions (PDF) versus scattering cosine. This option is triggered by `newfor=1` in NJOY's ACER input. This representation is superior to the 32-bin one for high-energy evaluations (those that go beyond 20 MeV), which have very sharply forward-peaked shapes. It also reduces biases in the average cosine for scattering at lower energies. Even though it is sometimes more bulky than the 32-bin representation, the newer cumulative format is now the default.

E. Secondary-Energy Distributions

In earlier versions of MCNP, and in the original MCN code, tabulated energy distributions for secondary neutrons from multi-body reactions like $(n, 2n)$ or composite reactions like (n, n'_c) were represented using equally probable bins (see LAW=1 in the DLW block). This representation turned out to be poor because it didn't sample low-probability important events like those in the high-energy tails of energy distributions. The current standard representation for tabulated energy distributions is LAW=4, the "Continuous Tabular Distribution." This scheme is based on sampling from a cumulative density distribution $C(E')$, which gives the probability that the energy of the emitted particle will be less than E'. Since this probability runs from 0 to 1, it is easy to select a random number in this range and interpolate for the corresponding value of E'. The differential density distribution $P(E')$ is also given for use in MCNP's interpolation scheme.

Analytic energy distribution laws, such as the LF=7 simple Maxwellian fission spectrum, the LF=9 evaporation spectrum, or the LF=11 energy-dependent Watt spectrum, are also stored into the DLW and LDLW blocks. The ACE representation is a faithful image of the ENDF representation.

F. Energy-Angle Distributions

A new feature of the ENDF-6 format is coupled energy-angle distributions in File 6. (There was a File 6 format available in earlier versions of the ENDF format, but it was never used. The new ENDF-6 MF=6 format is different.) For neutrons, there are four different representations to be considered:

- The Kalbach law for $\sigma(E \rightarrow E')$ angular distributions as used in the ENDF/B-VI and later evaluations from Los Alamos;
- Legendre coefficients for $\sigma(E \rightarrow E')$ in the laboratory system as used in the ENDF/B-VI and later evaluations from Oak Ridge;
- Secondary-energy distributions versus laboratory scattering cosine as used in the Livermore evaluation of 4-Be-9 in ENDF/B-VI VII; and
- The phase-space distribution as used in the Los Alamos evaluation of the $(n, 2n)$ reaction for 1-H-2 in ENDF/B-VI and VII.

New evaluations using tabulations of angular distributions in the laboratory frame, or coefficients or tabulations in the CM frame, are expected to appear soon.

a. Kalbach Systematics. Kalbach and Mann [35] examined a large number of experimental angular distributions for neutrons and charged particles. They noticed that each distribution could be divided into two parts: an equilibrium part symmetric in μ, and a forward-peaked pre-equilibrium part. The relative amount of the two parts depended on a parameter r, the pre-equilibrium fraction, that varied from zero for low E' to 1.0 for large E'. The shapes of the two parts of the distributions depended most directly on E'. This representation is very useful for pre-equilibrium statistical-model codes like GNASH [36], that can compute the parameter r, and all the rest of the angular information comes from simple universal functions. More specifically, Kalbach's latest work [37] says that

$$f(\mu) = \frac{a}{2\sinh(a)}\Big[\cosh(a\mu) + r\sinh(a\mu)\Big], \qquad (161)$$

where a is a simple function of E, E', and B_b, the separation energy of the emitted particle from the liquid-drop model without pairing and shell terms.

A special sampling scheme has been developed for this case. The MCNP code already had logic to select a secondary energy E' from a distribution. The problem was to select an emission cosine μ for this E'. First, the Kalbach distribution is written in the form

$$f(\mu) = \frac{a}{2\sinh(a)}\Big[(1-r)\cosh(a\mu) + r\mathrm{e}^{a\mu}\Big]\ . \qquad (162)$$

Now select a random number R_1. If $R_1 < r$, use the first distribution in Eq. (162). Select a second random number R_2, where

$$R_2 = \int_{-1}^{\mu} \frac{a\cosh(ax)}{2\sinh(a)}\,dx = \frac{\sinh(a\mu)}{2\sinh(a)} + \frac{1}{2}\ . \qquad (163)$$

Therefore, the emission cosine is

$$\mu = \frac{1}{a}\sinh^{-1}\Big[(2R_2 - 1)\sinh(a)\Big]\ . \qquad (164)$$

If $R_1 \geq r$, use the second distribution in Eq. (162). Select a random number R_2, where

$$R_2 = \int_{-1}^{\mu} \frac{a\mathrm{e}^{ax}}{2\sinh(a)}\,dx = \frac{\mathrm{e}^{a\mu} - \mathrm{e}^{-a}}{\mathrm{e}^{a} - \mathrm{e}^{-a}}\ , \qquad (165)$$

and emit a particle with cosine

$$\mu = \frac{1}{a}\ln\Big[R_2\,\mathrm{e}^{a} + (1-R_2)\,\mathrm{e}^{-a}\Big]\ . \qquad (166)$$

The ACE format for the Kalbach File 6 data is similar to the LAW=4 format used for other continuous energy distributions, namely, cumulative distribution functions. To this are added tables for the pre-equilibrium ratio r and the Kalbach slope parameter a. The result is the LAW=44 format.

b. Legendre or Tabulated Distributions for E to E'. This option is used in many of the newer Oak Ridge evaluations, such as the isotopes of chromium, ^{55}Mn, the isotopes of iron, the isotopes of nickel, the isotopes of copper, and the isotopes of lead. The distribution for outgoing neutrons is given as a set of normalized emission spectra $g(E, E')$ for various incident energies E. In addition, an angular distribution is given for each $E \rightarrow E'$ as a Legendre expansion. Emission energy and angle are given in the laboratory frame. Some recent European evaluations use a similar representation in the CM frame.

The last few versions of ACER tried various ways to handle these formats within the limitations of versions of MCNP up to 4B, but none of them were very satisfactory. Therefore, we added a new representation for MCNP4C called LAW=61. This law uses the cumulative density approach for sampling for E', just as in LAW=4 or 44. In addition, it gives a cumulative type distribution in the emission cosine for each E'. This is a bulky representation, but it has the advantage of not forcing any approximations on MCNP. This format is also selected by giving `newfor=1`, which is now the default for ACER.

For users who prefer to use the older versions of MCNP with the older representation, the option `newfor=0` can be selected. ACER will try to convert the Legendre data into an equivalent section using ENDF MF=7 format with 33 cosines. This section can then be processed into ACE LAW=67 format as described below. This process is reasonably straightforward for laboratory data. If necessary, ACER does attempt to convert CM data to the lab frame when building one of these MF=7 representations, but the methods used are fairly rough and approximate.

c. Laboratory Angle-Energy Distributions. The ENDF/B-VI evaluation for ^{9}Be prepared at the Lawrence Livermore National Laboratory (LLNL) uses the angle-energy option. That is, the outer loop is on incident energy E, the next loop is on laboratory scattering cosine μ, and the inner loop is on secondary energy E'. In order to sample from data in this form, the first step is to integrate over E' for each μ in order to obtain the differential angular distribution $f(E, \mu)$. This angular distribution is converted into 32 equally probable bins and stored into the ACE file using the same format used for two-body angular distributions like elastic scattering. The emission spectra for the individual μ values are normalized and stored into the file using a format called LAW=67 (named for ENDF File 6, Law 7). MCNP can sample from this representation as follows: for each emission, first sample from $f(\mu)$ to get an emission angle, then find the corresponding spectrum and sample from its cumulative probability distribution to get the value of E'.

d. N-Body Phase-Space Distributions. The phase-space distribution for particle i in the CM system is given by

$$P_i^{\mathrm{CM}}(\mu, E, E') = C_n\sqrt{E'}\,(E_i^{\max} - E')^{3n/2-4}\,, \qquad (167)$$

where $E_i^{\max}$ is the maximum possible CM energy for particle i, μ and E' are in the CM system, and the C_n are normalization constants. The value of $E_i^{\max}$ is a fraction of the energy available in the CM:

$$E_i^{\max} = \frac{M - m_i}{M} E_a \,, \tag{168}$$

where M is the total mass of the n particles being treated by this law, and

$$E_a = \frac{m_T}{m_p + m_T} E + Q \,. \tag{169}$$

Here, m_T is the target mass, and m_p is the projectile mass. In summary, the data items required for the phase-space law are:

Symbol	ENDF	Location
n	NPSX	N2 field of the LAW=6 CONT
m_i	AWI	C1 field of third card in MF=1
m_p	AWP	C2 field of LAW=6 TAB1 record
m_T	AWR	C2 field of section HEAD record
M	APSX	C1 field of LAW=6 CONT record
Q	Q	C1 field of the MF=3 TAB1 record

These equations are sampled with a compact numerical scheme similar to LAW=4. Note that all the spectra scale with the maximum possible outgoing energy. Therefore, it is easy to construct a single normalized distribution with $E_i^{\max}$=1 with a reasonable number of $x = E'/E_i^{\max}$ points and then to construct a cumulative distribution function for it. The grid uses uniform spacing above $x = 0.10$ and log spacing below. The x grid, the probability density values $P(x)$, the cumulative densities $C(x)$, NPXS, and APSX are stored in the Law=66 format. For any given E, the cumulative distribution function is sampled with a random number between 0 and 1. The resulting x value is then multiplied by $E_i^{\max}$ to get the emitted E' value. The corresponding CM cosine value is obtained by sampling uniformly in the interval $[-1, 1]$.

The CM to lab transformation is carried out by adding the CM velocity of the initial collision to the emitted particle velocity. The appropriate equations are

$$E'_{\rm LAB} = E_{\rm CM} + E'_{\rm CM} + 2\mu_{\rm CM}\sqrt{E_{\rm CM} E'_{\rm CM}} \,, \tag{170}$$

and

$$\mu_{\rm LAB} = \frac{\sqrt{E'_{\rm CM}}\,\mu_{\rm CM} + \sqrt{E_{\rm CM}}}{\sqrt{E'_{\rm LAB}}} \,, \tag{171}$$

where the CM energy is

$$E_{\rm CM} = \frac{A}{A+1} E \,. \tag{172}$$

G. Photon Production

Earlier versions of MCNP used a very simple representation for photon production from neutron reactions. There was a single total photon production cross section on the same union grid as the neutron data, and there were 600 words of data describing the spectrum of outgoing neutrons. This table contained 20 equally likely outgoing photon energies for each of 30 incident neutron groups. This representation did not achieve the MCNP goal of providing the best possible representation of the physics of the problem. It was inadequate in representing discrete photons because their real energies were often lost, and it was inadequate in representing low-probability events from the tails of distributions. This was especially noticeable in capture events because of the high photon energies possible. It is still possible to use this representation, but it is no longer recommended. The newer "Expanded Photon Production Data" option is preferred.

a. Photon Production Cross Section. In the earlier versions of the ENDF format, photon production cross section information was given in File 13 (photon production cross sections), or as a combination of File 3 (reaction cross sections) and File 12 (photon production yields). With the ENDF-6 format, photon production can also be computed using a combination of File 3 and File 6 (product yields and energy-angle distributions).

The first step in photon production processing is to examine MF=12 on the ENDF file for transition probability arrays (LO=2). If they are found, they are converted into the photon yield format (LO=1). The final photon yield data are written onto a scratch file. Next, the MF=13 data are copied, and MF=14 (photon angular distributions) is updated to reflect the changes made in MF=12. Finally, if File 6 is present, any photon production subsections found are converted into a special MF=16 format on the scratch file. In the next step, the sum of MF=13, MF=12×MF=3, and MF=16×MF=3 is computed for all the photon reactions on the normal union energy grid. Later, this total photon production cross section is written into the ACE GPD block.

b. Expanded Photon Production Data. This preferred representation allows each discrete photon to be treated with its proper energy, and it allows for a much better representation of the spectrum of continuum photons. In the ACE representation, the MTRP block lists all the photon reactions included by ENDF MT number. Since some reactions may describe more than one photon (for example, radiative capture reactions usually describe many discrete photons), the identifier numbers are given as 1000×MT plus a photon index. Thus 102002 would stand for the second photon described under radiative capture (MT=102). Each of the NMTR photons listed in the MTRP block can have its own cross section or yield as described in the SIGP and LSIGP blocks, its own angular distribution as described in the ANDP and LANDP blocks, and its own energy distribution as described in the DLWP and LDLWP blocks. In addition, the YP block contains a list of reaction MT numbers that are needed as photon production yield multipliers.

Continuous photon spectra from File 15 are given by tabulations of emission probability *vs* photon energy. These are converted directly into tables giving the cumulative density function (cdf) and partial density function (pdf) *vs* photon energy.

H. Probability Tables for the Unresolved Region

Starting with Version 4B, MCNP has been able to make use of cross section probability tables for energies in the unresolved resonance range to get proper self-shielding effects. These tables are produced by the PURR module of NJOY. The tables provide a cumulative density function that gives the probability that the total cross section observed at some energy E in the unresolved resonance range will be less than some particular values. MCNP can then throw a random number and search this table to get a sample value for the total cross section at each collision. The probability tables also include conditional probability distributions that give values for scattering, fission, capture, and heating for each particular value of the total. The probability tables are read from the input PENDF file.

I. Charged-Particle Production

Another recent addition to the continuous-energy neutron data class for MCNP is a detailed representation of the emission of light charged particles from neutron-induced reactions. These kinds of data are now available for a number of materials in ENDF/B evaluations, including the large set of evaluations that go to incident neutron energies of 150 MeV.

When present, the charged-particle production data reside in a set of ACE blocks at the end of the ACE file. There is a set of data given for each charged particle produced: protons, deuterons, tritons, He-3s, and alphas. These data sets give a production cross section and a heating value referenced to the standard union energy grid from the ESZ block, and they also give the fraction of the production coming from each reaction producing the particle, together with the associated angle and energy distribution data for the reaction.

These charged particle production distributions will be used in advanced versions of MCNP to provide the source from neutron reactions for subsequent charged-particle transport, thus providing a true n-particle Monte Carlo capability.

Care must be taken to handle heating correctly in n-particle transport calculations. The heating value in the main ESZ block of the ACE format contains energy deposition resulting from all the charged particles resulting from nuclear reactions. If a user wants to do a coupled neutron-gamma-proton calculation, it is necessary to subtract the proton heating from the main heating value first. The subsequent non-local energy deposition from the transported protons will be handled directly. This is why the new charged-particle blocks include the separate heating contribution associated with each particle.

J. Gas Production

During the NJOY run that makes the input for ACER processing, the user can choose to run the GASPR module. It goes through all the reactions given on its input ENDF and PENDF files and constructs reaction cross sections for the production of the light charged particles (p, d, t, He3, and alpha) and writes them on a new version of the PENDF file. When `acelod` processes this PENDF file, the cross sections are made available in the ACE file for use in MCNP tallies. Watch for reaction names like "(n,Xp)."

These gas production cross sections are basically the same as the charged-particle production cross sections in the new charged-particle sections on the ACE file (except for reactions using the ENDF LR flags), but the latter are not available for simple tallies.

K. Consistency Checks and Plotting

As part of the Quality Assurance (QA) process for producing ACE library files, ACER has the capability to read in an ACE file and check the data for some common problems. These are called "consistency checks," and the checks are provided for class "c" libraries are as follows:

- check reaction thresholds against Q values,
- check the main energy grid is monotonic,
- check angular distributions for correct reference frame,
- check angular distributions for unreasonable cosine values (mu out of range, mu values not monotonic, cumulative probability out of range, cumulative probabilities not monotonic)
- check energy distributions (illegal interpolation, E' greater than the maximum possible value, bad cumulative probability, decreasing cumulative probability, bad Kalbach r, bad angular cumulative probability, decreasing angular commutative probability),
- check photon production sum,
- check photon production distributions (bad cumulative probability, decreasing cumulative probability), and

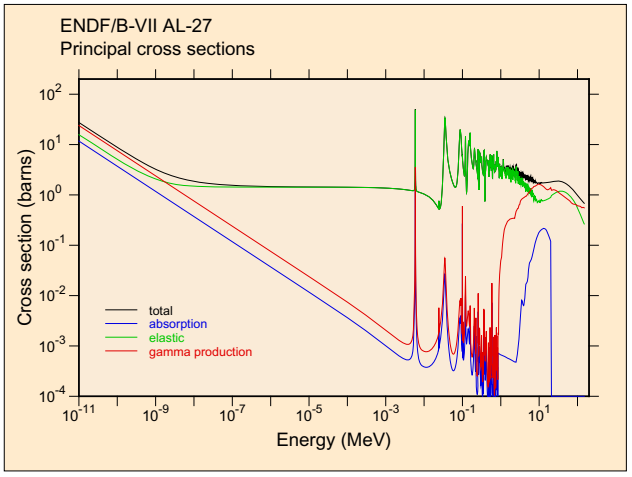

FIG. 36: A log plot of the ACE principle cross sections for ^{27}Al from ENDF/B-VII *vs* neutron energy. Note the extension beyond 20 MeV to 150 MeV.

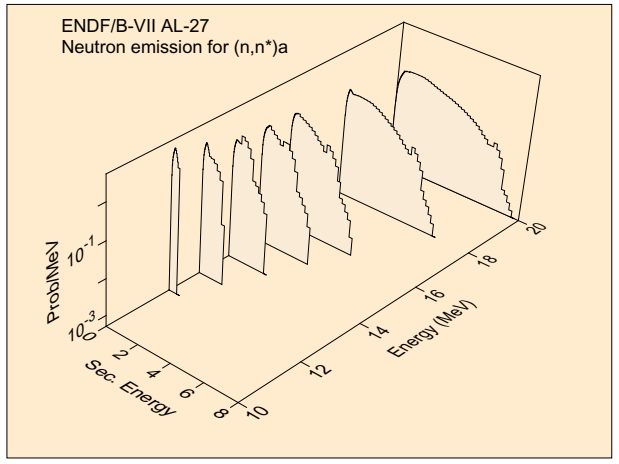

FIG. 37: A 3-D view of the energy distribution for neutrons emitted from the (n,n′α) reaction from ^{27}Al from ENDF/B-VII.

- check particle production sections (bad LAW=4 cumulative probability, decreasing LAW=4 cumulative probability, bad LAW=44 cumulative probability, decreasing LAW=44 cumulative probability, bad LAW=44 Kalbach r, bad LAW=61 cumulative probability, decreasing LAW=61 cumulative probability, bad LAW=61 angular cumulative probability, decreasing angular cumulative probability)

When E' values greater than the expected limit are found, the consistency-check routine can correct them. See the sections on running ACER for the details.

Another important part of the ACE QA procedure is to prepare an extensive set of plots and to scan through them for possible problems. The plots are generated in the form of an input file for the VIEWR module, which can then prepare the final plots as color Postscript files. The plots include pages showing the principle ACE cross sections (total, elastic, absorption, photon production), non threshold reactions (such as capture and heating), and threshold reactions in log form (to feature the low-energy region) and linear form (for higher energies). Several reactions are given per page. The routine also prepares expanded views of the cross sections in the resonance range to make the details of prominent resonances more apparent. In addition, the plots include 3-D perspective views of angular distributions for the new format. The 32-bin representation of the angular distributions is shown as contour plots. The energy and energy-angle distributions for tabulated representations are also shown as 3-D perspective plots. Finally, the particle production data, if present, are shown using similar 2-D and 3-D plots. Figure 36 is an example of the log plot for the principle cross sections, and Figure 37 is an example of a 3-D plot for particle emission. A large set of these plots for ENDF/B-VII evaluations is available on the web [38].

L. Thermal Cross Sections

Thermal data is the second class of ACE data to be considered. For energies below about 4–10 eV, the thermal motions of nuclei can lead to significant energy gains in neutron scattering. In addition, the binding of atoms into liquids and solids begins to affect the scattering cross section and the distribution of scattered neutrons in angle and energy. MCNP can handle thermal neutron scattering from the atoms of a free gas using internal kinematic formulas that assume a Boltzmann distribution. The bound-atom effects are treated using thermal data from ENDF evaluations stored in a special MCNP thermal library.

The ENDF format allows for several thermal processes. Thermal inelastic scattering is represented using the scattering law $S(\alpha,\beta)$, where α and β are dimensionless momentum and energy transfer parameters, respectively:

$$\sigma(E\rightarrow E',\mu) = \frac{\sigma_{\rm b}}{2kT}\sqrt{\frac{E'}{E}}\,{\rm e}^{-\beta/2}S(\alpha,\beta)\ , \qquad (173)$$

where

$$\alpha = \frac{E'+E-2\mu\sqrt{EE'}}{AkT}\ , \qquad (174)$$

$$\beta = \frac{E'-E}{kT}\ , \qquad (175)$$

and where E and E' are the incident and outgoing neutron energies, μ is the scattering cosine, T is the absolute temperature, A is the mass ratio to the neutron of the scatterer, and k is Boltzmann's constant. This process occurs in all the ENDF thermal materials, such as water, heavy water, graphite, beryllium, beryllium oxide, polyethylene, benzine, and zirconium hydride.

The THERMR module of NJOY uses this equation and evaluated $S(\alpha, \beta)$ data from an ENDF-format evaluation to compute $\sigma(E \rightarrow E', \mu)$. The E' dependence of the integral over μ is computed adaptively so as to represent the function using linear interpolation within a specified tolerance. The angular distribution at each of these E' values is then calculated in a similar way, but the curve of σ *vs* μ is then converted into equally probable bins (typically 20), and a discrete angle is selected for each bin that preserves the average scattering cosine for that bin. The data are written onto the PENDF tape using special MF=6-like formats.

The `acesix` subroutine reads this thermal section on the input PENDF file. In older versions of this method, the energy distribution $\sigma(E \rightarrow E')$ is converted into equally probable bins (typically 16), and a discrete energy is chosen for each bin that preserves the average energy in that bin. The result of this process is a set of equally probable events (typically $8 \times 16 = 128$ events) in E', μ space for each incident energy. It is very easy to sample from this representation, and it is fairly compact. See `iwt=0` in the input instructions.

However, it must be recognized that this scheme is only reasonable if each neutron undergoes several scattering events before being detected. The artificial discrete lines must be averaged out. Be careful when using this method to analyze experimental arrangements using optically thin elements and small-angle detectors. In addition, as in all equal-probability bin schemes, the wings of functions (which may be unlikely but important) are not well sampled. ACER includes a variation to partially relieve this problem: instead of equal bin weights, the pattern 1, 4, 10, 10,..., 10, 4, 1 is used (see "variable weighting" in the input instructions). This approach produces some samples fairly far out on the wings of the energy distribution. Angles are still equally weighted. See `iwt=1` in the input instructions.

In practice, this method using discrete energies can still leave some artificial peaks in typical thermal neutron spectra. These peaks don't have much effect on average quantities for most applications, but they are visually offensive. The newest versions of MCNP support continuous distributions of $\sigma(E \rightarrow E')$ with PDF and CDF values to drive the sampling. See `iwt=2` in the input instructions. This is the preferred representation.

The second ENDF process to consider is "coherent elastic" scattering. This process occurs in powdered crystalline materials, such as graphite, beryllium, and beryllium oxide. Bragg scattering from the crystal planes leads to jumps in the cross section *vs* energy curve as scattering from each new set of planes becomes possible. The formula for this process can be written in the following form:

$$\sigma(E, \mu) = \sigma_{\mathrm{c}} \frac{\pi \hbar^2}{4MEV} \sum_{\tau \neq 0}^{\tau < \tau_{\max}} f(\tau) \delta(\mu - \mu_0[\tau]) \ , \quad (176)$$

where

$$\tau_{\max} = \sqrt{\frac{8ME}{\hbar^2}} \ , \quad (177)$$

and

$$\mu_0 = 1 - \frac{\hbar^2 \tau^2}{4ME} \ , \quad (178)$$

and where E is the incident neutron energy, E' is the outgoing neutron energy, μ is the scattering cosine, σ_{c} is the characteristic coherent scattering cross section for the material, M is the target mass, V is the volume of the unit cell, τ is the radius of one of the reciprocal lattice shells, and $f(\tau)$ is the effective structure factor for that shell.

Examination of these equations shows that the angle-integrated cross section will go through a jump proportional to $f(\mu)$ when E gets large enough so that $\mu_0 = -1$ for a given value of τ. At this energy, a backward directed component of discrete-angle scattering will appear. As the energy increases, this discrete-angle line will shift toward the forward direction. It is clear that the only information that MCNP needs to represent this process in complete detail is a histogram $P(E)$ tabulated at the values of E where the cross section jumps. The cross section will then be given by $P(E)/E$. The intensity and angle of each of the discrete lines can be deduced from the sizes of the steps in $P(E)$ and the E values where the steps take place. The $P(E)$ function is computed from $\sigma(E)$ in `acesix`.

The third ENDF thermal process is incoherent elastic scattering. It occurs for hydrogenous solids like polyethylene and zirconium hydride by virtue of the large incoherent scattering length and small coherent scattering length of hydrogen. The equation describing this process is

$$\sigma(E, \mu) = \frac{\sigma_{\mathrm{b}}}{2} \mathrm{e}^{-2WE(1-\mu)/A} \ , \quad (179)$$

where σ_{b} is the characteristic bound cross section, and W is the Debye-Waller integral. The THERMR module of NJOY computes the integrated cross section $\sigma(E)$ for this process and a set of equally probable cosines for each incident energy E; it writes them onto the PENDF file using a special format. These quantities are copied into the ITCE and ITCA blocks of the ACE format.

No consistency checking is currently available for thermal files, but a set of plots is provided to help with QA. The plots are prepared as input for the VIEWR module, which can generate the final color Postscript files for plotting. A log plot of the total, inelastic, and elastic (if present) cross sections is given first, followed by by plots of the average scattering cosine, mubar, and the average energy of the scattering neutrons, ebar. If the continuous energy version for the scattering tables is used (`iwt=2`), 3-D perspective views of the thermal inelastic $\sigma(E \rightarrow E')$ are provided, followed by 3-D perspective views the angular distribution *vs* E' at selected incident

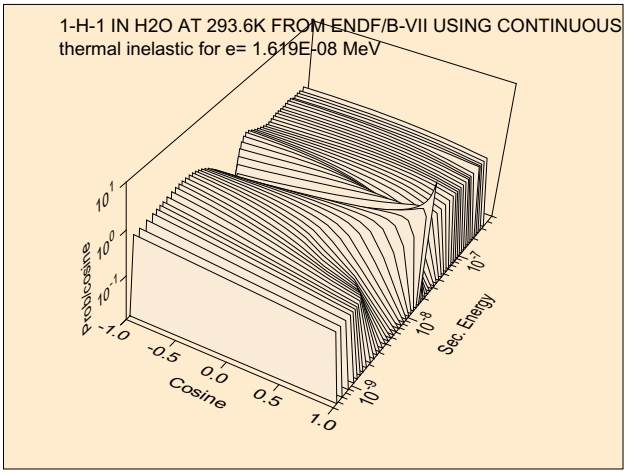

FIG. 38: A perspective view of an angle-energy distribution for H in H_2O.

energies. Fig. 38 shows an example of an angular distribution plot.

A large set of graphs for the thermal scattering materials of ENDF/B-VII was produced during ACER processing [32]. The PDF plots include cross sections, average cosine, average energy, perspective plots of energy distributions, and perspective plots of angle-energy distributions for selected incident energies.

M. Dosimetry Cross Sections

The ACE dosimetry data forms another class of MCNP data. This class of library provides cross sections to be used for response functions in MCNP; the data cannot be used for actual neutron transport. The information on a dosimetry file is limited to an MTR block, which describes the reactions included in the set, an LSIG block containing pointers to the cross section data for the reactions, and the SIGD block, which contains the actual data. The format for dosimetry cross section storage is different from the format for neutron cross sections. The union grid for linear interpolation is not used; instead, the cross sections are stored with their ENDF interpolation laws. If the file mounted as the input PENDF file is a real PENDF file (that is, if it has been through RECONR), all the reactions are already linearized, and the interpolation information stored in the SIGD block will indicate linear interpolation for a single interpolation range. However, if the file mounted as NPEND is actually an ENDF file, the SIGD block may indicate multiple interpolation ranges with nonlinear interpolation laws.

N. Photoatomic Data

Photoatomic data form another ACE class. Photons from direct sources and photons produced by neutron reactions are scattered and absorbed by atomic processes, producing heat at the same time. The existing MCNP "photon interaction" libraries were based on fairly old cross section data and assembled by hand [39, 40]. This version of ACER contains the beginnings of an automated capability to produce these libraries from the latest ENDF/B photoatomic data.

The cross sections for the basic photoatomic process incoherent scattering, coherent scattering, pair production, and photoelectric absorption are given on a union energy grid. Actually, the energies and the cross sections are stored as logarithms, and MCNP uses linear interpolation on them; therefore, the effective interpolation law is log-log. MCNP determines the mean-free-path to a reaction using the sum of these partial cross sections instead of a total cross section.

If the reaction is incoherent (Compton) scattering, the scattering is assumed to be given by the product of the free-electron Klein-Nishina cross section and the incoherent scattering function, $I(v)$. MCNP assumes that this function is tabulated on a given 21-point grid of v values, where v is the momentum of the recoil electron given in inverse angstroms. It is easy to extract the scattering function from the section MF=26,MT=504 of the photo-atomic ENDF library and to interpolate the function onto the required v grid.

If the reaction is coherent (Thomson) scattering, the photon will be scattered without energy loss, and the scattering distribution is given by the Thomson cross section times the coherent form factor. The sampling scheme used in MCNP requires the coherent form factor tabulated on a predefined set of 55 v values, and it also requires integrated form factors tabulated at 55 values of v^2, where the v values are the same set used for the form factors. These values are extracted from the section MF=26,MT=502.

Photoelectric absorption results in the emission of a complex pattern of discrete "fluorescence" photons and electrons (which lead to heating) due to the cascades through the atomic levels as the atom de-excites.

The fluorescence part is not coded yet. This section will be completed when the new fluorescence methods have been developed and installed in MCNP. In the meantime, ambitious users may be able to meld the new cross sections computed by the methods discussed in this section with the fluorescence data from the current MCNP library.

O. Photonuclear Data

Photonuclear data forms a new class of ACE data that is just becoming available. A fairly large number of photonuclear evaluations is now available in ENDF/B-VII. They are largely based on work coordinated by the Data Section of the IAEA.

The new ACE photonuclear format differs in some ways from the more familiar continuous-energy format

(class "c"). In this format, all emissions (neutrons, photons, and charged particles) are treated symmetrically using blocks with the style of the particle production blocks from the class "c" format. Each possible reaction product is described by a production cross section, a partial heating value, fractions represented by different reaction channels, and angle, energy, or energy-angle distributions for each reaction channel.

One class of the new evaluations consists of those performed at Los Alamos and a large set of materials using the same methods generated at the Korea Atomic Energy Research Institute (KAERI). These evaluations lump all the photonuclear processes into a single reaction with MT=5 and use subsections of MF=6/MT=5 to represent all the neutrons, photons, and charged particles produced. These are processed by using the energy grid of MT=5 as the union grid. The ACE total and heating values are constructed on this grid. Then the sections of File 6 are processed to generate production cross sections and partial heating values for each of the emitted species. Each emitted neutron or charged particle also has an energy-angle distribution associated with it, represented using the Kalbach LAW=44 format, and emitted photons are represented using LAW=4.

P. Type 1 and Type 2 ACE Files

ACE library files come in two different types in order to allow for efficiency and portability.

- Type 1 is a simple formatted file suitable for exchanging ACE libraries between different computers.
- Type 2 is a Fortran direct-access binary file for efficient use during actual MCNP runs.

There used to be Type 3 using word-addressable random access methods, but these methods were machine specific, and the type has been abandoned. ACER stores all values in memory as real numbers, and it is easy to write them out in Type-2 format, because all fields in that format are also represented by real numbers (except for some of the fields in the header record). However, the Type 1 format requires that fields that represent integers must be written in integer format, that is, right justified and without decimal points.

The ACER user can prepare libraries using either Type 1 or Type 2 output. The advantage of Type 1 is that the files can be easily moved to other machines or laboratories. The advantage of Type 2 is that it is more compact and can be used directly by MCNP with fewer performance penalties. At any time, ACER can be used to convert one format to another, or to make a listing of the data from any of the formats (see `iopt` values from 7 through 9 in the input instructions).

Q. Running ACER

The input instructions for ACER can be viewed using the comments in the source code, the NJOY manuals, or on the NJOY web pages. The following is a typical ACER input deck for producing a continuous-energy file (class "c"):

```
 1. acer
 2. 20 21 0 31 32
 3.  1 0 1 .10/
 4.  'ENDF/B-VII U-238'/
 5.  9237 293.6/
 6.  /
 7.  /
 8.  acer
 9.  0 31 33 34 35
10.  7 1 1/
11.  'ENDF/B-VII U-238'/
12.  viewr
13.  33 36/
14.  stop
```

This input includes two passes through ACER. The first reads ENDF data on unit 20 and PENDF data on unit 21. It then prepares the ACE output file on unit 31 and the XSDIR stub record on unit 32. The second reads in the ACE file from the first, runs it through consistency checking, makes a listing file, makes a set of plots (on unit 33), and produces a new ACE file (unit 34) and XSDIR stub (unit 35). Card 5 gives the MAT number for U-238 and the temperature to be extracted from the PENDF file. The XSDIR stub record is used when building a big multi-material library for MCNP.

In MCNP jobs, materials are identified by their "zaid" numbers (rhymes with "staid"); they are constructed by using the value 1000×Z+A, appending the value of `suff` (the suffix), and then adding a letter that indicates the library class ("c" for continuous, "t" for thermal, *etc.* (see Table IV). For example, 92235.70c denotes an ENDF/B-VII continuous (fast data) library entry for ^{235}U.

The parameters on Card 3 say to use the fast format, minimize printing, prepare Type 1 output, and use ".10" for the ZAID identifier. Card 4 gives a comment string for the contents. The defaulted Card 6 requests the "new format" and detailed photon production data. This is now standard for ACER. The defaulted Card 7 controls obsolete thinning options. On Card 10, the type parameter is set for print or edit and the output listing is turned on. The final step is to run VIEWR to make plots of the ACER results in Postscript format.

The following example shows how to prepare thermal data for H in H_2O at 600K:

TABLE V: Conventional values for the thermal MT numbers (`MTI` and `MTE`) used in ACER and THERMR.

Thermal Material	`MTI`	`MTE`
H in H_2O	222	
D in D_2O	228	
Be metal	231	232
Graphite	229	230
Benzine	227	
Zr in ZrH	235	236
H in ZrH	225	226
Be(BeO)	233	234
O(BeO)	237	238
H in Polyethylene	223	224
U(UO2)	241	242
O(UO2)	239	240
Al	243	244
Fe	245	246

```
 1. acer
 2. 20 21 0 31 32
 3. 2 0 1 .25/
 4. '1-h-1 in h2o at 600k from endf-vii'/
 5. 125 600 'hh2o'/
 6. 1001/
 7. 222 1 0 0 1 1000. 2/
 8. acer
 9. 0 31 22 33 34/
10. 7 1/
11. '1-h-1 in h2o at 600k from endf-vii'/
12. viewr
13. 22 23/
14. stop
```

The first entry on Card 3 is 2 to indicate thermal data. The next two entries say do not print and use Type 1 format. The new ZAID identifier will be ".25." On Card 5, the ENDF MAT number is 125 for H-1, the temperature is 600K, and the material name of "hh2o" will be used to define the ZA part of the ZAID; that is, the ZAID for this case will be "hh2o.25t." Card 6 gives the ZA number for the "fast" ZAID to be associated with this thermal ZAID in MCNP. Card 7 gives the thermal MT number that references the desired data on the PENDF file (see Table V). The next parameter is `nbint`, and for modern continuous thermal distributions, it should be set to 1. The two zeros are for elastic data. The "1" is used because this is not a mixed moderator, the upper limit of "1000." means use whatever upper limit is found on the input PENDF file, and the "2" refers to the number of primary scattering atoms in the molecule (2 for H_2O).

IX. GROUPR

GROUPR produces self-shielded multigroup cross sections, anisotropic group-to-group scattering matrices, and anisotropic photon production matrices for neutrons from ENDF evaluated nuclear data. With ENDF-6 format files, photonuclear data and incoming and outgoing charged particles can also be handled. Special features are provided for ratio quantities (for example, $\overline{\mu}$, $\overline{\nu}$, or photon yield), inverse velocity, delayed neutron spectra by time group, and anisotropic thermal neutron scattering. Fission is represented as a group-to-group matrix for full generality. Scattering matrices and photon production matrices may be self-shielded if desired.

The Bondarenko narrow-resonance weighting scheme [41] is normally used. Optionally, a weighting flux can be computed for various mixtures of heavy absorbers with light moderators. An accurate pointwise solution of the integral slowing down equation is used. This option is normally called on to account for intermediate resonance effects in the epithermal range.

Neutron data and photon-production data are processed in a parallel manner using the same weight function and quadrature scheme. This helps to assure consistent cross sections for coupled neutron-photon problems. Two-body scattering is computed with a center-of-mass (CM) Gaussian quadrature, which gives accurate results even for small Legendre components of the group-to-group matrix.

User conveniences include free-form input and complete control over which reactions are processed. The neutron group structure, photon group structure, and weight function can each be read in or set to one of the internal options. Output can be printed and/or written to an output "groupwise-ENDF" (GENDF) file for further processing by a formatting module (DTFR, CCCCR, MATXSR, WIMSR), by the covariance module (ERRORR), or by the MCNP continuous-energy Monte Carlo module (ACER).

A. Multigroup Constants

Multigroup constants are normally used by computer codes that calculate the distributions of neutrons and/or photons in space and energy, and that compute various responses to these distributions, such as criticality, dose to personnel, or activation of materials. These distributions are solutions of the neutral particle transport equation. The following development uses a notation based on Bell and Glasstone [42], where the lower-case sigma is used for both macroscopic and microscopic cross sections, depending on the context. One-dimensional slab geometry is used throughout for simplicity.

$$\mu \frac{\partial}{\partial x}\phi(x,\mu,E) + \sigma_t(x,E)\,\phi(x,\mu,E) = \int d\mathbf{\Omega}' \int dE'\,\sigma_X(x,E'\!\to\!E,\mathbf{\Omega}'\!\to\!\mathbf{\Omega})\,\phi(x,\mu',E') + Q(x,\mu,E)\ , \tag{180}$$

where the flux ϕ is allowed to vary with position x, direction $\mathbf{\Omega}$ with polar cosine μ, and energy E. Similarly, the macroscopic total cross section σ_t varies with position

and energy. The right-hand side of the equation contains the source due to transfers from other directions $\mathbf{\Omega}'$ and energies E' (as described by the macroscopic transfer cross section σ_X), and a fixed or external source Q.

The macroscopic cross sections (in units of cm^{-1}) in Eq. (180) can be calculated from microscopic cross sections for the component isotopes or elements (in barns) using

$$\sigma_t(x,E) = \sum_i \rho_i(x)\,\sigma_t^i(T[x],E)\ , \tag{181}$$

where ρ_i is the number density for a constituent (in $\mathrm{barns}^{-1}\mathrm{cm}^{-1}$), which may vary with position, and T is the temperature, which may also vary with position. A similar formula holds for σ_X.

The transfer cross section σ_X (which includes both scattering and fission processes) is normally assumed to depend only on the cosine of the scattering angle, $\mu_0 = \mathbf{\Omega}\cdot\mathbf{\Omega}'$. This allows σ_X to be expanded using Legendre polynomials

$$\sigma_X(x,E'\to E,\mathbf{\Omega}'\to\mathbf{\Omega}) = \sum_{\ell=0}^{\infty}\frac{2\ell+1}{4\pi}\,\sigma_{X\ell}(x,E'\to E)\,P_\ell(\mu_0)\ . \tag{182}$$

Application of the addition theorem and integration over azimuthal angle then gives

$$\mu\frac{\partial}{\partial x}\phi(x,\mu,E) + \sigma_t(x,E)\,\phi(x,\mu,E) = \sum_{\ell=0}^{\infty}\frac{2\ell+1}{2}P_\ell(\mu)\int\sigma_{X\ell}(x,E'\to E)\,\phi_\ell(x,E')\,dE' + Q(x,\mu,E)\ , \tag{183}$$

where

$$\phi_\ell(x,E) = \int P_\ell(\mu)\,\phi(x,\mu,E)\,d\mu\ . \tag{184}$$

The desired responses are then given by

$$R(x) = \int\sigma_r(x,E)\,\phi_0(x,E)\,dE\ , \tag{185}$$

where σ_r is the reaction cross section for the response. The next step is to integrate Eqs. (183) and (185) over a range of energies chosen to lie in group g. The results are

$$\mu\frac{\partial}{\partial x}\phi_g(x,\mu) + \sum_{\ell=0}^{\infty}P_\ell(\mu)\,\sigma_{t\ell g}(x)\,\phi_{\ell g}(x) = \sum_{\ell=0}^{\infty}\frac{2\ell+1}{2}P_\ell(\mu)\sum_{g'}\sigma_{X\ell g'\to g}(x)\,\phi_{\ell g'}(x) + Q_g(x,\mu)\ , \tag{186}$$

and

$$R(x) = \sum_g \sigma_{rg}(x)\,\phi_{0g}(x)\ , \tag{187}$$

where

$$\phi_{\ell g}(x) = \int_g \phi_\ell(x,E)\,dE\ , \tag{188}$$

$$\sigma_{t\ell g}(x) = \frac{\int_g \sigma_t(x,E)\,\phi_\ell(x,E)\,dE}{\int_g \phi_\ell(x,E)\,dE}\ , \tag{189}$$

$$\sigma_r(x) = \frac{\int_g \sigma_r(x,E)\,\phi_0(x,E)\,dE}{\int_g \phi_0(x,E)\,dE}\ , \tag{190}$$

and

$$\sigma_{X\ell g'\to g} = \frac{\int_g dE\int_{g'} dE'\,\sigma_{X\ell}(x,E'\to E)\,\phi_\ell(x,E)}{\int_g \phi_\ell(x,E)\,dE}\ . \tag{191}$$

The last three equations provide the fundamental definitions for the multigroup cross sections and the group-to-group matrix. Note that the values of the group constants depend upon the basic energy-dependent cross sections obtained from an ENDF-format evaluation by way of the RECONR, BROADR, UNRESR, HEATR, and THERMR modules of NJOY, and the shape of ϕ within the group.

B. Group Ordering

Since neutrons normally lose energy in scattering, the scattering source into group g depends on the flux at higher-energy groups g' and the cross section for transferring neutrons from g' to g. For this reason, Eq. (186) is usually solved by sweeping from high energies to low energies. (Any thermal upscatter or fission is handled by iteration.) Data libraries for use with transport codes normally number the groups such that group 1 is the highest-energy group, and all the scattering matrix elements that transfer neutrons into group 1 are given first, followed by those for scattering into group 2, and so on.

However, in ENDF files, the evaluated nuclear data are always given in order of increasing incident energy, and secondary neutron distributions are described by giving emission spectra for given incident energies. Therefore, GROUPR numbers its groups such that group 1 is the lowest-energy group, and it calculates that scattering out

of group 1, followed by the scattering out of group 2, and so on.

The "backward" energy-group numbering convention used by GROUPR is a possible source of confusion in interpreting output produced by the various modules of NJOY. All group indices printed by GROUPR or written to the GROUPR output file use the increasing-energy order. The covariance modules and photon interaction module follow the GROUPR ordering convention. Output modules such as DTFR, CCCCR, MATXSR, and WIMSR invert the group order and rearrange the scattering matrices from the GROUPR outscatter organization to the "transport" inscatter form. Any group indices printed by these three output modules will be in the conventional transport decreasing-energy order.

C. Basic ENDF Cross Sections

The basic energy-dependent cross sections and energy-angle distributions needed for Eqs. (189), (190), and (191) are obtained from evaluated nuclear data in ENDF format [2]. These data are indexed by material (MAT), type of information (MF), and reaction (MT). Materials can be single isotopes, elements, or compounds. Type of information includes energy-dependent cross section (MF=3), angular distributions (MF=4), secondary energy distributions (MF=5), and energy-angle distributions (MF=6). Reactions include the total (MT=1) required for Eq. (189), the partial scattering reactions that must be summed for Eq. (191) [that is, elastic (MT=2), discrete-level inelastic (MT=51–90), (n,2n) (MT=16), *etc.*], and the many partial reactions that can be used to calculate responses [for example, (n,2n) or (n,α) for activation, gas production, heat production (KERMA), radiation damage (DPA), *etc.*].

Before using GROUPR, the basic ENDF/B cross sections should have been converted into energy- and temperature-dependent pointwise cross sections in PENDF (pointwise ENDF) format using RECONR and BROADR. For heavy isotopes, unresolved self-shielding data should have been added to the PENDF file by PURR. If needed, heat production cross sections (KERMA), radiation damage production (or DPA), and thermal upscatter data could have also been added to the PENDF file using HEATR and THERMR. See the sections on these other modules for more information.

An example of an ENDF pointwise cross section compared with group-averaged cross sections from GROUPR is given in Fig. 39.

D. Weighting Flux

In general, the weighting flux ϕ is not known; it is, after all, the particle distribution being sought in the transport calculation. However, it is often possible to obtain fairly accurate group constants for a particular application if

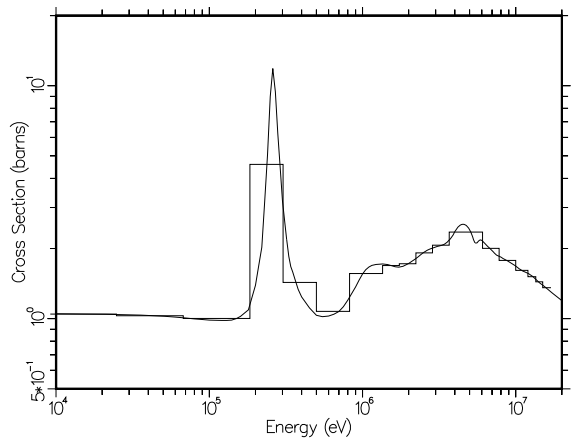

FIG. 39: A comparison of the pointwise and multigroup representations for the total cross section of Li-7. The Los Alamos 30-group structure is shown.

the shape of the flux is reasonably well known over the broad energy ranges of a particular few-group structure (for example, a fission spectrum, thermal Maxwellian, or $1/E$ slowing-down spectrum). Alternatively, one can use many small groups so that mistakes in guessing the shape inside the group are not very important. The key to using the multigroup method effectively is balancing the tradeoffs between the choice of weight function and the number of groups used for each different class of problem being solved.

In many cases of practical interest, the flux ϕ will contain dips corresponding to the absorption resonances of the various materials. In the reaction rate $\sigma(E) \times \phi(E)$, these dips clearly reduce (self-shield) the effect of the corresponding resonance. GROUPR provides two methods to estimate the effect of this self-shielding: the Bondarenko model and the flux calculator.

In the Bondarenko model [41], the narrow resonance (NR) approximation, and the $\mathrm{B_N}$ approximation for large systems [42] are invoked to obtain

$$\phi_\ell(E) = \frac{W_\ell(E)}{[\sigma_t(E)]^{\ell+1}} \ , \tag{192}$$

where ϕ_ℓ is the ℓ-th Legendre component of the angular flux, the $W_\ell(E)$ are smooth functions of energy (such as $1/E$+fission), and $\sigma_t(E)$ is the total macroscopic cross section for the material. GROUPR takes all of the W_ℓ to be equal to the single function $C(E)$, where $C(E)$ can be read in or set to one of several internally defined functions. It is further assumed that the important self-shielding effect of the flux can be obtained for isotope i by representing all the other isotopes with a constant "background cross section", σ_0. Therefore,

$$\phi^i_\ell(E) = \frac{C(E)}{[\sigma^i_t(E) + \sigma^i_0]^{\ell+1}} \ , \tag{193}$$

where σ_t^i is the microscopic total cross section for isotope i. The qualitative behavior of Eq. (193) is easy to understand. If σ_0 is larger than the tallest peaks in σ_t, the weighting flux ϕ is approximately proportional to the smooth weighting function $C(E)$. This is called infinite dilution; the cross section in the material of interest has little or no effect on the flux. On the other hand, if σ_0 is small with respect to σ_t, the weighting flux will have large dips at the locations of the peaks in σ_t, and a large self-shielding effect will be expected.

Each component material of a mixture has a different weight function. The macroscopic total cross sections are given by

$$\sigma_{t\ell g} = \sum_i \rho_i \, \sigma_{t\ell g}^i(\sigma_0^i, T) \ , \tag{194}$$

where

$$\sigma_{t\ell g}^i(\sigma_0, T) = \frac{\displaystyle\int_g \frac{\sigma_t^i(E,T)}{[\sigma_0 + \sigma_t^i]^{\ell+1}} C(E)\, dE}{\displaystyle\int_g \frac{1}{[\sigma_0 + \sigma_t^i]^{\ell+1}} C(E)\, dE} \ . \tag{195}$$

Similar equations are used for σ_R and $\sigma_{X\ell g \to g'}$. On the GROUPR level, σ_0 and T are simply parameters. Subsequent codes, such as TRANSX [43], can compute an appropriate value for σ_0, and then interpolate in tables of cross section versus σ_0 and T to get the desired self-shielded group constants.

The appropriate value for σ_0^i is obvious when a single resonance material is mixed with a moderator material (for example, $^{238}UO_2$), because the admixed materials typically have a constant cross section in the energy range where the heavy isotopes have resonances. For a mixture of resonance materials, the normal procedure is to preserve the average of Eq. (193) in each group by using

$$\sigma_{0g}^i = \frac{1}{\rho_i} \sum_{j \neq i} \rho_j \, \sigma_{t0g}^j(\sigma_{0g}^j, T) \ , \tag{196}$$

where the ρ_i are atomic densities or atomic fractions. Eq. (196) is solved by iteration. Interference between resonances in different materials is handled in an average sense only.

When heterogeneity effects are important, the background cross section method can be extended as follows. In an infinite system of two regions (fuel and moderator), the neutron balance equations are

$$V_f \sigma_f \phi_f = (1 - P_f) V_f S_f + P_m V_m S_m \ , \tag{197}$$

and

$$V_m \sigma_m \phi_m = P_f V_f S_f + (1 - P_m) V_m S_m \ , \tag{198}$$

where V_f and V_m are the region volumes, σ_f and σ_m are the corresponding total macroscopic cross sections, S_f and S_m are the sources per unit volume in each region, P_f is the probability that a neutron born in the fuel will suffer its next collision in the moderator, and P_m is the probability that a neutron born in the moderator will suffer its next collision in the fuel. As usual, use is made of the reciprocity theorem,

$$V_f \sigma_f P_f = V_m \sigma_m P_m \ , \tag{199}$$

and the Wigner rational approximation to the fuel escape probability,

$$P_f = \frac{\sigma_e}{\sigma_e + \sigma_f} \ , \tag{200}$$

where σ_e is a slowly varying function of energy called the escape cross section, to obtain an equation for the fuel flux in the form

$$(\sigma_f + \sigma_e)\phi_f = \frac{\sigma_e S_m}{\sigma_m} + S_f \ . \tag{201}$$

In the limit where the resonances are narrow with respect to both fuel and moderator scattering, the source terms S_f and S_m take on their asymptotic forms of σ_p/E and σ_m/E respectively, and this equation becomes equivalent to the Bondarenko model quoted above with

$$\sigma_0^f = \frac{\sigma_e}{\rho_f} \ , \tag{202}$$

and

$$C(E) = \frac{\sigma_e + \sigma_p}{\rho_f E} \ . \tag{203}$$

Note that a large escape cross section (a sample that is small relative to the average distance to collision), corresponds to infinite dilution as discussed above. To illustrate the general case, consider a neutron traveling through a lump of uranium oxide with an energy close to a resonance energy. If the neutron scatters from an oxygen nucleus, it will lose enough energy so that it can not longer react with the uranium resonance. Similarly, if the neutron escapes from the lump, it can no longer react with the uranium resonance. The processes of moderator scattering and escape are equivalent in some way. Comparing Eq. (202) with Eq. (196) gives an "equivalence principle" that says that a lump of particular dimensions and a mixture of particular composition will have the same self-shielded cross sections when the narrow resonance approximation is valid. The effects of material mixing and escape can simply be added to obtain the effective σ_0 for a lump containing admixed moderator material. Therefore, Eq. (196) is extended to read

$$\sigma_{0g}^i = \frac{1}{\rho_i} \left\{ \sigma_e + \sum_{j \neq i} \rho_j \, \sigma_{t0g}^j(\sigma_{0g}^j, T) \right\} \ , \tag{204}$$

where the escape cross section for simple convex objects (such as plates, spheres, or cylinders) is given by $(4V/S)^{-1}$, where V and S are the volume and surface

area of the object, respectively. Many codes that use the background cross section method modify the escape cross section as defined above to correct for errors in the Wigner rational approximation ("Bell factor", "Levine factor"), or to correct for the interaction between different lumps in the moderating region ("Dancoff corrections"). These enhancements will not be discussed here.

As an example of self-shielded cross sections and how they vary with the background cross section, Fig. 40 shows the first three capture resonances of U-238 (which are very important for thermal power reactors) at room temperature for σ_0 values ranging from infinity down to 10 barns. The range between 20 and 50 barns is typical for reactor pin cells.

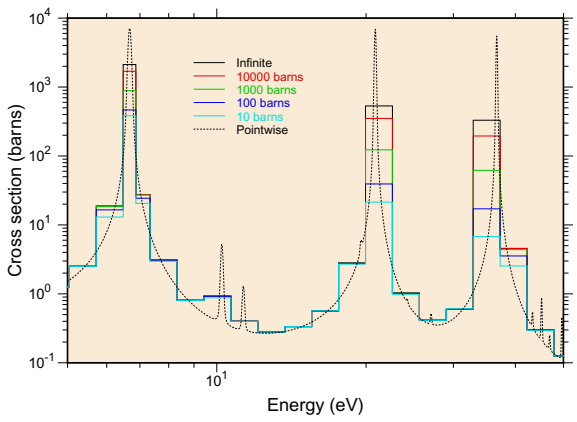

FIG. 40: The self-shielding effect on the first three U-238 capture resonances at room temperature in the 5 to 50 eV range. The multigroup boundaries are from the Los Alamos 187-group structure.

The BROADR section of this report showed its capability to compute standard resonance integrals. However, when self shielding and material temperature come into play, the effective resonance integral changes. GROUPR can calculate these quantities by doing one-group calculations with $1/E$ weighting over a standard energy range. Using the range 0.5 eV to the upper limit of the evaluation will match what BROADR does. Fig. 41 shows the temperature dependence of the capture resonance integral for U-238 from ENDF/B-VII at several different background cross sections. The range between 20 and 50 barns of background is typical for uranium-oxide pin cells.

E. Flux Calculator

This narrow-resonance approach is quite useful for practical fast reactor problems. However, for nuclear systems sensitive to energies from 1 to 500 eV, there are many broad- and intermediate-width resonances which cannot be self-shielded with sufficient accuracy using the

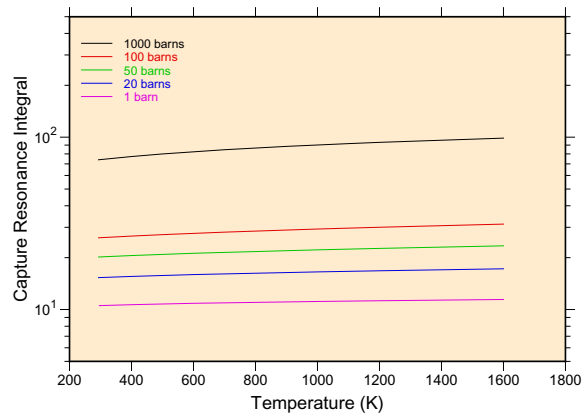

FIG. 41: The capture resonance integral for U-238 showing its variation with temperature and background cross section σ_0. The resonance integral at infinite dilution is 275.58 barns. Note the slope with temperature, which helps to produce a negative temperature coefficient for uranium systems.

Bondarenko approach. The GROUPR flux calculator is designed for just such problems.

Consider an infinite homogeneous mixture of two materials and assume isotropic scattering in the center-of-mass system. The integral slowing-down equation becomes

$$\sigma(E)\,\phi(E) = \int_E^{E/\alpha_1} \frac{\sigma_{s1}(E')}{(1-\alpha_1)E'}\,\phi(E')\,dE' + \int_E^{E/\alpha_2} \frac{\sigma_{s2}(E')}{(1-\alpha_2)E'}\,\phi(E')\,dE' \quad (205)$$

where α_1 and α_2 are the downscatter ranges $(A-1)^2/(A+1)^2$ for the two materials. Furthermore, assume that material 1 is a pure scatterer with constant cross section and transform to the σ_0 representation. The integral equation becomes

$$[\sigma_0 + \sigma_{t2}(E)]\,\phi(E) = \int_E^{E/\alpha_1} \frac{\sigma_0}{(1-\alpha_1)E'}\,\phi(E')\,dE' + \int_E^{E/\alpha_2} \frac{\sigma_{s2}(E')}{(1-\alpha_2)E'}\,\phi(E')\,dE' \,. \quad (206)$$

Finally, assume that the moderator (material 1) is light enough so that all the resonances of material 2 are narrow with respect to scattering from material 1. This allows the first integral to be approximated by its asymptotic form $1/E$. More generally, the integral is assumed to be a smooth function of E given by $C(E)$. In this way, material 1 can represent a mixture of other materials just as in the Bondarenko method. Fission source and thermal upscatter effects can also be lumped in $C(E)$.

The integral equation has now been reduced to

$$[\sigma_0 + \sigma_t(E)]\,\phi(E) = C(E)\,\sigma_0 + \int_E^{E/\alpha} \frac{\sigma_s(E')}{(1-\alpha)E'}\,\phi(E')\,dE' \ . \quad (207)$$

This is the simplest problem that can be solved using the flux calculator. The results still depend on the single parameter σ_0, and they can be used easily by codes that accept Bondarenko cross sections.

For heterogeneous problems, when the narrow-resonance approximation fails, both S_f and S_m in Eq. (201) will show resonance features. To proceed further with the solution of this equation, it is necessary to eliminate the moderator flux that is implicit in S_m. As a sample case, consider a fuel pin immersed in a large region of water. The fission neutrons appear at high energies, escape from the pin, slow down in the moderator (giving a $1/E$ flux), and are absorbed by the resonances in the pin. In this limit, any dips in the moderator flux caused by resonances in the fuel are small. On the other hand, in a closely packed lattice, the flux in the moderator is very similar to the flux in the fuel, and resonance dips in the moderator flux become very evident. Intermediate cases can be approximated [44] by assuming

$$\phi_m = (1-\beta)\,C(E) + \beta\phi_f \ , \quad (208)$$

where β is a heterogeneity parameter given by

$$\beta = \frac{V_f\sigma_e}{V_m\sigma_m} \ . \quad (209)$$

Note that $\beta \to 0$ gives the isolated rod limit and $\beta \to 1$ gives the close-packed lattice limit. This substitution reduces the calculation of the fuel flux to

$$(\sigma_f + \sigma_e)\,\phi_f = (1-\beta)\,C(E)\,\sigma_e + S_\beta \ , \quad (210)$$

where S_β is the source term corresponding to a homogeneous mixture of the fuel isotopes with the isotopes from the moderator region changed by the factor $\beta\sigma_e/\sigma_m$. If the fuel and moderator each consisted of a single isotope and for isotropic scattering in the center-of-mass system, the integral equation would become

$$[\sigma_0 + \sigma_t(E)]\,\phi_f(E) = (1-\beta)\,C(E)\,\sigma_0 + \int_E^{E/\alpha_m} \frac{\beta\sigma_0}{(1-\alpha_m)E'}\,\phi_f(E')\,dE' + \int_E^{E/\alpha_f} \frac{\sigma_{sf}(E')}{(1-\alpha_f)E'}\,\phi_f(E')\,dE' \ , \quad (211)$$

where σ_0 is σ_e divided by the fuel density (units are barns/atom), α_m and α_f are the maximum fractional energy change in scattering for the two isotopes (m for moderator, f for fuel), and $\sigma_{sf}(E')$ is the fuel scattering cross section.

This result has a form parallel to that of Eq. (207), but the solution depends on the two parameters β and σ_0. For any given data set, β must be chosen in advance. This might not be difficult if the data are to be used for one particular system, such as pressurized water reactors. The routine also has the capability to include one more moderator integral with a different α value and a constant cross section. The full equation is

$$[\sigma_0 + \sigma_t(E)]\,\phi_f(E) = (1-\beta)\,C(E)\,\sigma_0 + \int_E^{E/\alpha_3} \frac{\beta(1-\gamma)(\sigma_0-\sigma_{am}}{(1-\alpha_3)E'}\,\phi_f(E')\,dE' + \int_E^{E/\alpha_2} \frac{\sigma_{am}+\beta\gamma(\sigma_0-\sigma_{am}}{(1-\alpha_2)E'}\,\phi_f(E')\,dE' + \int_E^{E/\alpha_f} \frac{\sigma_{sf}(E')}{(1-\alpha_f)E'}\,\phi_f(E')\,dE' \ , \quad (212)$$

where σ_{am} is the cross section of the admixed moderator (with energy loss α_2), and γ is the fraction of the admixed moderator that is mixed with the external moderator (which has energy loss α_3). This allows calculations with H_2O as the moderator and an oxide as the fuel. The flux calculator can thus obtain quite realistic flux shapes for a variety of fuel, admixed moderator, and external moderator combinations. An example comparing the Bondarenko flux with a more realistic computed flux is given in Fig. 42.

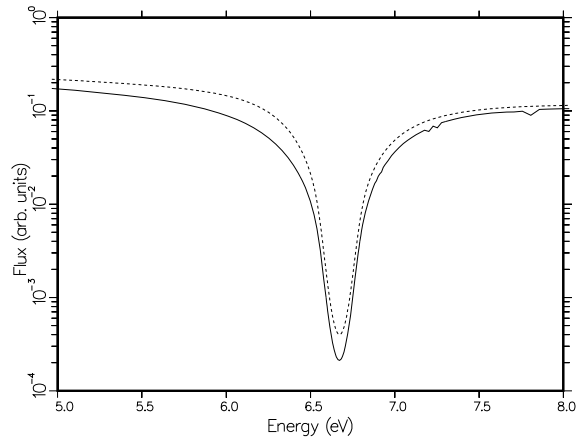

FIG. 42: A comparison of the Bondarenko flux model (dashed) with a realistic computed flux (solid) for a ^{238}U oxide pin in water in the region of the 6.7 eV resonance.

In practice, a fuel rod rarely contains only one resonance isotope. As an example, consider a mixture of a few percent of ^{239}Pu with ^{238}U as the major component. There will be a strong dip in the flux associated with the 6.7 eV ^{238}U resonance that will affect the flux in the region of the 7.8 eV ^{239}Pu resonance (the interference effect), and there will also be a dip in the flux corresponding to the 7.8 eV resonance (the self-shielding effect). This additional complication in the flux shape would be expected to change the group constants for ^{239}Pu since both features lie in the same group for typical

group structures. However, the effect of the ^{239}Pu on the ^{238}U group constants should be minimal. This argument suggests that the full flux calculation be used for ^{238}U as a single resonance material. The resulting flux would then be used to estimate the flux to be used in averaging the ^{239}Pu cross sections as follows:

$$\phi_{239}(E,\sigma_0) = \frac{\phi_{238}(E,50)}{\sigma_0 + \dfrac{\sigma_{239}(E)}{1+\dfrac{\sigma_{238}(E)}{50}}} \ , \tag{213}$$

where the ^{238}U flux is characteristic of a background of 50 barns/atom, which is representative of many thermal reactor systems. This formula assumes that the effect of ^{239}Pu on the scattering source for the mixture is small, but it retains the absorption effects. The self-shielding of ^{239}Pu is treated in the narrow resonance approximation only. The GROUPR flux calculator includes an option to write out a file containing the calculated flux and cross section needed for this formula (*e.g.*, for ^{238}U) and another option to skip the flux calculation and use the formula above to obtain the weighting flux (*e.g.*, for ^{239}Pu).

The slowing-down integral equation of Eq. (207) or (211) is solved point by point using the total and elastic cross sections on the PENDF file produced by RECONR. In order to keep this task within bounds, the flux is computed from the lower limit of the first group to a specified energy `ehi` or until `nfmax` values have been computed. The flux at higher energies is continued using the Bondarenko model described above.

F. Fission Source

The fission source was included in the transfer cross section σ_X in the development above. It is usually convenient to separate fission from scattering. Assuming isotropy, the source of fission neutrons into group g is given by

$$S_g = \sum_{g'} \sigma_{fg'\to g}\, \phi_{0g'} \ , \tag{214}$$

where the group-to-group matrix for fission is defined as in Eq. (191), but with ℓ equal to zero. Most existing transport codes do not use this matrix form directly because the upscatter is expensive to handle and a reasonably accurate alternative exists. Except for relatively high neutron energies, the spectrum of fission neutrons is only weakly dependent on initial energy. Therefore, the fission source can be written

$$S_g = \chi_g \sum_{g'} \overline{\nu}_{g'}\, \sigma_{fg'}\, \phi_{0g'} \ , \tag{215}$$

where $\overline{\nu}_g$ is the fission neutron yield, σ_{fg} is the fission cross section, and χ_g is the average fission spectrum (the familiar "chi" vector), which can be defined by

$$\chi_g = \frac{\sum_{g'} \sigma_{fg'\to g}\, \phi_{0g'}}{\sum_g \sum_{g'} \sigma_{fg'\to g}\, \phi_{0g'}} \ . \tag{216}$$

The fission neutron yield is given by

$$\overline{\nu}_g = \sum_{g'} \sigma_{fg\to g'} \,/\, \sigma_{fg} \ . \tag{217}$$

Clearly, χ_g as given by Eq. (216) depends on the flux in the system of interest. The dependence is weak except for high incident energies, and a rough guess for ϕ_{0g} usually gives an accurate spectrum. When this is not the case, the problem can be iterated, or the full matrix representation can be used.

It is possible to take advantage of the weak energy dependence of the shape of the fission spectrum at low energies to reduce the time required to process fission and to reduce the size of the fission data on the output file. GROUPR determines a break energy from File 5 such that the fission spectrum is constant below this energy by comparing the shapes of adjacent distributions. It only has to make a single calculation of this spectrum χ_g^{LE}. Then it computes a fission neutron production cross section σ_{fPg}^{HE} for groups below the break energy using the normal cross section processing methods. A full matrix $\sigma_{fg'\to g}^{HE}$ is computed for groups above the break point. As an example, consider using the GROUPR 187-group structure and finding that there are 130 constant groups. The 187×187 matrix is reduced to a 57×187 matrix, a 187-element spectrum vector, and a 130-group production vector, for a total reduction in size of 68%. The effective fission matrix is given by

$$\sigma_{fg'\to g} = \sigma_{fg'\to g}^{HE} + \chi_g^{LE} \sigma_{fPg'}^{LE} \ . \tag{218}$$

The fission matrix computed by GROUPR represents the prompt part of fission only. The delayed component of fission is represented by a delayed-neutron yield $\overline{\nu}_g^D$, decay constants for six time groups, λ_i^D, and emission probability spectra for six time groups, χ_{ig}^D. Steady-state fission can be obtained using

$$\overline{\nu}_g^{SS} = \overline{\nu}_g + \overline{\nu}_g^D \ , \tag{219}$$

and

$$\chi_g^{SS} = \frac{\sum_{g'} \sigma_{fg'\to g}\, \phi_{0g'} + \chi_g^D \sum_{g'} \overline{\nu}_{g'}^D\, \sigma_{fg'}\, \phi_{0g'}}{\sum_g \sum_{g'} \sigma_{fg'\to g}\, \phi_{0g'} + \sum_{g'} \overline{\nu}_{g'}^D\, \sigma_{fg'}\, \phi_{0g'}} \ , \tag{220}$$

where

$$\chi_g^D = \sum_i \chi_{ig}^D \ . \tag{221}$$

Note that χ_g^D sums to unity, but the χ for each time group sums to the fraction of the delayed neutron yield that appears in that time group.

In ENDF-format files, the total fission reaction is represented by MT=18. Important isotopes also give the partial fission reactions (n,f), (n,n′f), (n,2nf), and sometimes (n,3nf) using MT=19, 20, 21, and 38 respectively. The MT=18 representation is adequate for most fission reactor applications, but the partial reactions should be processed for applications with significant flux above 6 MeV. Caution: although the cross section for MT=18 equals the sum of its parts, the group-to-group fission matrix $\sigma_{fg\to g'}$ computed from MT=18 will not, in general, equal the sum of the partial matrices for MT=19, 20, 21, and 38 above the 6-MeV threshold for second-chance fission. The delayed neutron data are given in MT=455. Sample input instructions for processing the various combinations of fission reactions used in ENDF/B will be found below. GROUPR outputs all the components of fission separately in order to give succeeding modules or codes complete flexibility.

G. Photon Production and Coupled Sets

Photon transport is treated with an equation similar to Eq. (186), except that the flux, cross sections, and groups are all referred to photon interactions and photon energies instead of to the corresponding neutron quantities. Methods of producing the photon interaction cross sections are described in the GAMINR section of this manual. The external photon source Q_g depends on the neutron flux and the photon production cross sections through

$$Q_g(x,\mu) = \sum_{\ell=0}^{\infty} \frac{2\ell+1}{2} P_\ell(\mu) \times \sum_{g'} \sigma_{\gamma\ell g'\to g}(x)\,\phi_{\ell g'}(x) \ , \qquad (222)$$

where $\sigma_{\gamma\ell g'\to g}$ is defined by Eq. (191) with X replaced by γ. The ENDF files define σ_γ using a combination of photon production cross sections (MF=13), photon yields (MF=12) with respect to neutron cross sections (MF=3), discrete lines (MF=12 and 13), and continuous γ distributions (MF=15). Methods for working with these representations will be discussed in more detail below.

The low-energy groups for fission and capture normally have photon emission spectra whose shapes do not change with energy. The same method used for reducing the size of the fission matrix described above can be used for these photon production matrices. In mathematical form,

$$\sigma_{\gamma g'\to g} = \sigma^{HE}_{\gamma g'\to g} + s^{LE}_\gamma g \sigma^{LE}_{\gamma P g'} \ , \qquad (223)$$

where $s^{LE}_\gamma g$ is the normalized emission spectrum, and $\sigma^{LE}_{\gamma P g}$ is the associated production cross section.

H. Thermal Data

GROUPR uses the thermal data written onto the PENDF file by the THERMR module. It does not process ENDF File-7 data directly. Three different representations of thermal scattering are used in ENDF.

For crystalline materials like graphite, coherent elastic (with zero energy change) scattering can take place. The cross section for this process shows well-defined Bragg edges at energies that correspond to the various lattice-plane spacings in the crystalline powder. As shown in the THERMR section, the angular dependence of the coherent elastic cross section can be written as

$$\sigma^{coh}(E,\mu) = \sum_i \frac{b_i}{E}\,\delta(\mu-\mu_i) \ , \qquad (224)$$

where

$$\mu_i = 1 - 2\frac{E_i}{E} \ , \qquad (225)$$

and where the E_i are the energies of the Bragg edges. THERMR integrates Eq. (224) over all angles, and writes the result to the PENDF file. Clearly, the b_i can be recovered from

$$E\,\sigma^{coh}(E) = \sum_i{}' \, b_i \ , \qquad (226)$$

where the primed sum is over all i such that $E_i < E$. It is only necessary to locate the steps in $E\,\sigma^{coh}(E)$. The size of the step gives b_i, and the E for the step gives E_i. The Legendre cross sections become

$$\sigma^{coh}_\ell(E) = \sum_i{}' \frac{b_i}{E} P_\ell(\mu_i) \ , \qquad (227)$$

where any terms with $\mu_i < -1$ are omitted from the primed sum. An example of a pointwise cross section for coherent elastic scattering is given in Fig. 43.

For hydrogenous solids like polyethylene and zirconium hydride, the process of incoherent elastic scattering is important. Here the angular cross section is given by

$$\sigma^{iel}(E,\mu) = \frac{\sigma_b}{2}\exp\left[-\frac{2EW}{A}(1-\mu)\right] \ , \qquad (228)$$

THERMR converts this into an integrated cross section, $\sigma^{iel}(E)$, and a set of N equally probable emission cosines, $\overline{\mu}_i$. These angles are present in File 6 on the PENDF file. GROUPR can easily determine the Legendre components of the scattering cross section using

$$\sigma^{iel}_\ell(E) = \sigma^{iel}(E)\frac{1}{N}\sum_{i=1}^{N} P_\ell(\mu_i) \ . \qquad (229)$$

The third thermal process is incoherent inelastic scattering. Here the neutron energy can either increase or decrease. The data from THERMR are given as a cross

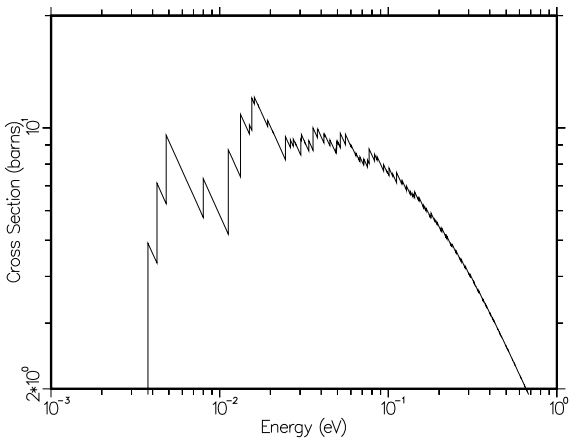

FIG. 43: The coherent elastic scattering cross section for BeO showing the Bragg edges. The shape of $\sigma(E)$ between edges is $1/E$. Therefore, the function $E\sigma(E)$ is a stair-step function, where the height of each step depends on the structure factor for scattering from that set of lattice planes [see Eq. (226)].

section in File 3 and an energy-angle distribution in File 6. The distribution is represented by sets of secondary-energy values E' for particular incident energies E. For each $E \rightarrow E'$, a scattering probability $f^{inc}(E \rightarrow E')$ and a set of equally probable cosines $\mu_i(E \rightarrow E')$ are given. The scattering probabilities for each value of E integrate to unity. Although the thermal scattering cross section is a smooth function of incident neutron energy, this is not true for the scattering from E to one particular final energy group g', since the differential cross section tends to peak along the line $E'=E$ and at energy-transfer values corresponding to well-defined excitations in the molecule or lattice. If interpolation between adjacent values of E were to be performed along lines of constant E', the excitation peaks and the $E=E'$ feature would produce double features in the intermediate spectrum, as shown in Fig. 44. To avoid this problem, while still using a relatively sparse incident energy grid, GROUPR interpolates between E and E' along lines of constant energy transfer. Of course, this breaks down at low values of E', because one of the spectra will go to zero before the other one does. In this range, GROUPR transforms the low-energy parts of the two spectra onto a "unit base," combines them in fractions that depend on E, and scales the result back out to the interpolated value of E' corresponding to E.

I. Generalized Group Integrals

In order to unify many formerly different processing tasks, GROUPR uses the concept of a generalized group

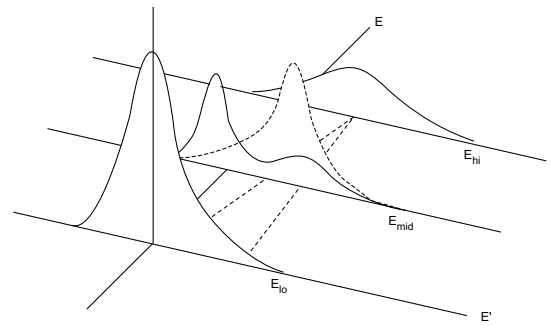

FIG. 44: Illustration of thermal interpolation showing the double-humped curve resulting from simple Cartesian interpolation for a discrete excitation (solid) and the more realistic curve obtained by interpolating along lines of constant energy transfer (dashed).

integral

$$\sigma_g = \frac{\displaystyle\int_g \mathcal{F}(E)\,\sigma(E)\,\phi(E)\,dE}{\displaystyle\int_g \phi(E)\,dE} \ , \qquad (230)$$

where the integrals are over all initial neutron energies in group g, $\sigma(E)$ is a cross section at E, and $\phi(E)$ is an estimate of the flux at E. The function $\mathcal{F}(E)$ is called the "feed function". It alone changes for different data types. To average a neutron cross section, $\mathcal{F}$ is set to 1. To average a ratio quantity like $\overline{\mu}$ with respect to elastic scattering, $\mathcal{F}$ is set to $\overline{\mu}$. For photon production, $\mathcal{F}$ is the photon yield. For matrices, $\mathcal{F}$ is the ℓ-th Legendre component of the normalized probability of scattering into secondary energy group g' from initial energy E. This definition is clearly independent of whether the secondary particle is a neutron or a photon.

The generalized integration scheme discussed here is also used by the GAMINR and ERRORR modules.

J. Two-Body Scattering

Elastic scattering (MT=2) and discrete inelastic neutron scattering (with MT=51–90) are both examples of two-body kinematics and are treated together by GROUPR. Some other cases that occur for charged particles or in File 6 will be discussed later. The feed function required for the group-to-group matrix calculation may be written

$$\begin{aligned} \mathcal{F}_{\ell g'}(E) \ &= \ \int_{g'} dE' \int_{-1}^{+1} d\omega \\ &\times \ f(E \rightarrow E', \omega)\, P_\ell(\mu[\omega]) \ , \end{aligned} \qquad (231)$$

where $f(E \rightarrow E', \omega)$ is the probability of scattering from E to E' through a center-of-mass cosine ω and P_ℓ is a

Legendre polynomial for laboratory cosine μ. The laboratory cosine corresponding to ω is given by

$$\mu = \frac{1 + R\omega}{\sqrt{1 + R^2 + 2R\omega}} \ , \tag{232}$$

and the cosine ω is related to secondary energy E' by

$$\omega = \frac{E'(1+A)^2/A' - E(1+R^2)}{2RE} \ , \tag{233}$$

where A' is the ratio of the emitted particle mass to the incident particle mass (A'=1 for neutron scattering). In Eqs. (232) and (233), R is the effective mass ratio

$$R = \sqrt{\frac{A(A+1-A')}{A'}}\sqrt{1 - \frac{(A+1)(-Q)}{AE}} \ , \tag{234}$$

where A is the ratio of target mass to neutron mass, and $-Q$ is the energy level of the excited nucleus (Q=0 for elastic scattering). Integrating Eq. (231) over secondary energy gives

$$\mathcal{F}_{\ell g'}(E) = \int_{\omega_1}^{\omega_2} f(E,\omega)\, P_\ell(\mu[\omega])\, d\omega \ , \tag{235}$$

where ω_1 and ω_2 are evaluated using Eq. (233) for E' equal to the upper and lower bounds of g', respectively. The scattering probability is given by

$$f(E,\omega) = \sum_{\ell=0}^{L} f_\ell(E)\, P_\ell(\omega) \ , \tag{236}$$

where the Legendre coefficients are either retrieved directly from the ENDF File 4 or computed from File 4 tabulated angular distributions.

The integration in Eq. (235) is performed simultaneously for all Legendre components using Gaussian quadrature [45]. The quadrature order is selected based on the estimated polynomial order of the integrand. A reasonable estimate is given by

$$\mathrm{ND} + \mathrm{NL} + \log(300/A) \ , \tag{237}$$

where ND is the number of Legendre components desired for the feed function, and NL is the number of components required to represent $f(E,\omega)$. The log term approximates the number of additional components required to represent the center-of-mass to lab transformation.

The two-body feed function for higher Legendre orders is a strongly oscillatory function in some energy ranges. An example is shown in Fig. 45. Furthermore, the integral of the oscillatory part is often small with respect to the magnitude of the function. Such functions are very difficult to integrate with adaptive techniques, which converge to some fraction of the integral of the absolute value. Gaussian methods are capable of integrating such oscillatory functions exactly if they are polynomials. Since a polynomial representation of the feed function is

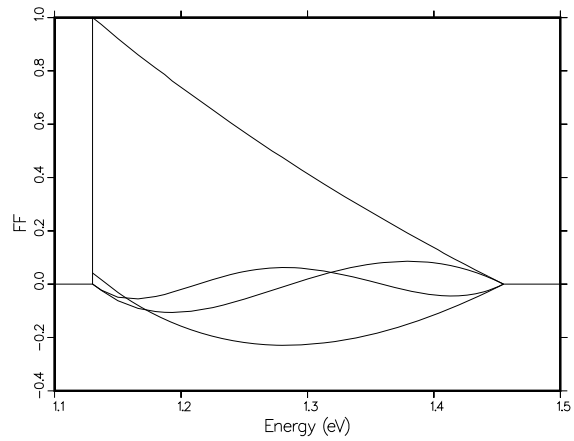

FIG. 45: A typical feed function (FF) for two-body scattering showing the oscillations that must be treated correctly by the integration over incident energy. The P_1 function is almost linear, and each of the P_2, P_3, and P_4 functions shows an additional oscillation as the order increases.

fairly accurate, a Gaussian quadrature scheme was chosen. The scheme used is also well adapted to performing many integrals in parallel. In GROUPR, all Legendre components and all final groups are accumulated simultaneously.

The boundaries of the various regions of the feed function are called "critical points." Between critical points, the feed function is a smooth analytic function of approximately known polynomial order. It is only necessary to add these critical points to the incident energy grid of the feed function and to determine what quadrature order to use. The critical points are determined by solving Eq. (233) for the values of E for which $\omega = +1$ and $\omega = -1$ when E' is equal to the various group boundaries. This can be done by writing

$$\frac{E_g}{E} = \frac{1}{(1+A)^2}\left[R^2 \pm 2R + 1\right] \ , \tag{238}$$

substituting for E using Eq. (234), and then solving for R. The result is

$$E_{crit} = \frac{\dfrac{(A+1)(-Q)}{A}}{1 - \dfrac{A'F^2}{A(A+1-A')}} \ , \tag{239}$$

where

$$F = \frac{1 \pm \sqrt{D}}{1 + \dfrac{E_F}{(-Q)}} \ , \tag{240}$$

$$D = \left[\frac{A(A+1-A')}{A'}\left(1 + \frac{E_F}{(-Q)}\right) - 1\right]\frac{E_F}{(-Q)} \ , \tag{241}$$

and

$$E_F = \frac{1+A}{A+1-A'} E_g \ . \tag{242}$$

File 4 can also contain angular distributions for charged-particle emission through discrete levels (see MT=600–648 for protons, MT=650–698 for deuterons, and so on; the elastic case, MT=2, is discussed in the next section). Moreover, File 6 can contain angular distributions for discrete two-body scattering (see Law 3). It can also declare that a particular particle is the recoil particle from a two-body reaction (Law 4), in which case the appropriate angular distribution is obtained from the corresponding Law 3 subsection by complementing the angle. The representation of the angular distribution for these laws is almost identical to that in File 4, and the calculation uses much of the File 4 coding. The effects on the kinematics of the difference in the mass of the emitted and incident particles are handled by the variable A' in the above equations.

K. Charged-Particle Elastic Scattering

Coulomb scattering only occurs in the elastic channel. The problem with Coulomb scattering is that the basic Rutherford formula becomes singular at small angles. In practice, this singularity is removed by the screening effects of the atomic electrons. The forward transport of charged particles in this screening regime is usually handled by continuous-slowing-down theory by using a "stopping power". The new ENDF-6 format allows for three different ways to handle the large-angle scattering regime. First and most general is the nuclear amplitude expansion:

$$\begin{aligned} \sigma &= |\text{nucl} + \text{coul}|^2 \\ &= \sigma_{\text{nucl}} + \sigma_{\text{coul}} + \text{interference} \end{aligned} \tag{243}$$

The Coulomb term is given by the Rutherford formula, a Legendre expansion is defined for the nuclear term, and a complex Legendre expansion is defined for the interference term. This representation cannot be generated directly from experimental data; an R-matrix or phase-shift analysis is necessary.

A method very closely related to experiment (σ_{exp}) is the nuclear plus interference (NI) formula:

$$\sigma_{\text{NI}}(\mu, E) = \sigma_{\text{exp}}(\mu, E) - \sigma_{\text{coul}}(\mu, E), \tag{244}$$

where the function is only defined for angles with cosines $\mu<\mu_{\text{max}}$. The minimum angle is usually taken to be somewhere around 20 degrees (GROUPR uses μ_{max}=0.96). This function is still ill-behaved near the cutoff, and it must be tabulated. The third option is the residual cross section expansion:

$$\sigma_R(\mu, E) = (1-\mu)\big[\sigma_{\text{exp}}(\mu, E) - \sigma_{\text{coul}}(\mu, E)\big]. \tag{245}$$

The $(1-\mu)$ term removes the pole at the origin. The residual is uncertain, but it is usually small enough that the entire curve can be fitted with Legendre polynomials without worrying about what happens at small angles. In practice, both the nuclear amplitude expansion and the nuclear plus interference representation are converted to the residual cross section representation and processed in the same way.

L. Continuum Scattering and Fission

In older ENDF evaluations, scattering from many closely-spaced levels or multibody scattering such as (n,2n), (n,n′α) was often represented using a separable function of scattering cosine and secondary neutron energy

$$f(E \to E', \mu) = F(E, \mu)\, g(E \to E') \ , \tag{246}$$

where F is the probability that a neutron will scatter through a laboratory angle with cosine μ irrespective of final energy E'. It is obtained from MF=4. Similarly, g is the probability that a neutron's energy will change from E to E' irrespective of the scattering angle, and it is given in MF=5. Continuum reactions are easily identified by MT numbers of 6–49 and 91. The more modern ENDF/B-VII evaluations use the better File 6 representation for these continuum reactions. However, fission emission is still handled in this manner. The secondary-energy distributions can be represented in the following ways:

Law	Description
1	Arbitrary tabulated function
5	General evaporation spectrum (Used for delayed neutrons only.)
7	Simple Maxwellian fission spectrum
9	Evaporation spectrum
11	Energy-dependent Watt fission spectrum
12	Energy-dependent spectrum of Madland and Nix

The feed functions for continuum scattering are simply

$$\begin{aligned} \mathcal{F}_{\ell g'}(E') &= \int_{-1}^{+1} P_\ell(\mu)\, f(E, \mu)\, d\mu \\ &\times \int_{g'} g(E, E')\, dE' \ . \end{aligned} \tag{247}$$

M. File 6 Energy-Angle Distributions

If the File 6 data are expressed as a continuous energy-angle distribution (Law 1) in the laboratory system, it is fairly easy to generate the multigroup transfer matrix. As usual for GROUPR, the task is to calculate the "feed function" (the Legendre moments for transferring to all possible secondary-energy groups starting with incident

energy E). These data are expected to be in Law-1 format, where the looping order is E, E', ℓ. The only problems here are handling the new ENDF-6 interpolation laws for two-dimensional tabulations (for example, unit base) and converting tabulated angle functions for $E \rightarrow E'$ into Legendre coefficients (which is done with a Gauss-Legendre quadrature of order 8).

If the File 6 data are given in the energy-angle form of Law 7 (where the looping order is E, μ, E'), GROUPR converts it to the Law 1 form using subroutine `ll2lab`. It does this by stepping through an E' grid that is the union of the E' grids for all the different angles given. At each of these union E' values, it calculates the Legendre coefficients using a Gauss-Legendre quadrature of order 8, and it checks back to see if the preceding point is still needed to represent the distribution to the desired degree of accuracy. Now `getmf6` and `f6lab` can be used to complete the calculation as above.

If the File 6 data are expressed in the CM system, or if the phase-space option is used, more processing is necessary to convert to the LAB system. This conversion is done for each incident energy E given in the file. The grid for laboratory secondary energy E'_L is obtained by doing an adaptive reconstruction of the emission probability $p_{L\ell}(E, E'_L)$ such that each Legendre order can be expressed as a linear-linear function of E'_L. The values for $p_{L\ell}(E, E'_L)$ are obtained in subroutine `f6cm` by doing an adaptive integration along the contour E'_L=const in the E'_C, μ_C plane using μ_L as the variable of integration:

$$p_{L\ell}(E, E'_L) = \int_{\mu_{\min}}^{+1} p_C(E, E'_C, \mu_C)\, P_\ell(\mu_L)\, J\, d\mu_L \ , \quad (248)$$

where μ is a scattering cosine and L and C denote the laboratory and center-of-mass (CM) systems, respectively. The Jacobian of the transformation is given by

$$J = \sqrt{\frac{E'_L}{E'_C}} = \frac{1}{\sqrt{1 + c^2 - 2c\mu_L}} \ , \quad (249)$$

and the cosine transformation is given by

$$\mu_C = J(\mu_L - c) \ . \quad (250)$$

The constant c is given by

$$c = \sqrt{\frac{A'}{(A+1)^2}} \sqrt{\frac{E}{E'_L}} \ , \quad (251)$$

where A is the ratio of the atomic weight of the target to the atomic weight of the projectile, and A' is the ratio of the atomic weight of the emitted particle to the atomic weight of the projectile. The lower limit of the integral depends on the maximum possible value for the center-of-mass (CM) secondary energy as follows:

$$\mu_{\min} = \frac{1}{2c}\Big(1 + c^2 - \frac{E'_{C\max}}{E'_L}\Big), \quad (252)$$

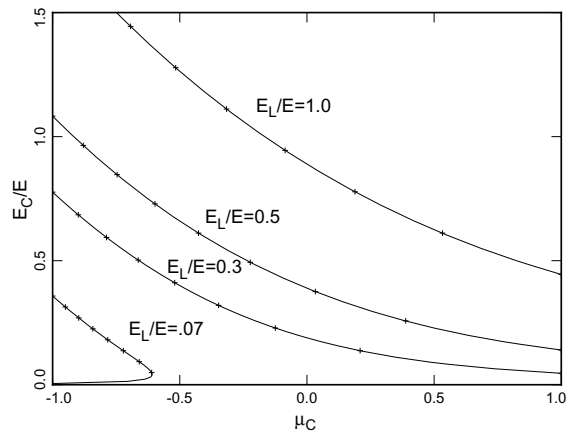

FIG. 46: Coordinate mapping between CM and laboratory reference frames for A=2. The parameters E_L and E_C are the secondary energies in the lab and CM frames. The crosses on the curves are at μ_L values of -1.0, -.75, -.50, . . . , .75, and 1.0. This figure also illustrates the advantage of integrating over μ_L for the contour in the lower-left corner. The values of E_C and μ_C are single-values functions of μ_L.

where

$$E'_{L\max} = E\Big(\sqrt{\frac{E'_{C\max}}{E}} + \sqrt{\frac{A'}{(A+1)^2}}\Big)^2. \quad (253)$$

An example of the integration contours for this coordinate transformation is given in Fig. 46.

The CM energy-angle distribution can be given as a set of Legendre coefficients or a tabulated angular distribution for each possible energy transfer $E \rightarrow E'$, as a "precompound fraction" $r(E, E')$ for use with the Kalbach-Mann [35] or Kalbach [37] angular distributions, or as parameters for a phase-space distribution. The first three options are processed using `f6ddx`, and the last using `f6psp`. As noted previously in the ACER section, but repeated here for clarity, the Kalbach option leads to a very compact representation. Kalbach and Mann examined a large number of experimental angular distributions for neutrons and charged particles. They noticed that each distribution could be divided into two parts: an equilibrium part symmetric in μ, and a forward-peaked preequilibrium part. The relative amount of the two parts depended on a parameter r, the preequilibrium fraction, which varies from zero for low E' to 1.0 for large E'. The shapes of the two parts of the distributions depended most directly on E'. This representation is very useful for preequilibrium statistical-model codes like GNASH [36], because they can compute the parameter r, and all the rest of the angular information comes from simple universal functions. More specifically, Kalbach's latest work says that

$$f(\mu) = \frac{a}{2\sinh(a)}\Big[\cosh(a\mu) + r\sinh(a\mu)\Big], \quad (254)$$

where a is a simple function of E, E', and B_b, the separation energy of the emitted particle from the liquid-drop model without pairing and shell terms.

The File 6 processing methods in GROUPR apply equally well to neutrons, photonuclear photons, and charged particles. The effects on the kinematics due to the difference in mass between the incident particle and the emitted particle are handled by the variable A' in the above equations.

N. GENDF Output

The group constants produced by GROUPR are normally written to an output file in GENDF (groupwise ENDF) format for use by other modules of NJOY. For example, DTFR can be used to convert a GENDF material to DTF (or "transport") format; CCCCR produces the standard interface files [8] ISOTXS, BRKOXS, and DLAYXS; MATXSR produces a file in MATXS format [43]; and POWR produces libraries of the Electric Power Research Institute (EPRI) codes EPRI-CELL and EPRI-CPM. Other formats can easily be produced by new modules, and some functions such as group collapse, are conveniently performed directly in GENDF format. The GENDF format is also used for GAMINR output, and both ACER and ERRORR use GENDF input for some purposes.

Depending on the sign of `ngout2` (see below), the GENDF file will be written in either coded mode (*e.g.*, ASCII) or in the special NJOY blocked-binary mode. Conversion between these modes can be performed subsequently by the MODER module.

The GENDF material begins with a header record (MF=1, MT=451), but the format of this first section is different from MT=451 on an ENDF or PENDF file. The section consists of a "CONT" record, containing

`ZA`	Standard 1000*Z+A value
`AWR`	Atomic weight ratio to neutron
`0`	Zero
`NZ`	Number of σ_0 values
`-1`	Identifies a GENDF-type data file
`NTW`	Number of words in title

and a single "LIST" record, containing

`TEMPIN`	Material temperature (Kelvin)
`0.`	Zero
`NGN`	Number of neutron groups
`NGG`	Number of photon groups
`NW`	Number of words in LIST
`0`	Zero
`TITLE`	Title from GROUPR input (`NTW` words)
`SIGZ`	Sigma-zero values (`NZ` words)
`EGN`	Neutron group boundaries, low to high (`NGN+1` words)
`EGG`	Photon group boundaries, low to high (`NGG+1` words).

For photoatomic GENDF files produced by the GAMINR module, the photon group structure is stored in `ngn` and `egn`, and the number of photon groups is given as `ngg=0`. The word count is seen to be NW=NTW+NZ+NGN+1+NGG+1. The LIST record is followed by a standard ENDF file-end record (FEND). The normal ENDF section-end (SEND) is omitted.

This header is followed by a series of records for reactions. The ENDF ordering requirements are relaxed, and MF and MT values can occur in any order. Each section starts with a "CONT" record.

`ZA`	Standard 1000*Z+A value
`AWR`	Standard atomic weight ratio
`NL`	Number of Legendre components
`NZ`	Number of sigma-zero values
`LRFLAG`	Break-up identifier flag
`NGN`	Number of groups

It is followed by a series of LIST records, one for every incident-energy group with nonzero result,

`TEMP`	Material temperature (Kelvin)
`0.`	Zero
`NG2`	Number of secondary positions
`IG2LO`	Index to lowest nonzero group
`NW`	Number of words in LIST
`IG`	Group index for this record
`A(NW)`	Data for LIST (NW words),

where `NW=NL*NZ*NG2`. The last LIST record in the sequence is the one with `IG=NGN`. It must be given even if its contents are zero. The last LIST record is followed by a SEND record.

The contents of `A(NW)` change for various types of data. For simple cross section "vectors" (MF=3), NG2 is 2, and A contains the two Fortran arrays

`FLUX(NL,NZ), SIGMA(NL,NZ)`

in that order. Following the normal Fortran convention, the first index changes the fastest. For ratio quantities like fission $\overline{\nu}$, `NG2` is 3, and `A` contains

`FLUX(NL,NZ), RATIO(NL,NZ), SIGMA(NL,NZ).`

For transfer matrices (MF=6, 16, 21, *etc.*), `A` contains

`FLUX(NL,NZ), MATRIX(NL,NZ,NG2-1).`

The actual secondary group indices for the last index of `MATRIX` are usually `IG2LO`, `IG2LO+1`, *etc.*, using the GROUPR convention of labeling groups in order of increasing energy. If the low-energy part of the fission matrix (or the fission or capture photon production matrices) uses the special space-saving format described above, the spectrum will be found in a LIST record with `IG=0` and the production cross section will be found in a series of records with `IG2LO=0`. The group range for the spectrum ranges from `IG2LO` to `IG2LO+NG2-1`. For `IG2LO=0`, `NG2` will be 2 as for a normal cross section, and the two values will be the flux for group `IG` and the corresponding production cross section.

Finally, for delayed neutron spectra (MF=5), `NL` is used to index the 6 or 8 time groups, `NZ` is 1, and there

is only one incident energy record (`IG=IGN`). The array `A` contains

`LAMBDA(NL), CHID(NL,NG2-1),`

where `LAMBDA` contains the 6 or 8 delayed-neutron time constants and `CHID` contains the 6 or 8 spectra.

The GENDF material ends with a material-end (MEND) record, and the GENDF tape ends with a tape-end (TEND) record.

O. Running GROUPR

The input instructions for GROUPR are available as comments in the source code, in the NJOY manuals, or on the NJOY web site. Here is the first part of an example of running GROUPR to make a 187-group library for H-1 from ENDF/B-VII. This is a general purpose library with both fast and thermal groups.

```
 1. groupr
 2. 20 22 0 23
 3. 125 10 7 7 5 8 1 1
 4. 'h1 endf/b-vii 187x24'/
 5. 293.6 400 500 600 800 1000 1200 1600/
 6. 1e10/
 7. 3/
 8. 3 221 'free'/
 9. 3 222 'h(h2o) thermal scattering'/
10. 3 223 'h(ch2) inelastic thermal scattering'/
11. 3 224 'h(ch2) elastic thermal scattering'/
12. 3 225 'h(zrh) inelastic thermal scattering'/
13. 3 226 'h(zrh) elastic thermal scattering'/
14. 3 227 'h(c6h6) thermal scattering'/
15. 3 251 'mubar'/
16. 3 252 'xi'/
17. 3 253 'gamma'/
18. 3 259 '1/v'/
19. 6/
20. 6 221 'free'/
21. 6 222 'h(h2o) thermal scattering'/
22. 6 223 'h(ch2) inelastic thermal scattering'/
23. 6 224 'h(ch2) elastic thermal scattering'/
24. 6 225 'h(zrh) inelastic thermal scattering'/
25. 6 226 'h(zrh) elastic thermal scattering'/
26. 6 227 'h(c6h6) thermal scattering'/
27. 16/
28. 0/
```

On Card 3, we see the MAT number of H-1 (125). This is followed by options for the neutron group structure (10), the photon group structure (7), the weighting function (7), and the Legendre order (5). There are 8 temperatures and 1 σ_0 value. The print option is set to 1 for a maximum listing. Card 4 provides a comment line for the run. Cards 5 and 6 give the lists of temperatures and σ_0 values. Only infinitely dilute data are needed for H-1. With Card 7, the selection of reactions to be calculated begins.

Reactions are requested by giving `MFD` (PENDF file desired), `MTD` (PENDF section desired), and `mtname`. As will be described below, there are options to select reactions explicitly, or to automatically select all reactions of a given type. The entry on Card 7 asks for all the reactions in File 3 of the PENDF file. The thermal scattering reactions have to be selected explicitly, and you see their cross sections being requested in Cards 8 through 14. Other explicit selections are given in Cards 15 through 18. The next card asks for all reactions from File 6 on the PENDF file, but once again, thermal reactions have to requested explicitly. Card 27 asks for all the photon production reactions from combinations of File 12, 13, and 16 on the PENDF file, and Card 28 terminates the input for the first temperature. We will return to complete the input for this example below

`ign`	meaning
1	arbitrary structure (read in)
2	csewg 239-group structure
3	lanl 30-group structure
4	anl 27-group structure
5	rrd 50-group structure
6	gam-i 68-group structure
7	gam-ii 100-group structure
8	laser-thermos 35-group structure
9	epri-cpm 69-group structure
10	lanl 187-group structure
11	lanl 70-group structure
12	sand-ii 620-group structure
13	lanl 80-group structure
14	eurlib 100-group structure
15	sand-iia 640-group structure
16	vitamin-e 174-group structure
17	vitamin-j 175-group structure
18	xmas nea-lanl
19	ecco 33-group structure
20	ecco 1968-group structure
21	tripoli 315-group structure
22	xmas lwpc 172-group structure
23	vit-j lwpc 175-group structure

FIG. 47: Standard neutron group structures.

The names of the available group structures are given in the input instructions, and the energy bounds are tabulated in data blocks in the code. The options currently available are given in Figure 47. Note that the example being presented here uses the LANL 187-group structure. Of course, it is always possible to read in an arbitrary group structure. The energies must be given in increasing order (note that this is opposite from the usual convention). Here is an example of the input cards for the conventional 4-group structure used in older thermal reactor codes:

```
4/ card6a
1e-5 .625 5530 .821e6 10e6 / card6b  .
```

These cards are read by the standard Fortran READ* method. Fields are delimited by space, and "/" terminates the processing of input on a card. Anything after the slash is a comment.

The standard photon group structures are listed in Figure 48. Note that the example being presented here uses the LANL 24-group structure. Other group bounds can be read in as described above.

igg	meaning
0	none
1	arbitrary structure (read in)
2	csewg 94-group structure
3	lanl 12-group structure
4	steiner 21-group gamma-ray structure
5	straker 22-group structure
6	lanl 48-group structure
7	lanl 24-group structure
8	vitamin-c 36-group structure
9	vitamin-e 38-group structure
10	vitamin-j 42-group structure

FIG. 48: Standard photon group structures.

iwt	meaning
1	read in smooth weight function
2	constant
3	1/e
4	1/e + fission spectrum + thermal maxwellian
5	epri-cell lwr
6	(thermal) – (1/e) – (fission + fusion)
7	same with t-dep thermal part
8	thermal–1/e–fast reactor–fission + fusion
9	claw weight function
10	claw with t-dependent thermal part
11	vitamin-e weight function (ornl-5505)
12	vit-e with t-dep thermal part
-n	compute flux with weight n
0	read in resonance flux from ninwt

FIG. 49: Standard weight function options.

The available weight function options are listed in the input instructions under `iwt` and summarized in Figure 49.

Note that the example being presented here uses the "(thermal)–(1/E)–(fission+fusion)" option with a temperature dependent thermal part. Just as in the case of group structures, an arbitrary weight function can be read in. This function is presented to GROUPR as an ENDF/B "TAB1" record. This means that a count of $E, C(E)$ pairs and one or more interpolation schemes are given. For the case of a single interpolation scheme, the input cards would be

```
0. 0. 0 0 1 N N INT
E1 C(E1) E2 C(E2) ... / card8b,
```

where `N` is the count of E, C pairs, and `INT` specifies the interpolation law to be used for values of E between the grid values E_1, E_2,... according to the following "graph paper" schemes:

INT	Meaning
1	histogram
2	linear-linear
3	linear-log
4	log-linear
5	log-log

For example, INT=5 specifies that $\ln C$ is a linear function of $\ln E$ in each panel $E_i \leq E < E_j$. Similarly, INT=4 specifies that $\ln C$ is a linear function of E in each panel. As many physical lines as are needed can be used for "`card8b`", as long as the terminating slash is included.

One of the weighting options is a generalized "$1/E$+fission+thermal" function where the thermal temperature, fission temperature, and breakpoint energies (all in eV) are given on `card8c`. As an example,

```
.10 .025 820.3e3 1.4e6 / card8c   .
```

The GROUPR flux calculator is selected by a negative sign on `iwt`. An additional card is then read. The calculator option used is determined by the number of parameters given and their values. The parameters `ehi` and `nflmax` are used to select the energy range for the flux calculation, and they also determine the cost in time and storage. The actual value for `sigpot` is not very critical—a number near 10 barns is typical for fissionable materials.

Nonzero values for `ninwt` and `jsigz` will cause the computed flux for a given fissionable isotope (such as ^{238}U) to be written out onto a file. This saved flux can be used as input for a subsequent run for a fissile material (such as ^{239}Pu) with `iwt`=0 to get an approximate correction for resonance-resonance interference. See Eq. (213).

Nonzero values for some of the last five parameters on `card8a` select the extended flux calculation of Eq. (211). The simplest such calculation is for an isolated pin containing a heavy absorber with an admixed moderator. For $^{238}UO_2$, the card might be

```
400 10.6 5000 0 0 .7768 7.5 / card8a  ,
```

where 7.5 barns is twice the oxygen cross section and α is computed from $\left[(A-1)/(A+1)\right]^2$ with $A = 15.858$. A more general case would be a PWR-like lattice of $^{238}UO_2$ fuel rods in water:

```
400 10.6 5000 0 0 .7768 7.5 .40
   1.7e-7 .086 / card8a  ,
```

where .086 is computed using 3.75 barns for O and 40 barns for the two H atoms bound in water; that is,

$$\gamma = \frac{3.75}{2 * 20 + 3.75} = .086 \,. \tag{255}$$

A third example would compute the flux for a homogeneous mixture of ^{238}U and hydrogen

```
400 10.6 5000 0 0 0 0 1. 1.7e-7 / card8a  .
```

As a final example, consider a homogeneous mixture of uranium and water. This requires `beta=1` and `sam=0`. Thus,

```
400 10.6 5000 0 0 .7768 0. 1.
  1.7e-7 .086 / card8a  .
```

We now return to the H-1 example. Following the reaction requests for the first temperature, we provide additional requests for the higher temperatures terminating each set with a "/" symbol. For 400K, we give

```
29. 3   1 'total'/
30. 3   2 'elastic'/
31. 3 102 'capture'/
32. 3 221 'free'/
33. 3 222 'h(h2o) thermal scattering'/
34. 3 225 'h(zrh) inelastic thermal scattering'/
35. 3 226 'h(zrh) elastic thermal scattering'/
36. 3 227 'h(c6h6) thermal scattering'/
37. 3 301 'heating'/
38. 3 444 'damage'/
39. 6   2 'elastic'/
40. 6 221 'free'/
41. 6 222 'h(h2o) thermal scattering'/
42. 6 225 'h(zrh) inelastic thermal scattering'/
43. 6 226 'h(zrh) elastic thermal scattering'/
44. 6 227 'h(c6h6) thermal scattering'/
45. 0/
```

We don't use the automatic selection for File 3 because not all the reactions have reasonable temperature dependence. In this case, the total, elastic, capture, heating, and damage reactions can depend on temperature. In addition, the elastic matrix and all the thermal cross sections and scattering matrices will depend on temperature. A tricky part shows up now. Some of the temperatures requested are above the maximum temperature provided in the thermal data. In these cases, we just list the remaining reactions. The final temperature for our example input file looks like this:

```
 3   1 'total'/
 3   2 'elastic'/
 3 102 'capture'/
 3 221 'free'/
 3 301 'heating'/
 3 444 'damage'/
 6   2 'elastic'/
 6 221 'free'/
0/
0/
```

where only the free gas scattering (221) of the thermal reactions remains.

We will now discuss reaction selection in more detail. Because of the many types of data that it can process, GROUPR does not have a completely automatic mode for choosing reactions to be processed. On the basic level, it asks the user to request each separate cross section or group-to-group matrix using the parameters `mfd`, `mtd`, and `mtname`. However, simplified input modes are also available as shown above.

For completeness, the full input for `matd`, `mfd`, and `mtname` will be described first. Most readers can skip to the description of automated processing below. The value of `mfd` depends on the output desired (vector, matrix) and the form of the data on the ENDF evaluation. Simple cross section "vectors" σ_{xg} are requested using `mfd`=3 and the `mtd` numbers desired from the list of reactions available in the evaluation (check the directory in MF=1,MT=451 of the ENDF and PENDF files for the reactions available). A typical example would be

```
3   1 'total'/
3   2 'elastic'/
3  16 '(n,2n)'/
3  51 '(n,nprime)first'/
3 -66 '(n,nprime)next'/
3  91 '(n,nprime)continuum'/
3 102 'radiative capture'/.
```

The combinations of "3 51" followed by "3 −66" means process all the reactions from 51 through 66; that is, (n,n$'_1$), (n,n$'_2$), ..., (n,n$'_{16}$). If self-shielding is requested, the following reactions will be processed using `nsigz` values of background cross section: total (MT=1), elastic (MT=2), fission (MT=18 and 19), radiative capture (MT=102), heat production (MT=301), kinematic KERMA (MT=443), and damage energy production (MT=444). The other File-3 reactions will be computed at $\sigma_0=\infty$ only. This list of reactions can be altered by small changes in `init` if desired.

There are several special options for `mtd` available when processing cross section vectors:

`mtd`	Option
259	Average inverse neutron velocity for group in s/m.
258	Average lethargy for group.
251	Average elastic scattering cosine $\overline{\mu}$ computed from File 4.
252	Continuous-slowing-down parameter $\overline{\xi}$ (average logarithmic energy decrement for elastic scattering) computed from File 4.
253	Continuous-slowing-down parameter $\overline{\gamma}$ (the average of the square of the logarithmic energy decrement for elastic scattering, divided by twice the average logarithmic energy decrement for elastic scattering) computed from File 4.
452	$\overline{\nu}$: the average total fission yield computed from MF=1 and MF=3.
455	$\overline{\nu}^D$: the average delayed neutron yield computed from MF=1 and MF=3.
456	$\overline{\nu}^P$: the average prompt fission neutron yield computed from MF=1 and MF=3.

There are also some special options for `mfd` that can be used when processing cross sections:

`mfd`	Option
12	Photon production cross section computed from File 12 and File 3.
13	Photon production cross section computed from File 13. Recent versions of GROUPR will automatically shift between 12 and 13, if necessary.

1zzzaaam	nuclide production for zzzaaam from a subsection of MF=3
2zzzaaam	nuclide production for zzzaaam from a subsection of MF=6
3zzzaaam	nuclide production for zzzaaam from a subsection of MF=9
3zzzaaam	nuclide production for zzzaaam from a subsection of MF=10

An example of the isomer production capability would be the radiative capture reaction of ENDF/B-V ^{109}Ag(n,γ) from Tape 532:

```
30471090 102 '(n,g) to g.s.'/
30471091 102 '(n,g) to isomer'/  .
```

The next class of reactions usually processed is the group-to-group neutron scattering matrices. The complete list of `mtd` values is most easily found under File 4 in the MF=1,MT=451 "dictionary" section of the evaluation. An example follows:

```
6   2 'elastic matrix'/
6  16 '(n,2n) matrix'/
6  51 '(n,nprime)first matrix'/
6 -66 '(n,nprime)next matrix'/
6  91 '(n,nprime)continuum matrix'/  .
```

Using `mfd=6` implies that File 4, or File 4 and File 5, will be used to generate the group-to-group matrix. The elastic matrix will be computed for `nsigz` values of background cross section, but the other reactions will be computed for $\sigma_0=\infty$ only. The list of matrices to be self-shielded can be changed by making small changes to `init` if desired.

The case of fission is more complex. For the minor isotopes, only the total fission reaction is used, and the following input is appropriate for the prompt component:

```
3 18 'fission xsec'/
6 18 'prompt fission matrix'/  .
```

For the important isotopes, partial fission reactions are given. They are really not needed for most fission reactor problems, and the input above is adequate. However, for problems where high-energy neutrons are important, the following input should be used:

```
3 18 'total fission'/
3 19 '(n,f)'/
3 20 '(n,nf)'/
3 21 '(n,2nf)'/
3 38 '(n,3nf)'/
6 19 '(n,f)'/
6 20 '(n,nf)'/
6 21 '(n,2nf)'/
6 38 '(n,3nf)'/  .
```

Note that "6 18" is omitted because it will, in general, be different from the sum of the partial matrices. Some materials don't have data for (n,3nf); in these cases, omit the two lines with MTD=38 from the input. The fission matrix is not self-shielded. Since resonance-to-resonance fission-spectrum variations are not described in the ENDF format, it is sufficient to self-shield the cross section and then to use the self-shielding factor for the cross section to self-shield the fission neutron production.

Delayed fission data are available for the important actinide isotopes, and the following input to GROUPR is used to process them:

```
3 455 'delayed nubar'/
5 455 'delayed spectra'/  .
```

The line for MFD=5 automatically requests spectra for all time groups of delayed neutrons. The time constants are also extracted from the evaluation. Formatting modules such as DTFR and CCCCR must combine the prompt and delayed fission data written onto the GENDF tape in order to obtain steady-state fission parameters for use in transport codes.

Starting with the ENDF-6 format, neutron production data may also be found in File 6, and `mfd=8` is used to tell the code to use MF6 for this `mtd`. When using full input, the user will have to check the File 1 directory and determine what subsections occur in File 6.

Photon production reactions can be found in the ENDF dictionary under MF=12 and 13. To request a neutron-to-photon matrix, add 4 to this number. (In recent versions of NJOY, GROUPR will automatically shift between 16 and 17 using data read from the ENDF dictionary by the `conver` subroutine. Thus, use of `mfd`=17 is no longer necessary.) For example,

```
17   3 'nonelastic photons'/
16   4 'inelastic photons'/
16  18 'fission photons'/
16 102 'capture photons'/  .
```

Yields (MF=12) are normally used with resonance reactions (MT=18 or MT=102), or for low-lying inelastic levels (MT=51, 52, ...). MT=3 is sometimes used by evaluators as a catch-all reaction at high energies where it is difficult to separate the source reactions in total photon emission measurements. In these cases, photon production cross sections from other reactions like MT=102 are normally set equal to zero at high energies. The general rule for photon emission is that the total production is equal to the sum of all the partial production reactions given in the evaluation. Starting with the ENDF-6 format, photon production may also appear in File 6. Use `mfd=18` to process these contributions. Since resonance-to-resonance variations in photon spectra are not given in ENDF evaluations, GROUPR does not normally self-shield the photon production matrices (although this can be done if desired by making a small change in `init`); instead, it is assumed that only the corresponding cross section needs to be shielded. Subsequent codes can use the cross section self-shielding factor with the infinite-dilution photon production matrix to obtain self-shielded photon production numbers.

This version of GROUPR can also generate group-to-group matrices for charged-particle production from neutron reactions and for all kinds of matrices for incident charged particles. The incident particle is determined by the input tape mounted. The identity of the secondary

particle is chosen by using one of the following special `mfd` values:

For distributions given in File 6 (energy-angle):

`mfd`	Meaning
21	proton production
22	deuteron production
23	triton production
24	He-3 production
25	alpha production
26	residual nucleus (A>4) production

For distributions given in File 4 (angle only):

`mfd`	Meaning
31	proton production
32	deuteron production
33	triton production
34	He-3 production
35	alpha production
36	residual nucleus (A>4) production

If necessary, `mfd`=21–26 will automatically change to 31–36.

The user will normally process all reactions of interest at the first temperature (for example, 293.6 K). At higher temperatures, the threshold reactions should be omitted, because their cross sections do not change significantly with temperature except at the most extreme conditions. This means that only the following reactions should be included for the higher temperatures (if present): total (MT=1), elastic (MT=2), fission (MT=18), radiative capture (MT=102), heating (MT=301), kinematic KERMA (MT=443), damage (MT=444), and any thermal cross sections (MT=221–250). Only the elastic and thermal matrices should be included at the higher temperatures.

Warning: when using the explicit-input option, it is a fatal error to request a reaction that does not appear in the evaluation, cannot be computed from the evaluation, or was not added to the PENDF file by a previous module. Reactions with thresholds above the upper boundary of the highest energy group will be skipped after printing a message on the output file.

Automated processing of essentially all reactions included in an ENDF/B evaluation is also available. As mentioned above, the single card

```
3/
```

will process all the reactions found in File 3 of the input PENDF file. However, this list excludes thermal data (MT=221–250) and special options such as MTD=251–253, 258–259, and 452–456. If any of these reactions are needed, they should be given explicitly (see example below). Similarly, the single card

```
6/
```

will process the group-to-group matrices for all reactions appearing in File 4 of the ENDF/B file, except for MT=103–107 and thermal scattering matrices (MT=221–250). If MT=18 and 19 are both present, only MT=19 will be processed into a fission matrix. For ENDF-6 evaluations, the "6/" option will also process every neutron-producing subsection in File 6. Photon production cross sections are requested using

```
13/
```

and photon-production matrices are requested with the single card

```
16/   .
```

In both cases, all reactions in both File 12 and File 13 will be processed without the need for using MFD=12 or MFD=17. For ENDF-6 libraries, this option will also process all photon-production subsections in File 6. There is no automatic option for delayed neutron data. An example of a processing run for a fissionable isotope with thermal cross sections follows:

```
3/
3 221/ thermal xsec
3 229/ av.inverse velocity
3 455/ delayed nubar
5 455/ delayed spectra
6/
6 221/ thermal matrix
16/ photon production matrix   .
```

An example of charged-particle processing for the incident-neutron part of a coupled n-p-γ library follows:

```
3/ cross sections
6/ neutron production matrix
16/ photon production matrix
21/ proton production matrix   .
```

The layout of data in a n-p-γ coupled set is shown in Figure 50.

There is a new ENDF format now becoming available that helps to describe the production of all isotopes and isomers in a given system. It uses a directory in File 8 to direct the code to productions represented by MF=3, by MF=9 times MF=3, by MF=10, or by sections of MF=6 times MF=3. When this format is used, it is possible to give the simple command

```
10/
```

to have production cross sections generated for every nuclide that is produced. These production sections are labeled with the ZA and isomeric state of the products for use by subsequent NJOY modules.

X. GAMINR

The GAMINR module of NJOY is designed to produce complete and accurate multigroup photoatomic interaction cross sections from ENDF data. Total, coherent, incoherent, pair production, and photoelectric cross sections can be averaged using a variety of group structures

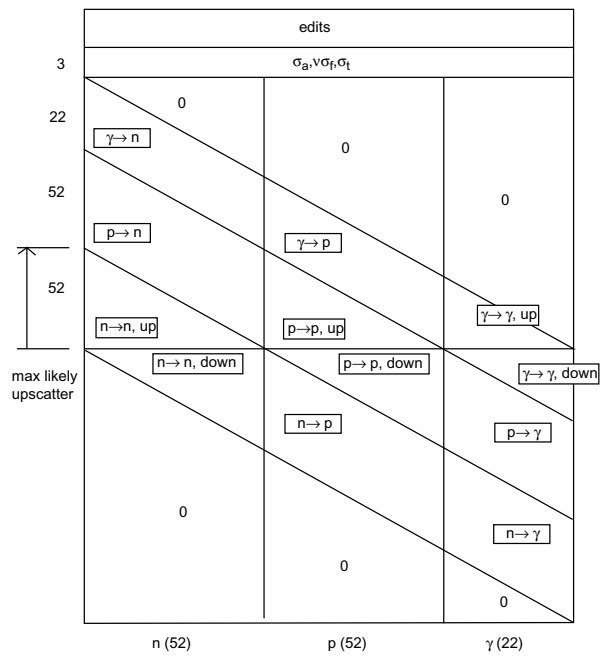

FIG. 50: Layout of a coupled table for the simultaneous transport of neutrons, protons, and gamma rays. Normally, only the lower triangle of the "p to n" block would contain values for the upscatter portion of the table (the part above the line in the middle). The "group" index increases from left to right, and the "position" index increases from top to bottom.

and weighting functions. The Legendre components of the group-to-group coherent and incoherent scattering cross sections are calculated using the form factors [46] now available in ENDF/B. These form factors account for the binding of the electron in its atom. Consequently, the cross sections are accurate for energies as low as 1 keV. GAMINR also computes partial heating cross sections or kinetic energy release in materials (KERMA) factors for each reaction and sums them to obtain the total heating factor. The resulting multigroup constants are written on an intermediate interface file for later conversion to any desired format. Photonuclear reactions such as (γ,n) are not computed by this module.

A. Description of the Photon Interaction Files

In the ENDF photon interaction files, the coherent scattering of photons by electrons is represented by

$$\sigma_C(E,E',\mu)\,dE'd\mu = \frac{3\sigma_T}{8}(1+\mu^2)\,|F(q,Z)|^2\delta(E-E')\,dE'd\mu \ , \quad (256)$$

where E is the energy of the incident photon, E' is the energy of the scattered photon, μ is the scattering cosine, σ_T is the classical Thomson cross section (0.66524486 b), Z is the atomic number of the scattering atom, and F is the atomic form factor. The coherent form factor is a function of momentum transfer q given by

$$q = 2k\sqrt{\frac{1-\mu}{2}} \ , \quad (257)$$

where k is the energy in free-electron units ($k = E/511003.4$ eV), but F is actually tabulated versus the quantity $x = 20.60744q$. The coherent form factor is given as MT=502 in File 27.

Incoherent scattering is represented by the expression

$$\sigma_I(E,E',\mu)\,dE'd\mu = S(q,Z)\,\sigma_{KN}(E,E',\mu)\,dE'd\mu \ , \quad (258)$$

where S is the incoherent scattering function and σ_{KN} is the free-electron Klein-Nishina cross section

$$\sigma_{KN}(E,E',\mu) = \frac{3\sigma_T}{8k^2}\left[\frac{k}{k'}+\frac{k'}{k}+2\left(\frac{1}{k}-\frac{1}{k'}\right)+\left(\frac{1}{k}-\frac{1}{k'}\right)^2\right] \quad (259)$$

The scattering angle and momentum transfer for incoherent scattering are given by

$$\mu = 1+\frac{1}{k}-\frac{1}{k'} \ , \quad (260)$$

and

$$q = 2k\sqrt{\frac{1-\mu}{2}}\,\frac{\sqrt{1+(k^2+2k)(1-\mu)/2}}{1+k(1-\mu)} \ . \quad (261)$$

As was the case for coherent scattering, $S(q,Z)$ is actually tabulated *vs.* $x = 20.60744q$. It is important to note that S is essentially equal to Z for x greater than Z. The incoherent scattering function is given as MT=504 in File 27.

The ENDF photon interaction files also contain tabulated cross sections for total, coherent, incoherent, pair production, and photoelectric reactions. They are given in File 23 as MT=501, 502, 504, 516, and 602 respectively (in ENDF/B-VI files, 602 is changed to 522). The coherent and incoherent cross sections were obtained by the evaluator by integrating Eqs.(256) and (258) and are therefore redundant. Due care must be taken to avoid introducing inconsistencies.

B. Calculational Method

The multigroup cross sections produced by GAMINR are defined as follows:

$$\sigma_{xg} = \frac{\int_g \sigma_x \, \phi_0(E) \, dE}{\int_g \phi_0(E) \, dE} , \quad (262)$$

$$\sigma_{T\ell g} = \frac{\int_g \sigma_T(E) \, \phi_\ell(E) \, dE}{\int_g \phi_\ell(E) \, dE} , \text{ and} \quad (263)$$

$$\sigma_{x\ell g \to g'} = \frac{\int_g \mathcal{F}_{x\ell g'}(E) \, \sigma_x(E) \, \phi_\ell(E) \, dE}{\int_g \phi_\ell(E) \, dE} . \quad (264)$$

In these expressions, g represents an energy group for the initial energy E, g' is a group of final energies E', x stands for one of the reaction types, T denotes the total, and ϕ_ℓ is a Legendre component of a guess for the photon flux. In the last equation, $\mathcal{F}$ is the "feed function"; that is, the total normalized probability of scattering from E into group g'. The feed function for coherent scattering is

$$\mathcal{F}_{C\ell g'}(E) = \frac{\int_{-1}^{+1} \sigma_C(E, E', \mu) \, P_\ell(\mu) \, d\mu}{\sigma_C(E)} = \frac{\int_{-1}^{+1} (1-\mu^2) \, |F(x)|^2 P_\ell(\mu) \, d\mu}{\int_{-1}^{+1} (1-\mu^2) \, |F(x)|^2 d\mu} , \quad (265)$$

for E in g' and zero elsewhere.

Here $P_\ell(\mu)$ is a Legendre polynomial. Note that $\mathcal{F}_{C0g'} = 1$; this form assures that the coherent scattering cross sections are consistent with the values tabulated in File 23.

For incoherent scattering,

$$\mathcal{F}_{I\ell g'} = \frac{\int_{g'} S(q, Z) \, \sigma_{KN}(E, E', \mu) \, P_\ell(\mu) \, dE'}{\int_{g'} S(q, Z) \, \sigma_{KN}(E, E', \mu) \, dE'} . \quad (266)$$

Because S is simply equal to Z for high values of q, it is not necessary to completely recompute the incoherent matrix when processing a series of elements in the order of increasing Z. GAMINR automatically determines that some groups for element Z_{now} can be obtained from element Z_{last} by scaling by Z_{now}/Z_{last}.

Finally, pair production is represented as a $(\gamma, 2\gamma)$ scattering event, where

$$\mathcal{F}_{PP\ell g'}(E) = \begin{cases} 2, & \text{if } g' \text{ includes } 511003.4 \text{ eV}; \\ 0, & \text{otherwise} . \end{cases} \quad (267)$$

In addition, GAMINR computes multigroup heating cross sections or KERMA factors as follows:

$$\sigma_{Hxg} = \frac{\int_g [E - \overline{E}_x(E)] \, \sigma_x(E) \, \phi_0(E) \, dE}{\int_g \phi_o(E) \, dE} , \quad (268)$$

where $\overline{E}_x(E)$ is the average energy of photons scattered from E by reaction type x. The average energies are computed using

$$\overline{E}_C = E , \quad (269)$$

$$\overline{E}_{PP} = 1.022007 \, \text{MeV} , \quad (270)$$

$$\overline{E}_{PE} = 0, \text{ and} \quad (271)$$

$$\overline{E}_I(E) = \int_0^\infty E' \, \sigma_I(E, E', \mu) \, dE' / \, \sigma_I(E) . \quad (272)$$

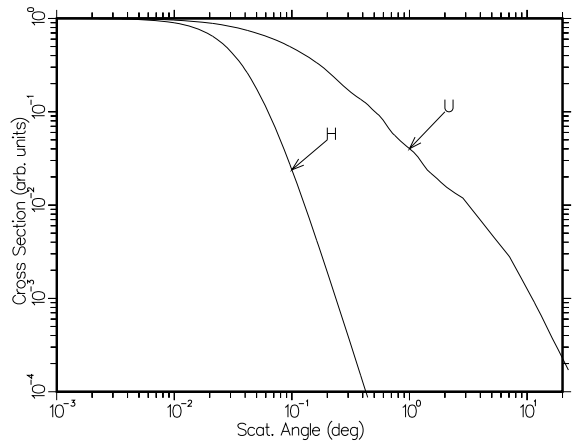

FIG. 51: Angular distribution for coherent scattering from hydrogen and uranium at high photon energy.

These separate contributions to the heating cross section are summed to get a quantity that can be combined with a calculated flux to obtain the total heating rate.

C. Integrals Involving Form Factors

The integrals of Eqs.(265) and (266) that involve the form factors are very difficult to perform because of the extreme forward peaking of the scattering at high energies. Figure 51 illustrates the problem for coherent scattering.

For coherent scattering, the integral of Eq.(265) is broken up into panels by the tabulation values of x. Each panel is integrated in the x domain using Lobatto quadrature of order 6 for $\ell = 3$ or less and order 10 for larger Legendre orders. Equation (257) is used to compute the μ value for each x and to obtain the Jacobian required.

Since μ is quadratic in x, the polynomial order of the integrand in the numerator of Eq.(265) is $2\ell + 2$ plus twice the polynomial order of F in the panel. For $\ell = 3$, the theory of Gaussian quadrature implies that the integral will be exact if F can be represented as a quadratic function over the panel.

The incoherent integral of Eq.(266) is also broken up into panels, but here the panels are defined by the union of the tabulation values of x and the momenta corresponding to the secondary-energy group bounds. The relationship between x, μ, and secondary energy is given

in Eqs.(260) and (261). This time, the integral is performed *vs* E' using Lobatto quadrature of order 6 for ℓ less than or equal to 5 and order 10 for larger ℓ values.

All form factors and structure factors are interpolated using ENDF/B log-log interpolation as specified by the format. However, the cross sections in the file were evaluated using a special log-log-quadratic scheme. Ignoring this complication may lead to a 5% error in the incoherent cross section at 0.1 keV with a negligible error at the higher energies that are of most practical concern [46].

D. Running GAMINR

The user input for GAMINR is described by the comment cards in the source code, the NJOY manuals, and the NJOY web pages. The following sample run prepares a GENDF file for two elements. The numbers on the left are for reference; they are not part of the input.

```
 1.   reconr
 2.   20 21
 3.   'pendf for 2 elements from ENDF/B-VII'/
 4.   100 1 0
 5.   .001 0./
 6.   '1-hydrogen'/
 7.   9200 1 0
 8.   .001 0./
 9.   '92-uranium'/
10.   0/
11.   gaminr
12.   20 21 0 22
13.   100 7 3 4 0
14.   '24-group photon interaction library'/
15.   -1/
16.   9200
17.   -1/
18.   0/
19.   stop
```

Note that both an ENDF/B-VII file and a PENDF file from RECONR are required. Material numbers are equal to $100{*}Z$ for recent versions of the Photon Interaction files. The first number on Card 13 is the MAT value for hydrogen, and this is followed by the photon group structure (7), the weighting function (3), and the Legendre order (4). Card 14 is a comment string. Card 15 is an automatic reaction request. The normal mode of operation uses `mfd=-1`, which automatically processes MT=501, 502, 504, 516, 602, and 621. For ENDF/B-VI and VII, the photoelectric cross section is changed from 602 to 522, and the heating cross section is changed from 621 to 525.

Card 16 starts a new material—in this case, it is uranium. Then Card 17 requests the automatic reaction processing for that material. The final "0/" terminates the loop over materials. It is recommended that the material numbers for the loop be given in increasing order for maximum efficiency.

The following list of photon group structures is copied from the GAMINR input instructions:

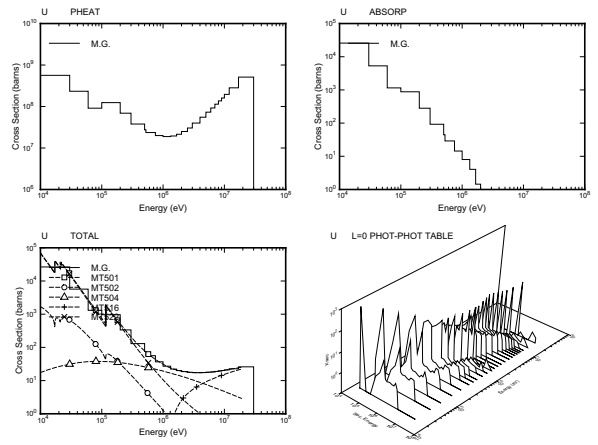

FIG. 52: Plots of the photon interaction cross sections and the photon scattering distribution for uranium showing both 24-group and continuous cross sections. Note the prominent photoelectric absorption edge near 100 keV.

```
igg     meaning
---     -------
 0      none
 1      arbitrary structure (read in)
 2      csewg 94-group structure
 3      lanl 12-group structure
 4      steiner 21-group gamma-ray structure
 5      straker 22-group structure
 6      lanl 48-group structure
 7      lanl 24-group structure
 8      vitamin-c 36-group structure
 9      vitamin-e 38-group structure
10      vitamin-j 42-group structure
```

Similarly, here is a list of the photon weighting functions available for GAMINR:

```
iwt     meaning
---     -------
 1      read in
 2      constant
 3      1/e + rolloffs
```

Option `iwt=3` is used in the example and is recommended for general use. Figures 52 and 53 give plots of the results of this sample run. These graphs were made using the DTFR and VIEWR modules.

Starting with ENDF/B-VI, the photon interaction (or photoatomic) files contain detailed photoelectric cross sections, not just the MT=522 total photoelectric cross section. These photoelectric cross sections have MT numbers starting with 534. As an example, Fig.54 shows the first 9 partial cross sections for uranium—the K, L, and M subshells—taken from ENDF/B-VII. GAMINR input is somewhat more complicated when these reactions are included because they are different for every element.

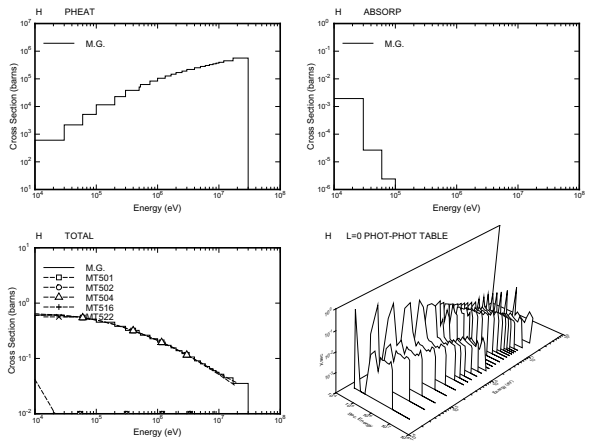

FIG. 53: Plots of the photon interaction cross sections and the photon scattering distribution for hydrogen showing both 24-group and continuous cross sections. The cross sections are simpler for this case.

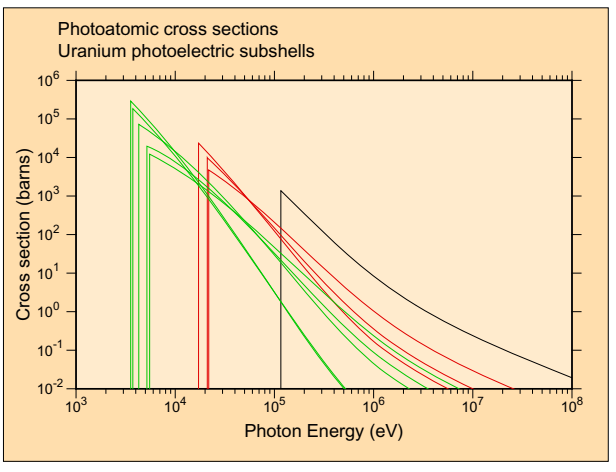

FIG. 54: The first nine photoelectric subshell cross sections for uranium from ENDF/B-VII. Black is the S subshell ($1s^{1/2}$), red is the L1, L2, and L3 subshells ($2s^{1/2}$, $2p^{1/2}$, $2p^{3/2}$), and green is the M1 through M5 subshells ($3s^{1/2}$, $3p^{1/2}$, $3p^{3/2}$, $3d^{3/2}$, $3d^{5/2}$).

XI. ERRORR

The ERRORR module is used to produce cross section and distribution covariances from covariance files in ENDF format. After evaluators have completed their review of the available measurements of various nuclear data (having true values $\sigma_1, \sigma_2, \sigma_3, \cdots$) and the theoretical analysis, they will have formed at least a subjective opinion of the joint probability distribution of the data examined; that is, the probability

$$P(\sigma_1, \sigma_2, \cdots)\, d\sigma_1\, d\sigma_2 \cdots$$

that the true value of σ_1 lies in the range $(\sigma_1, \sigma_1 + d\sigma_1)$, and that σ_2 lies in the range $(\sigma_2, \sigma_2 + d\sigma_2)$, *etc.* In the

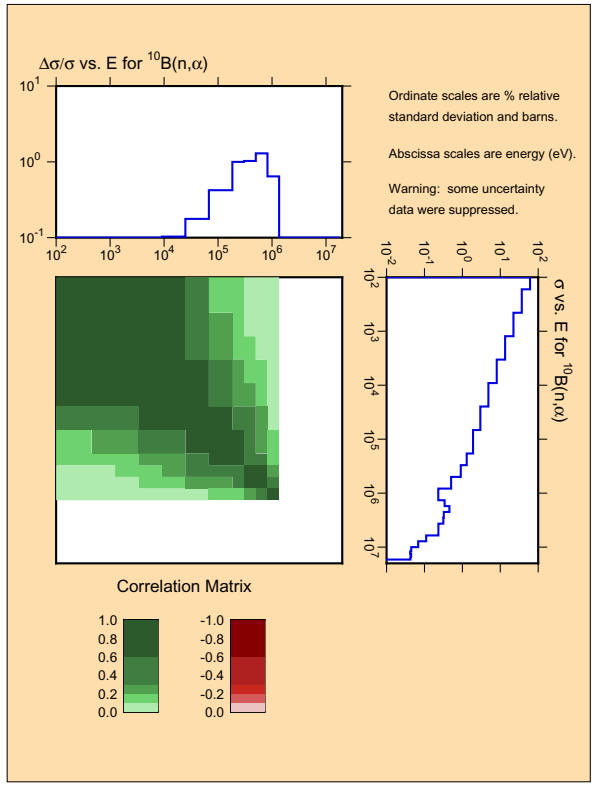

FIG. 55: Covariance plot for ^{10}B(n,α) from ENDF/B-VII. This reaction is used as a standard in the ENDF system.

early versions of the ENDF format, only the first moments (expectation values) of this probability distribution could be included in the numerical data files. However, beginning with ENDF/B-IV and expanding significantly in ENDF/B-V, ENDF/B-VI, and ENDF/B-VII, the second moments of the data probability distributions have been included in many of the files. These second moments (or "data covariances") contain information on the uncertainty of individual data, as well as correlations that may exist. Fig. 55 shows an example of this for ^{10}B from ENDF/B-VII. The top plot shows the first moment (the percent standard deviation) of the uncertainty in the (n,α) cross section. The right-hand plot shows the cross section. The center plot shows the correlations between the (n,α) cross section at one energy to itself at other energies.

Data covariances have many applications. For example, they can be combined with sensitivity coefficients to obtain the uncertainty, due to the data, in calculated quantities of applied interest [47]. This information can be used in turn to judge the adequacy of the data for that application.

The availability of data covariances also makes it possible to use the generalized method of least squares to improve the data evaluation after new integral or differential measurements have been performed [48]. The least-squares method requires only data covari-

ances (not the full probability distribution), and the improved, or adjusted, data are guaranteed [49] to have the smallest possible uncertainties, regardless of the actual shape of the underlying probability distribution function, $P(\sigma_1, \sigma_2, \cdots)$. Thus, the ENDF-formatted covariance files contain, in about as compact a form as possible, a statement about the quality of the data, as well as sufficient information (in principle) to carry out future improvements on an objective basis.

In many of these applications, it is necessary to begin by converting energy-dependent covariance information in ENDF format [2] into multigroup form. This task can be performed conveniently in the NJOY environment using the ERRORR module. In particular, ERRORR calculates the uncertainty in infinitely dilute multigroup cross sections (or multigroup $\bar{\nu}$ values), as well as the associated correlation coefficients. These data are obtained by combining absolute or relative covariances from ENDF Files 31, 32, 33, and 40 with cross-section or $\bar{\nu}$ data from an existing processed cross-section library, either containing resonance-reconstructed "point" data or containing multigroup data from the GROUPR module of NJOY. ERRORR is coded to treat all approved ENDF-4, -5, and -6 covariance formats for these four files. ERRORR can now also treat resolved-resonance covariances given in File 32 using the old Breit-Wigner resolved-resonance parameter uncertainties (`LRF=1` and `2`) in Version-5 format, the "Version-5 compatible" option of Version 6 (`LCOMP=0`) using the new formats that include resonance-resonance covariances, and the newest format based on Reich-Moore-Limited parameters that include resonance-resonance correlations between different reactions. ERRORR can also handle covariances for the P_1 Legendre component of angular distributions given in File 34 and covariances for energy distributions given in File 35.

The methodology of ERRORR assumes that the weighting flux used to convert energy-dependent cross sections into multigroup averages is free of uncertainty. In cases in which the cross-section information is obtained from an existing multigroup library, it is usually necessary to make assumptions about the shape of the cross section and the weight function within certain input energy groups.

A. Definitions of Covariance-Related Quantities

For convenient reference in discussing the methodology and input requirements of the ERRORR module, we next review the basic definitions of covariance-related quantities. Let x_0 and y_0 be the evaluated values of x and y, respectively:

$$x_0 \equiv \mathrm{E}[x] \ , \tag{273}$$

and

$$y_0 \equiv \mathrm{E}[y] \ . \tag{274}$$

Here E is the expectation operator, which performs an average over the joint probability distribution of x and y. The second moment of this distribution is called the covariance of x with y:

$$\mathrm{cov}(x, y) \equiv \mathrm{E}[(x - x_0)\,(y - y_0)] \ . \tag{275}$$

Covariance is a measure of the degree to which x and y are both affected by the same sources of error. The covariance of x with itself is called the variance of x:

$$\mathrm{var}(x) \equiv \mathrm{cov}(x, x) = \mathrm{E}[(x - x_0)^2] \ . \tag{276}$$

The more familiar standard deviation Δx (also called the "uncertainty") is simply

$$\Delta x \equiv [\mathrm{var}(x)]^{1/2} = [\mathrm{cov}(x, x)]^{1/2} \ . \tag{277}$$

The correlation between x and y (also called the correlation coefficient) is defined as

$$\mathrm{corr}(x, y) \equiv \frac{\mathrm{cov}(x, y)}{\Delta x \ \Delta y} \ . \tag{278}$$

The absolute value of a correlation coefficient is guaranteed to be less than or equal to unity. Another useful quantity is the relative covariance of x with y,

$$\mathrm{rcov}(x, y) \equiv \frac{\mathrm{cov}(x, y)}{x_0 \ y_0} \ . \tag{279}$$

Unlike $\mathrm{cov}(x, y)$, the relative covariance $\mathrm{rcov}(x, y)$ is a dimensionless quantity. Closely related to the relative covariance is the relative standard deviation,

$$\frac{\Delta x}{x_0} = \frac{[\mathrm{cov}(x, x)]^{1/2}}{x_0} \ , \tag{280}$$

which, from Eq. 279, can be written as

$$\frac{\Delta x}{x_0} = [\mathrm{rcov}(x, x)]^{1/2} \ . \tag{281}$$

Combining Eqs. 278 and 279, we have another useful result,

$$\mathrm{corr}(x, y) = \frac{\mathrm{rcov}(x, y)}{(\Delta x/x_0)(\Delta y/y_0)} \ . \tag{282}$$

While it is customary to speak of uncertainties and correlations as separate entities, these are actually just two different aspects of the covariance. If one has a set of absolute covariances for various reactions, including the self-covariance, then Eqs. 277 and 278 can be used to calculate Δx and $\mathrm{corr}(x, y)$. Similarly, if one has a set of relative covariances, one can use Eqs. 281 and 282 to calculate $\Delta x/x_0$ and $\mathrm{corr}(x, y)$.

Consider now a set of nuclear data σ_i with uncertainties characterized by the covariances $\mathrm{cov}(\sigma_i, \sigma_j)$. Let A and B be two linear functions of the σ_i,

$$A = \sum_i \ a_i \ \sigma_i$$

$$B = \sum_j b_j \, \sigma_j \ ,$$

where the a_i and b_j are sets of known constants. The above definitions can be used to calculate the covariances of the functions A and B induced by the covariances of the data. From Eq. 275,

$$\begin{aligned}\mathrm{cov}(A,B) &= \mathrm{E}\left\{ \left(\sum_i a_i \, \sigma_i \ - \sum_i a_i \mathrm{E}(\sigma_i) \right) \right. \\ &\qquad \left. \left(\sum_j b_j \, \sigma_j - \sum_j b_j \, \mathrm{E}(\sigma_j) \right) \right\} \\ &= \sum_{i,\,j} a_i \, b_j \, \mathrm{E}\left\{ \left(\sigma_i - \mathrm{E}(\sigma_i)\right) \, \left(\sigma_j - \mathrm{E}(\sigma_j)\right) \right\} \ ,\end{aligned}$$

so that

$$\mathrm{cov}(A,B) = \sum_{i,\,j} a_i \, b_j \, \mathrm{cov}(\sigma_i, \sigma_j) \ . \qquad (283)$$

This result, called the "propagation of errors" formula, is fundamental to the subject of multigroup processing of ENDF covariance data and will be referenced frequently in later parts of this section.

B. Structure of ENDF Files 31, 33, and 40: Energy-Dependent Data

Data in ENDF format are stored in various numbered "files," where the file number depends on the type of information contained. For example, the covariances of $\bar{\nu}(E)$ (the average number of neutrons per fission, which is a function of the incident neutron energy) are stored in File 31, where the possible "reaction" types are prompt $\bar{\nu}$, delayed $\bar{\nu}$, and total $\bar{\nu}$. File 33 contains the covariances of energy-dependent cross sections. In general, for data given in File N, covariance data are given in File (N+30). The structures of Files 31, 33, and 40 are identical and will be described first.

Files 31, 33, and 40 describe the covariances of energy-dependent data. To expand on this point, we recall that the full energy dependence of a cross section $\sigma(E)$, for example, is described in ENDF File 3 by specifying the cross-section values at a relatively small number of energy points and then providing a set of interpolation laws to be used in reconstructing the actual cross section at any intermediate energy. Somewhat the same philosophy is used to describe the two-dimensional energy dependence of data covariances in Files 31, 33, and 40. That is, one specifies a set of numerical data and a set of formulae, which together can be used to compute $\mathrm{cov}(x,y)$ for any desired pair of energies, E_x and E_y. Although the interpolation laws are presently restricted to the simple forms described below, it is not true (as sometimes stated) that ENDF contains multigroup covariances. The expression "multigroup covariance" refers to the covariance of one multigroup-averaged quantity with another averaged quantity, whereas ENDF contains the covariances between point-energy data. It is precisely the task of ERRORR to compute multigroup covariances, starting from point covariances.

Files 31, 33, and 40 of an evaluation for material `MAT` are divided into "sections," indexed by the reaction type `MT`. A section `(MAT,MT)` is further subdivided into "subsections." A subsection is the repository for all explicit statements of the two-dimensional energy dependence of the covariances of reaction `(MAT,MT)` with another reaction `(MAT1,MT1)`. Because covariances are symmetric, a subsection with `MAT1=MAT` and `MT1<MT` would be redundant with a subsection in an earlier section, and such data are, by convention, omitted from the ENDF files.

Subsections are further divided into "sub-subsections." Two different types of sub-subsections are used in the ENDF-5 and ENDF-6 formats. (Although ERRORR also treats data covariances in the earlier ENDF-4 format, this is of little practical interest since only three covariance evaluations were released in Version-4 format.) "NI-type" sub-subsections are used to express covariances explicitly, while "NC-type" sub-subsections are used to indicate the existence of connections between various data that result in "implicit" covariance contributions for various reaction pairs. We shall return to this point when discussing NC-Type Sub-Subsections.

a. NI-Type Sub-Subsections. Multiple NI-type sub-subsections are used to describe multiple, statistically independent sources of uncertainty for a given reaction pair. If $\mathrm{cov}(x,y)_n$ is the covariance computed from the data in one sub-subsection, then, because the uncertainties in different sub-subsections are uncorrelated,

$$\mathrm{cov}(x,y) \ = \sum_{n=1}^{\mathrm{NI}} \mathrm{cov}(x,y)_n \ ,$$

where NI is the number of NI-type sub-subsections in the current subsection. The numerical content of one NI-type sub-subsection consists of either one or two energy grids, a collection of constants, and a parameter `LB`. The parameter `LB` governs how the energies and constants are to be used in constructing the covariance in various rectangular regions of E_x–E_y space. For `LB=0, 1, 2,` and `8`, a single table containing pairs (E_i, F_i) is given. For `LB=3` and `4`, two such tables are given. For `LB=5`, a single set of energies E_i is given, along with an associated square matrix of constants G_{ij}. Finally, for `LB=6`, two energy grids are given, along with an associated rectangular matrix of constants G'_{ij}.

The first of these tables (E_i, F_i) defines a function $f(E)$ that is constant except for discrete steps at energies E_i,

$$f(E) = F_i, \ \text{if } E_i \le E < E_{i+1} \ . \qquad (284)$$

Similarly, if there is a second table,

$$f'(E) = F'_i, \ \text{if } E'_i \le E < E'_{i+1} \ . \qquad (285)$$

The LB=5 matrix data G_{ij} also define a function,

$$g(E_x, E_y) = G_{ij}, \text{ if } E_i \le E_x < E_{i+1} \text{ and } E_j \le E_y < E_{j+1} \ , \tag{286}$$

and similarly for LB=6

$$g'(E_x, E_y) = G'_{ij}, \text{ if } E_i \le E_x < E_{i+1} \text{ and } E'_j \le E_y < E'_{j+1} \ . \tag{287}$$

These functions are simply histograms in either one or two dimensions. Using the functions f, f', g, and g' thus defined, we can list the formulae in Eq. 288, which are used to specify energy-dependent covariances for the different allowed values of LB. Thus, if x is the value of the cross section or of the $\bar{\nu}$ value for the reaction (MAT,MT) that determines the ENDF/B section, and if y is the value for the reaction (MAT1,MT1) that determines the subsection, then for

$$\begin{aligned}
\text{LB} &= 0, \ \text{cov}(x,y)_n &&= f(E_x)\,\delta(E_x,E_y) \\
\text{LB} &= 1, \ \text{rcov}(x,y)_n &&= f(E_x)\,\delta(E_x,E_y) \\
\text{LB} &= 2, \ \text{rcov}(x,y)_n &&= f(E_x)\,f(E_y) \\
\text{LB} &= 3, \ \text{rcov}(x,y)_n &&= f(E_x)\,f'(E_y) \\
\text{LB} &= 4, \ \text{rcov}(x,y)_n &&= f(E_x)\,\delta(E_x,E_y)\ f'(E_x)\,f'(E_y) \\
\text{LB} &= 5, \ \text{rcov}(x,y)_n &&= g(E_x,E_y) \\
\text{LB} &= 6, \ \text{rcov}(x,y)_n &&= g'(E_x,E_y)\ .
\end{aligned} \tag{288}$$

The symbol $\delta(E_x, E_y)$ has the following meaning: $\delta(E_x, E_y)$=1 if E_x and E_y fall in the same energy interval of the first table (E_i, F_i), and $\delta(E_x, E_y)$=0 otherwise.

The final covariance law, LB=8, is an exceptional case that cannot be expressed in terms of point covariances. LB=8 is used primarily to represent uncertainty effects due to suspected, but unresolved, energy-dependent structure in a given cross section. If ΔE_j is the energy width of the j-th "union" energy group, and if this group lies within the k-th range (E_k, E_{k+1}) of an ENDF LB=8 energy grid, then the effect of the LB=8 subsection is to trigger the addition of an uncorrelated contribution of $F_k\,(E_{k+1} - E_k)/\Delta E_j$ to the (absolute) variance of the j-th union-group cross section. No contributions to off-diagonal elements of the multigroup covariance matrix are generated by an LB=8 sub-subsection.

b. NC-Type Sub-Subsections. NC-type sub-subsections, which describe covariances indirectly, are used in several evaluation situations, which are flagged by different values of the parameter LTY. The first situation, LTY=0, occurs when the following two conditions are met: (a) the covariances of a given reaction MT, both with itself and with other reactions, can be deduced in an energy range (E1,E2) solely from the application of a cross-section "derivation relation,"

$$x(\texttt{MAT,MT}; E) = \sum_i C_i\ x(\texttt{MAT,MT}_i; E), \tag{289}$$

and (b) the covariances of all of the reaction cross sections on the right-hand side of Eq. 289 have been given directly (that is, using only NI-type sub-subsections) throughout the range (E1,E2). The energy boundaries E1 and E2, the constants C_i, and the reaction identifiers $\texttt{MT}_i$ are specified in an NC-type sub-subsection with LTY=0. This format is widely used in ENDF/B libraries, and it makes possible the elimination of large volumes of otherwise redundant data. It also introduces considerable complexity in the multigroup processing, as discussed in Sect. XII, and adds to the computer running times. The presence of this one short sub-subsection affects the calculation of the covariances for many different reaction pairs, such as x(MAT,MT) with $x(\texttt{MAT,MT}_i)$. Less widely used are NC-type sub-subsections with LTY=1. These are employed when a reaction MT in material MAT is evaluated in some energy range (E1,E2) as a ratio to a standard reaction MTS in some other material MATS. That is,

$$x(\texttt{MAT,MT}; E) = R(E)\,x(\texttt{MATS,MTS}; E). \tag{290}$$

In practical evaluation situations, the uncertainty of R is almost never correlated with that of x(MATS,MTS). Because of this, the relative uncertainty in R is treated simply as one independent component of the relative uncertainty in x(MAT,MT), and it is described using normal NI-type sub-subsections. On the other hand, the contribution from uncertainty in x(MATS,MTS) is represented with an NC-type sub-subsection with LTY=1, which contains, in ENDF/B-V, only E1, E2, MATS, and MTS. The actual numerical covariance information must be read from the evaluation for the standard material MATS, which usually resides on an entirely different ENDF file. An NC-type sub-subsection with LTY=1, which occurs in a given subsection (MAT,MT; MAT,MT), affects the calculation of the covariances only for the current reaction pair (reaction MT with itself) and, in this respect, is more like an NI-type sub-subsection than, for example, an NC-type sub-subsection with LTY=0. This similarity is exploited in the processing of ratio covariances.

An NC-type sub-subsection with LTY=2 is used, in a similar way, to describe the covariances of x(MAT,MT) with x(MATS,MTS). As in the LTY=1 case, an LTY=2 sub-subsection contains only E1, E2, MATS, and MTS.

An NC-type sub-subsection with LTY=3 is included in material MATS to describe the (redundant) covariances of x(MATS,MTS) with x(MAT,MT). The numerical information contained here is the same as for LTY=1 and 2. An important function of LTY=3 data is to help locate reactions other than (MAT,MT) that have been measured relative to the same standard (MATS,MTS).

C. Resonance-Parameter Formats—File 32

File 32 contains covariances of resonance parameters from File 2. Older versions of ERRORR could only handle the ENDF-5 format or the "Version-5 compatible" format. Current versions can handle the newer ENDF-6 formats that include resonance-resonance correlations.

a. ENDF-5 Type Resonance Formats. With these formats (`LCOMP=0`), ERRORR processes File 32 in the following limited sense: when infinite-dilution cross-section covariances are processed (see previous section) from File 33, the diagonal elements of the resulting (self-reaction) multigroup covariance matrices are augmented by contributions based on the parameter covariances in File 32. These methods were sufficient for processing the covariance files in ENDF/B-VI.

For either of the permitted resolved resonance formalisms (`LRF=1` or `2`), the parameters considered in File 32 are the resonance energy E_r, the neutron width Γ_n, the radiative capture width Γ_γ, the fission width Γ_f, and (in Version 5 only) the total angular momentum J. All cross-parameter relative covariances, such as rcov(Γ_n,Γ_γ), are included, with the exception of the covariances of E_r with the remaining parameters, which are assumed to be negligible. Cross-resonance covariances, such as cov(E_r^i, E_r^j), where i and j refer to different resonances, are omitted in the `LCOMP=0` option.

b. ENDF/B-VII Resonance Covariances. For ENDF/B-VII, a number of evaluations include covariance formats that represent the correlations between resonance parameters for a given resonance and between different resonances. In NJOY99, these cases are handled by the ERRORJ [50] module, which was developed in Japan and contributed to the NJOY project. For NJOY10, the ERRORJ coding was integrated into the ERRORR module. The ERRORJ approach is based on computing the sensitivity of a cross section to a give resonance by numerical differencing and then combining these sensitivities with parameter covariances from the evaluation.

There are three basic covariance format options, governed by the `LCOMP` and/or `LRU` flags. The `LRU` flag distinguishes between resolved resonance (`LRU=1`) and unresolved resonance (`LRU=2`) data. For `LCOMP=1/LRU=1`, there may be two classes of data, designated as `NSRS` or `NLRS` sub-subsections. NSRS sub-subsections provide covariance data among parameters of specified resolved resonances. Covariances may be given for all ENDF recognized resolved resonance formats (as specified via the `LRF` flag). NLRS sub-subsections provide long-range parameter covariances. These data are allowed for all resolved resonance formats. At present there are no NLRS sub-subsection data in ENDF/B-VII and NJOY does not process these data.

The LCOMP=2/LRU=1 format provides a combination of resolved resonance parameters and their uncertainties plus a correlation matrix. The correlation matrix is given in a special packed form using the ENDF INTG record format with matrix elements consisting of as few as 2 to as many as 6 digits. Covariance matrix elements, V_{ij} can be reconstructed as

$$V_{ij} = D_i C_{ij} D_j \tag{291}$$

where D_i is the uncertainty on parameter i and C_{ij} is the correlation matrix element relating parameters i and j. Since the correlation matrix is symmetric, and its diagonal terms are unity, only the upper triangle of matrix elements is given in the ENDF file. This more compact representation of the covariances helps for materials with large numbers of resonances.

Finally, the `LRU=2` (`LCOMP` is not defined and its location in the ENDF file is typically set to zero) format is used to define covariances for unresolved resonance parameters. The data represent an energy independent relative covariance matrix even if the underlying unresolved resonance parameters are energy dependent. This is a symmetric matrix and so only the upper triangular portion of the matrix is provided in the ENDF file.

The original covariance data formats did not allow the evaluator to define an uncertainty in the scattering radius (only for resonance energy and the total or partial resonance widths). This deficiency was eliminated in 2009, but until new evaluations are released there are no scattering radius uncertainty data in current evaluated library files. NJOY can include scattering radius uncertainty effects if such data are given in the evaluated file. In addition, users can specify an uncertainty in the ERRORR input so that this important component to the cross section uncertainty is considered when processing existing files that lack these data.

c. Reich-Moore-Limited Covariance Representations. The ENDF-6 format also allows resonances using the `LRF=7` Reich-Moore-Limited representation. In addition to the covariances described above, the RML approach can define channel-channel covariances properly. There are two experimental evaluations currently available that use this approach. The ^{19}F evaluation from the Oak Ridge National Laboratory (ORNL) represents the four reactions elastic, capture, (n,n$_1$), and (n,n$_2$). Fig. 56 shows that ERRORR can generate the covariances between elastic and (n,n$_1$). The ^{35}Cl evaluation from ORNL represents the reactions elastic, capture, and (n,α). The coding in NJOY10 to handle these cases uses analytic calculations for the parameter sensitivities borrowed from SAMMY.

D. Secondary Particle Angular Distribution Covariances—File 34

File 34 contains covariances for angular distributions of secondary particles. While the underlying angular distribution data in File 4 may be given as tabulated distributions or as Legendre polynomial coefficients, File 34's covariance data are only given for Legendre coefficients. At present there is no provision to specify covariances between cross sections from File 3 and angular distributions from File 4, nor to specify angular distribution covariances between different materials. The data format is governed by the `LB` flag discussed above in Sect. XI B with LB values of 0, 1, 2, 5 and 6 allowed.

File 34 in the ENDF file may include covariance data for any MT reaction defined in File 4, but currently

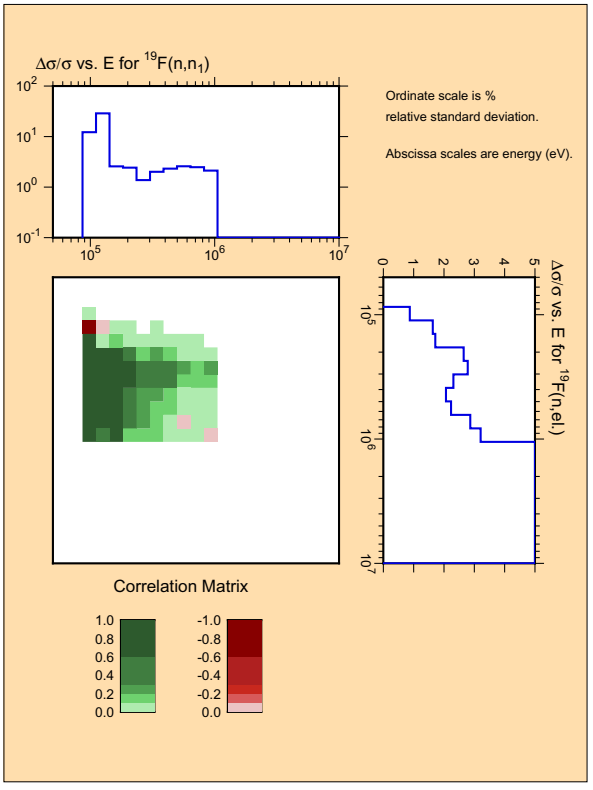

FIG. 56: Covariance plot for the elastic and (n,n$_1$) reactions in the experimental evaluation for ^{19}F, demonstrating the channel-channel covariances possible when using the Reich-Moore-Limited resonance representation.

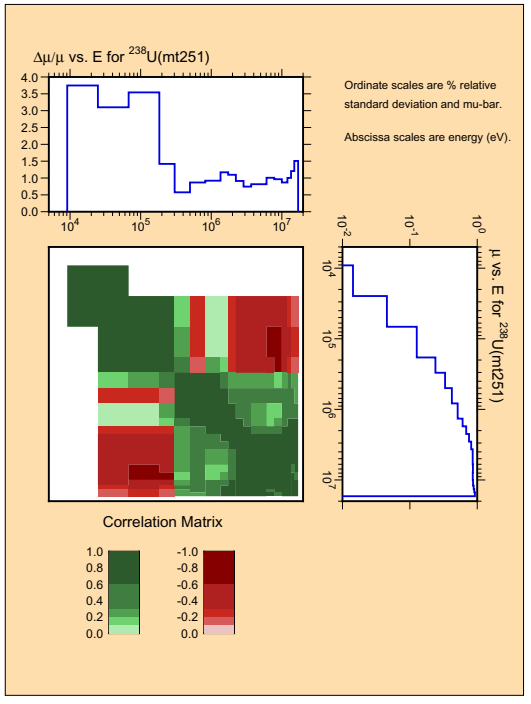

FIG. 57: Example of angular distribution covariances.

NJOY only processes the P1 component of MT=2 (elastic scattering).

Fig. 57 shows an example of the covariances computed from File 34 for U-238 from JENDL-3.3. As usual, there are three components to this plot. On the right side, we illustrate $\bar{\mu}$ *vs* energy, which displays the expected forward-peaked characteristic; on the top is the uncertainty in $\bar{\mu}$ *vs* energy; and in the center is the correlation matrix. The ERRORR input for this case will be presented below.

E. Secondary Particle Energy Distribution Covariances—File 35

File 35 contains covariance matrices for the energy distributions of secondary particles given in File 5. If the spectral distributions are correlated with angular distributions and given in File 6, the covariance information still appears in File 35 (the MT section identifier is the common datum relating these data) and refers to the angle-integrated distributions only. The secondary energy distribution is usually defined on a relatively fine energy grid, and multiple distributions are given as a function of increasing incident particle energy. Since the uncertainties in secondary distributions are usually highly correlated as a function of incident particle energy, it is generally sufficient to only define a few covariance matrices over relatively broad incident energy groups. There is no provision to specify covariances between these groups nor with data from other files such as File 3 cross sections or File 1 prompt, delayed or total nu.

File 35 covariance data are given in a series of `NK` subsections, with each subsection covering a unique incident particle energy range. The lowest energy of the first subsection and the highest energy of the last subsection must cover the entire incident particle energy range from File 5 (or File 6). The data tabulated in the ENDF file are normalized probabilities of absolute covariances identified as an `LB=7` subsection. `LB=7` is identical to `LB=5` defined in Sect. XI B except that these data are now absolute rather than relative covariances.

The ENDF format manual notes these matrices are probability distributions that must remain normalized to unity and therefore the elements in these symmetric matrices are constrained such that the sum of the elements in any row (or column) must be zero. Of course, when dealing with real numbers of finite precision whose values can vary by orders of magnitude it is virtually impossible to rigorously conform to this rule. Therefore, if the sum is sufficiently small, the rule is judged to be satisfied. If not, the ENDF manual provides a correction formula to apply to all matrix elements. When processing File 35 data NJOY, checks this summation requirement and, if necessary, applies the required correction.

At present there are no File 35 data in ENDF/B-VII

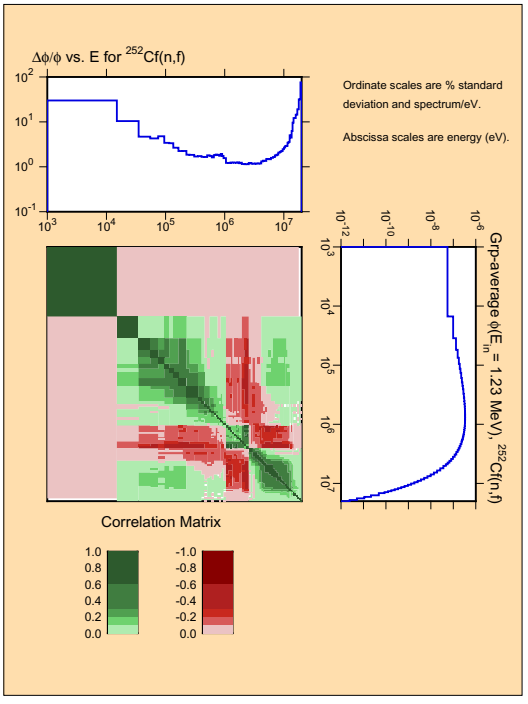

FIG. 58: Example of energy distribution covariances.

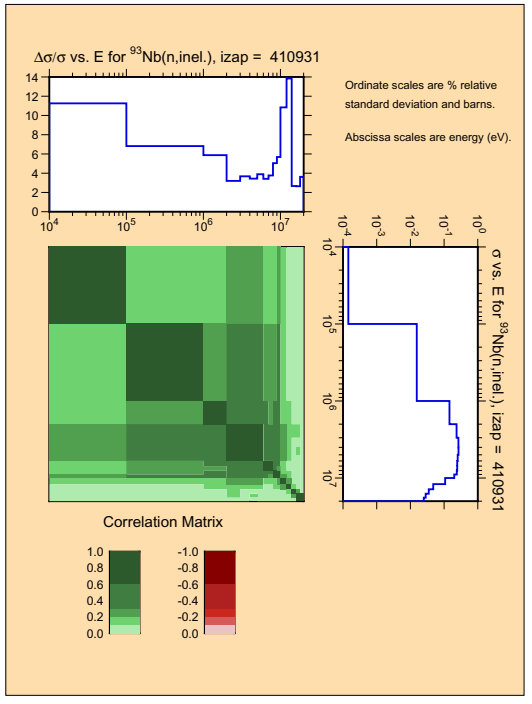

FIG. 59: Example of radioactive nuclide production covariances.

neutron files, but the ^{252}Cf decay file does contain such data. These data are becoming available in preliminary ENDF/B-VII.1 files currently stored at the National Nuclear Data Center (NNDC) and Brookhaven National Laboratory (BNL) as "ENDF/A" files. These files follow the "zero sum' rule discussed above, but users are cautioned that other internationally distributed files, particularly those from JENDL-3.3, may provide File 35 data in an alternate format. The matrix elements in the JENDL-3.3 files have been divided by the group energy width to yield energy-group averaged probability distributions, as this is the data representation expected by the Japanese ERRORJ code [50]. A conversion code, chmf35, has been developed and is available from the Nuclear Energy Agency (*http://www.nea.fr/dbprog/Njoy/chmf35.for*) to reformat these files so that they are suitable for processing by NJOY.

Fig. 58 shows the results of processing this fictitious evaluation. The ERRORR input for this case will be presented below.

F. Radioactive Nuclide Production Covariances–File 40

There is only one file in ENDF/B-VII that includes File 40 covariance data—^{93}Nb. The input for calculating the ^{93m}Nb production uncertainty will be shown below, but the resulting covariances are shown here in Figure 59. The form of the graphical output obtained when processing File 40 is identical to that produced when processing File 33; with the cross section shown to the right, the uncertainty shown to the top and the correlation matrix shown in the center. That these are File 40 data can be determined from the label on the uncertainty portion of the plot. In 2009, the File 40 format was modified to include the IZAP parameter that identifies the daughter product. We include this IZAP value in the plot label so that the user can more fully identify these data. If an older File 40 file is processed that does not define IZAP, then the text string "MF40" will appear in the title. In the example shown here we have modified the original ENDF/B-VII ^{93}Nb file to include the proper IZAP value in File 40, which then appears in the plot title.

G. Calculation of Multigroup Fluxes, Cross Sections, and Covariances on the Union Grid

As mentioned before, the main function of the ERRORR module is to calculate the uncertainty in group-averaged cross sections at infinite dilution due to uncertainty in the ENDF point data. In this and the following sections, we describe the procedures used in performing this task.

In order to proceed, it is necessary to introduce into the discussion three different energy grids, namely, the user grid, the ENDF grid, and the union grid. The user grid is the multigroup structure in which the output multigroup covariances are to be produced. The ENDF grid is the collection of energies obtained by forming the

union of (a) all energy "lists" appearing in any NI-type sub-subsection of any subsection to be processed in the current ENDF material, and (b) all energy pairs used to define the range of effectiveness of any NC-type sub-subsection of any subsection to be processed. The union grid is the union of the user grid and ENDF grid. The utility of the union grid is that (a) the covariances are particularly simple to calculate in this grid, as discussed below, and (b) the multigroup covariances needed by the user are then easily obtainable by a straightforward collapse from this to the user grid. In fact, the design of the current ENDF covariance format was strongly influenced by the desire to employ this particular procedure for multigroup processing [51, 52].

After the union grid is formed, the cross sections $x(E)$ and weighting flux $\phi(E)$ are integrated to produce x_I and ϕ_I, multigrouped on this grid. If point cross sections are supplied, an exact integration is done. If, on the other hand, a multigroup cross-section library is supplied, then a collapsing method is used. If a library group is subdivided by a union-group boundary, then over the span of that library group, the unknown energy dependencies of $x(E)$ and $\phi(E)$, which are needed to calculate x_I and ϕ_I, are approximated by constants. Normally, the effect of this approximation is not large and, in any case, can be reduced or eliminated by increasing the number of groups in the input library.

We next consider the theoretical basis for the calculation of union-grid multigroup covariances, as performed in subroutine `covcal`. By definition, x_I is just the average of $x(E)$ over union group I,

$$x_I \equiv \frac{\int_I \phi(E)\, x(E)\, dE}{\int_I \phi(E)\, dE} , \tag{292}$$

where $\phi(E)$ is the flux "model" assumed for the multigroup calculations. Let y_J denote similarly the average of $y(E)$ over union group J.

Let us imagine that these groups are subdivided into many subintervals of infinitesimal width, so that in the i-th subinterval of group I, for example, $x(E)$ can be well approximated by the constant x_i. By this device, the integrals that define x_I and y_J can be converted to discrete sums:

$$x_I = \frac{\sum_{i\in I} \phi_i\, x_i}{\phi_I} = \sum_{i\in I} \alpha_{Ii}\, x_i , \tag{293}$$

where

$$\phi_i = \int_i \phi(E)\, dE , \tag{294}$$

$$\phi_I = \sum_{i\in I} \phi_i = \int_I \phi(E)\, dE , \tag{295}$$

and

$$\alpha_{Ii} \equiv \frac{\phi_i}{\phi_I} . \tag{296}$$

From these definitions, clearly

$$\sum_{i\in I} \alpha_{Ii} = 1 . \tag{297}$$

Similarly for y,

$$y_J = \sum_{j\in J} \alpha_{Jj}\, y_j , \tag{298}$$

and

$$\sum_{j\in J} \alpha_{Jj} = 1 . \tag{299}$$

The methodology of ERRORR assumes that $\phi(E)$ in Eq. 292 is free of uncertainty. Under this assumption, the terms α_{Ii} and α_{Jj} are simply known constants. The covariance of x_I with y_J can then be calculated using the propagation-of-errors formula, Eq. 283, together with Eqs. 293 and 298.

$$\begin{aligned} \mathrm{cov}(x_I, y_J) &= \sum_{\substack{i\in I\\ j\in J}} \alpha_{Ii}\, \alpha_{Jj}\, \mathrm{cov}(x_i, y_j) \\ &= \sum_{\substack{i\in I\\ j\in J}} \alpha_{Ii}\, \alpha_{Jj} \sum_n \mathrm{cov}(x_i, y_j)_n , \end{aligned} \tag{300}$$

where the summation over n results from the different independent contributions to the ENDF point covariances coming from the different NI-type sub-subsections. Changing the order of summation, we obtain

$$\mathrm{cov}(x_I, y_J) = \sum_n \mathrm{cov}(x_I, y_J)_n , \tag{301}$$

where

$$\mathrm{cov}(x_I, y_J)_n = \sum_{\substack{i\in I\\ j\in J}} \alpha_{Ii}\, \alpha_{Jj}\ \mathrm{cov}(x_i, y_j)_n . \tag{302}$$

To evaluate the sum in Eq. 302, we make use of the fact that union groups I and J do not cross any ENDF grid boundaries. Recalling the discussion of NI-type sub-subsections (and excluding for the moment the case where sub-subsections with `LB=8` are present), there are only two possibilities for the energy dependence of the covariance between ENDF grid points; thus, over the limits of the sum, either $\mathrm{cov}(x_i, y_j)_n$ is independent of i and j (if `LB=0`) or $\mathrm{rcov}(x_i, y_j)_n$ is independent of i and j (if $1 \le$ `LB` ≤ 6). We consider first the constant-absolute-covariance case, `LB=0`. Eq. 302 can then be rewritten as

User	|		ϕ_1, X_1		|			ϕ_2, X_2			|
ENDF	|	F_1		|		F_2			|	F_3	|
Union	|	ϕ_1, x_1		|	ϕ_2, x_1	|	ϕ_3, x_3		|	ϕ_4, x_4	|

FIG. 60: Illustration of energy grid relations.

follows:

$$\begin{aligned}\mathrm{cov}(x_I, y_J)_n &= \mathrm{cov}(x,y)_n \sum_{\substack{i\in I\\ j\in J}} \alpha_{Ii}\ \alpha_{Jj} \\ &= \mathrm{cov}(x,y)_n \left(\sum_{i\in I} \alpha_{Ii}\right)\left(\sum_{j\in J} \alpha_{Jj}\right) . \end{aligned} \tag{303}$$

Invoking Eqs. 297 and 299, we obtain

$$\mathrm{cov}(x_I, y_J)_n = \mathrm{cov}(x,y)_n \qquad (\mathtt{LB}=0) . \tag{304}$$

For LB-values ranging from 1 to 6, the relative covariance is constant over each union group, so we rewrite Eq. 302 in the form

$$\begin{aligned}\mathrm{cov}(x_I, y_J)_n &= \sum_{\substack{i\in I\\ j\in J}} \alpha_{Ii}\ \alpha_{Jj}\ x_i\ x_j\ \mathrm{rcov}(x_i, x_j)_n \\ &= \mathrm{rcov}(x,y)_n \sum_{\substack{i\in I\\ j\in J}} (\alpha_{Ii}\ x_i)\ (\alpha_{Jj}\ y_j) \\ &= \mathrm{rcov}(x,y)_n \left(\sum_{i\in I} \alpha_{Ii}\ x_i\right)\left(\sum_{j\in J} \alpha_{Jj}\ y_j\right) \end{aligned} \tag{305}$$

Substituting here from Eqs. 293 and 298, we obtain

$$\mathrm{cov}(x_I, y_J)_n = x_I\ y_J\ \mathrm{rcov}(x,y)_n \qquad (1 \le \mathtt{LB} \le 6) . \tag{306}$$

The final union-group multigroup covariance is obtained by inserting these results, Eqs. 304 and 306, back into Eq. 301

$$\begin{aligned}\mathrm{cov}(x_I, y_J) &= \sum_{\mathrm{n(LB=0)}} \mathrm{cov}(x,y)_n \\ &+ \sum_{\mathrm{n(LB=1,6)}} x_I\ y_J\ \mathrm{rcov}(x,y)_n , \end{aligned} \tag{307}$$

where the first sum runs over all sub-subsections with LB=0 and the second runs over all sub-subsections with LB=1 through 6. The quantities $\mathrm{cov}(x,y)_n$ and $\mathrm{rcov}(x,y)_n$ here are simply the point-energy covariances from the ENDF covariance file, as described in Eqs. 284 through 288. Equation 307 then is the basic equation used in subroutine covcal to calculate the desired union-group covariances.

The final step, if sub-subsections with LB=8 are present, is to increment the diagonal elements (variances) as follows:

$$\mathrm{cov}(x_I, x_I) = \mathrm{cov}(x_I, x_I) + F_k\,(E_{k+1} - E_k)/\Delta E_I , \tag{308}$$

where k indexes the range of the LB=8 energy grid that includes union group I.

H. Basic Strategy for Collapse to the User Grid

The union-group fluxes ϕ_I are used to collapse the union-group cross sections to the coarser user grid. Changing notation slightly, let us denote by $x_I(a)$ the cross section in union group I for reaction a, and similarly let $X_K(a)$ be the cross section in user group K for the same reaction. In complete analogy with Eq. 293

$$X_K(a) = \frac{\sum_{I\in K} \phi_I\ x_I(a)}{\phi_K} = \sum_{I\in K} A_{KI}\ x_I(a) , \tag{309}$$

where

$$\phi_K = \sum_{I\in K} \phi_I , \tag{310}$$

and

$$A_{KI} = \frac{\phi_I}{\phi_K} . \tag{311}$$

Applying the propagation-of-errors formula again gives

$$\mathrm{cov}\,[X_K(a), X_L(b)] = \sum_{\substack{I\in K\\ J\in L}} A_{KI}\ A_{LJ}\ \mathrm{cov}[x_I(a), x_J(b)] . \tag{312}$$

An alternative expression, obtained by simply rearranging coefficients, is

$$\mathrm{cov}\,[X_K(a), X_L(b)] = \frac{1}{\phi_K \phi_L} \sum_{\substack{I\in K\\ J\in L}} T_{IJ}(a,b) , \tag{313}$$

where

$$T_{IJ}(a,b) = \phi_I\ \phi_J\ \mathrm{cov}[x_I(a), x_J(b)] . \tag{314}$$

Eqs. 313 and 314 provide the basic framework in the ERRORR module for the production of multigroup covariances in the coarse user-group structure. Finally, if the user requests it, the absolute covariances from Eq. 313 are converted to relative form,

$$\mathrm{rcov}[X_K(a), X_L(b)] = \frac{\mathrm{cov}[X_K(a), X_L(b)]}{X_K(a)\ X_L(b)} . \tag{315}$$

I. Group-Collapse Strategy for Data Derived by Summation

The procedure described in the previous section must be modified if, in some energy range, either reaction a or b

is a "derived" quantity in the sense of Eq. 289 or Eq. 290. In such a case, one cannot apply Eq. 307 directly to the calculation of $\mathrm{cov}[x_I(a), x_J(b)]$, because the covariances on the right-hand side of Eq. 307 will be missing from the ENDF file in the affected energy region.

We consider first the case in which some cross sections are evaluated ("derived") by simply summing other evaluated cross sections, as in Eq. 289. Equation 308 can then be rewritten as

$$X_K(a) = \sum_{I \in K} A_{KI} \left\{ \sum_{\text{all c}} C_I(a,c)\, x_I(c) \right\} , \qquad (316)$$

where the quantity in curly brackets is the value of the derived cross section "a" in union group I, as reconstructed from the directly evaluated cross sections $x_I(c)$. The derivation coefficients $C_I(a,c)$ depend on the union-group index I, because the evaluator is permitted to employ different derivation strategies in different energy ranges in order to simplify and shorten the covariance files.

To expand on this last point, suppose there are only three reactions, and suppose that the cross sections $x(1)$ and $x(2)$ rigorously sum to $x(3)$ at all energies. Further suppose that within the energy range covered by the first union group, the cross-section evaluator has used this logical connection in order to "derive" $x(3)$, that is to evaluate $x(3)$, by summing the existing evaluations for $x(1)$ and $x(2)$,

$$x_1(3) = x_1(1) \; + \; x_1(2) \; . \qquad (317)$$

Further suppose that in the range of union group 2, $x(1)$ is derived by employing the same logical connection, but in a different way,

$$x_2(1) = x_2(3) - x_2(2) \; , \qquad (318)$$

and in union group 3, all three reactions are directly evaluated.

In this example $C_I(a,c)$ has the following values:

$$\begin{array}{lll}
C_1(1,1) = 1 & C_1(1,2) = 0 & \boxed{C_1(1,3) = 0} \\
C_1(2,1) = 0 & C_1(2,2) = 1 & \boxed{C_1(2,3) = 0} \\
C_1(3,1) = 1 & C_1(3,2) = 1 & \boxed{C_1(3,3) = 0} \\
 & & \\
\boxed{\begin{array}{l} C_2(1,1) = 0 \\ C_2(2,1) = 0 \\ C_2(3,1) = 0 \end{array}} & \begin{array}{l} C_2(1,2) = -1 \\ C_2(2,2) = 1 \\ C_2(3,2) = 0 \end{array} & \begin{array}{l} C_2(1,3) = 1 \\ C_2(2,3) = 0 \\ C_3(3,3) = 1 \end{array} \\
 & & \\
\begin{array}{l} C_3(1,1) = 1 \\ C_3(2,1) = 0 \\ C_3(3,1) = 0 \end{array} & \begin{array}{l} C_3(1,2) = 0 \\ C_3(2,2) = 1 \\ C_3(3,2) = 0 \end{array} & \begin{array}{l} C_3(1,3) = 0 \\ C_3(2,3) = 0 \\ C_3(3,3) = 1 \end{array}
\end{array}$$

Note that in all cases we have formally considered the evaluated cross sections to be derived from themselves, $C_I(a,c)|_{\text{eval}}=\delta_{ac}$. This device allows us to use Eq. 315 for all reactions, regardless of whether they are derived or evaluated. Note that, in every case where reaction c is derived in group I, $C_I(a,c)$=0. (See boxed submatrices.) This null "sensitivity coefficient" is important, because it means that Eq. 315 remains a linear relation involving only dependent quantities with known covariances (the evaluated subset of the union-group cross sections) on the right. This allows us once again to use the propagation-of-errors formula to obtain the desired user-group covariances.

$$\mathrm{cov}[X_K,(a), X_L(b)] = \sum_{\substack{\text{all c} \\ \text{all d}}} \sum_{\substack{I \in K \\ J \in L}} A_{KI}\, C_I(a,c)\, A_{LJ}\, C_J(b,d)\, \mathrm{cov}[x_I(c), x_J(d)], \qquad (319)$$

or, alternatively,

$$\mathrm{cov}[X_K,(a), X_L(b)] = \frac{1}{\phi_K \phi_L} \sum_{\substack{\text{all c} \\ \text{all d}}} \sum_{\substack{I \in K \\ J \in L}} C_I(a,c)\, C_J(b,d)\, T_{IJ}(c,d) \; , \qquad (320)$$

where

$$T_{IJ}(c,d) = \phi_I \; \phi_J \; \mathrm{cov}[x_I(c), x_J(d)] \; . \qquad (321)$$

In the fairly common situation where cross sections $x(a)$ and $x(b)$ are directly evaluated over the whole energy range of the data file,

$$C_I(a,c) = \delta_{ac}, \text{ for all } I, \qquad (322)$$

and

$$C_J(b,d) = \delta_{bd}, \text{ for all } J. \qquad (323)$$

In this case, Eq. 319 can be greatly simplified.

$$\begin{aligned} \mathrm{cov}[X_K(a), X_L(b)] &= \frac{1}{\phi_K \phi_L} \sum_{\substack{\text{all c} \\ \text{all d}}} \sum_{\substack{I \in K \\ J \in L}} \delta_{ac}\, \delta_{bd}\, T_{IJ}(c,d) \\ &= \frac{1}{\phi_K \phi_L} \sum_{\substack{I \in K \\ J \in L}} T_{IJ}(a,b) \; . \end{aligned} \qquad (324)$$

J. Processing of Data Derived from Ratio Measurements

The treatment of implicit covariances that arise from ratio measurements, as in Eq. 290, is totally different from the treatment of data derived by summation, Eq. 289, discussed in the previous section. Before discussing the processing details, it is helpful first to review the subject of ratio evaluations generally. Returning to the "x-y" notation, let $x(E_x)$ be the value of the cross section for reaction x at energy E_x, and $y(E_y)$ the cross section for reaction y at energy E_y. In some energy region (L_x, H_x), suppose that the best knowledge of $x(E_x)$

is obtained through the application of a measured ratio, $u(E_x)$:

$$x(E_x) = u(E_x)\ z(E_x),\ \text{if}\ L_x \le E_x \le H_x\ , \qquad (325)$$

where reaction z is some well-known cross section, possibly one of the official ENDF/B standard cross sections. Similarly, suppose y is derived from the same standard, over a possibly different energy range:

$$y(E_y) = v(E_y)\ z(E_y),\ \text{if}\ L_y \le E_y \le H_y\ . \qquad (326)$$

By performing a first-order Taylor-series expansion and then applying the formula for propagation of errors, one can obtain an expression for the contribution to the relative covariance $\mathrm{rcov}[x(E_x), y(E_y)]$ that is attributable to these ratio measurements. In the usual case, where the ratios u and v are only weakly correlated with the standard cross section z, the result is quite simple:

$$\mathrm{rcov}[x(E_x), y(E_y)]_{\mathrm{ratio}} = \mathrm{rcov}[u(E_x), v(E_y)] + \mathrm{rcov}[z(E_x), z(E_y)]\ , \qquad (327)$$

if $L_x \le E_x \le H_x$ and $L_y \le E_y \le H_y$, and

$$\mathrm{rcov}[x(E_x), y(E_y)]_{\mathrm{ratio}} = 0\ \ \text{otherwise.} \qquad (328)$$

Thus, in this fairly common evaluation situation, the covariance separates naturally into a part involving only the measured ratios and a part involving only the standard. Because the second contribution, $\mathrm{cov}(\mathrm{MT}_z,\ \mathrm{MT}_z)$, can be read directly from the NI-type sub-subsections in the evaluation for the standard, it is not included explicitly in the ENDF subsections for the derived quantities, $\mathrm{cov}(\mathrm{MT}_x,\ \mathrm{MT}_x)$ or $\mathrm{cov}(\mathrm{MT}_y,\ \mathrm{MT}_y)$. Instead, the existence of this additional contribution to the covariance is signaled by the presence of an NC-type sub-subsection.

In addition to identifying the standard reaction, the NC-type sub-subsection contains a control parameter `LTY` and two energies `EL` and `EH` whose significance depends on `LTY`. `LTY` is used to identify particular evaluation scenarios: reaction x is the same as reaction y (or at least they are derived from z over the same energy range) (`LTY=1`); y is identical to the standard (`LTY=2`); or x is identical to the standard (`LTY=3`). A fourth possibility, namely, that x, y, and z are entirely distinct, cannot presently be treated with a single NC-type sub-subsection; that is, there is no `LTY` value defined for this case. However, as discussed later, it is still possible to process covariances for this situation by combining information from sub-subsections in two different evaluations. For convenience, we shall refer to this fourth case as `LTY=4`.

The interrelationship of `LTY`, `EL`, `EH`, and the windows (L_x, H_x) and (L_y, H_y) used in ERRORR for the "zeroing-out" operation of Eq. 328, is summarized below:

TABLE VI: Covariance matrices affected by ratio measurements in ENDF/B-V.

	^{10}B (n,α)	^{238}U (n,γ)	^{235}U (n,f)	^{239}Pu (n,f)	^{239}Pu (n,γ)	^{241}Am (n,f)	^{242}Pu (n,f)
^{10}B(n,α)	$*$	3					
^{238}U(n,γ)	2	1					
^{235}U(n,f)			$*$	3	3	3	3
^{239}Pu(n,f)			2	1	1	4	4
^{239}Pu(n,γ)			2	1	1	4	4
^{241}Am(n,f)			2	4	4	1	4
^{242}Pu(n,f)			2	4	4	4	1

$*$ = standard
Integer = `LTY` value (see text)

`LTY=1`

$$(L_x, H_x) = (L_y, H_y) = (\mathtt{EL}, \mathtt{EH})$$

`LTY=2`

$$(L_x, H_x) = (\mathtt{EL}, \mathtt{EH})$$
$$(L_y, H_y) = (10^{-5}\ \mathrm{eV},\ 20\ \mathrm{MeV})$$

`LTY=3`

$$(L_x, H_x) = (10^{-5}\ \mathrm{eV},\ 20\ \mathrm{MeV})$$
$$(L_y, H_y) = (\mathtt{EL}, \mathtt{EH})$$

`LTY=4`

$$(L_x, H_x) = (\mathtt{EL}, \mathtt{EH})$$
$$(L_y, H_y) = (\mathtt{EL'}, \mathtt{EH'})\ . \qquad (329)$$

If the user requests covariance data $\mathrm{cov}(x, y)$, where x and y are different and both are distinct from the standard (`LTY=4`), the ERRORR module obtains (`EL,EH`) from the `LTY=2` subsection in the evaluation for x that "points" to z. Then, the covariance file for z is read to obtain both (a) the explicit covariances $\mathrm{cov}(z, z)$ and (b) the second energy window (`EL'`, `EH'`), the latter being found in the `LTY=3` sub-subsection that points back to y.

We will illustrate ratio covariance processing using data from ENDF/B-V. Table VI lists all ENDF/B-V reactions that contain ratio-to-standard covariance data. The symbols entered in the reaction-by-reaction matrix indicate which reactions are referenced as standards ($*$), and which reaction pairs have implicit nonzero covariances (`LTY`). ERRORR will produce multigroup covariances for any of the reaction pairs in Table I that are marked with ($*$) or (`LTY`). The cross-material covariances (`LTY=2`, `3`, or `4`) must be requested individually using the `IREAD=2` option (see Sect. XI M), especially the discussion of Cards 10 and 11). An attempt to process covariances for any of the cases `LTY=1` through `4` without supplying a separate ENDF file containing the needed standard will result in an error stop in subroutine `gridd`. The error diagnostic, however, will supply the details of the corrective action required.

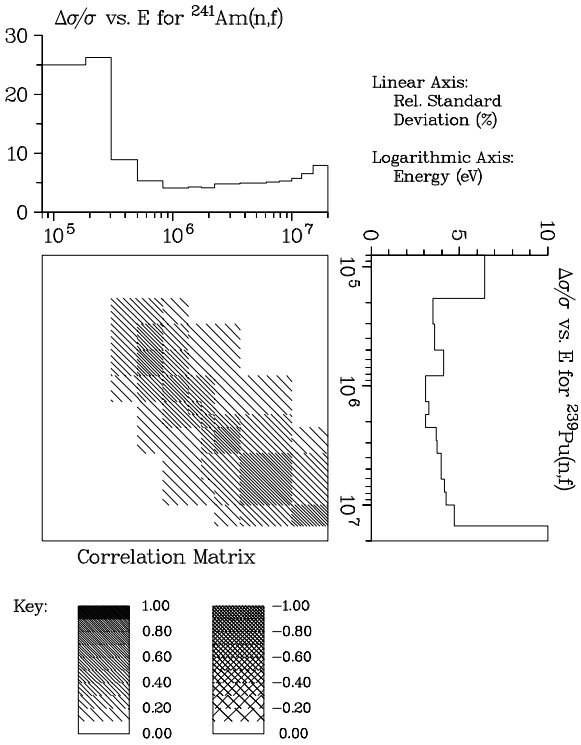

FIG. 61: Covariance data for ^{239}Pu(n,f) with ^{241}Am(n,f). This older figure illustrates the black&white option for covariance plots using hatching to represent levels of correlation.

As an example of the ratio-data capabilities of ERRORR, Fig. 61 shows the covariances between the fission cross sections of ^{239}Pu (reaction x) and those of the important actinide ^{241}Am (reaction y). This is an `LTY=4` case where (L_x, H_x)= (0.2 MeV, 15 MeV) and (L_y, H_y)=(0.2 MeV, 20 MeV). The effect of using two different windows is apparent in the lower right corner of the correlation matrix. The plot itself was produced with the COVR module. It illustrates the black&white format for correlation plots using hatching to represent the various levels of the correlation. The complete NJOY input needed to compute and plot these data is given below.

K. Multigroup Processing of Resonance-Parameter Uncertainties

In some materials, and in certain energy regions, the cross-section uncertainty is dominated by the uncertainty in resonance parameters. As described above, there are several formats available in the ENDF-6 format for describing resonance covariances. When using the early ENDF-5 format (which is available in the ENDF-6 format as `LCOMP=0`, the resonance-parameter contribution to the uncertainty in infinite-dilution fission and capture cross sections is included automatically when cross-section covariances are processed. The contribution is obtained from the Breit-Wigner formulae for the fission and capture areas of a resonance, A_f and A_γ. By differentiating these formulae with respect to the resonance parameters, one obtains a set of sensitivities. With these sensitivities and the covariance matrix of the parameters from MF32, one can apply the propagation-of-errors formula to obtain the covariances $\mathrm{cov}(A_\gamma, A_\gamma)$, $\mathrm{cov}(A_\gamma, A_f)$, and $\mathrm{cov}(A_f, A_f)$.

The resonance contribution is properly weighted with the isotopic abundance and the ratio of the weight function at the resonance to the average weight in the group. It is assumed, however, that the area of a resonance lies entirely within the group that contains the resonance energy E_r. Because of this assumption, and because ENDF-5 and the `LCOMP=0` option of ENDF-6 provide no correlations between parameters of different resonances, the calculated resonance-parameter contribution affects only the diagonal elements of the affected matrices, $\mathrm{cov}[X_K(a), X_K(b)]$.

When one of the more advanced options (`LCOMP=1` or `LCOMP=2`) is found in MF32, and for Single-Level Breit-Wigner, Multi-Level Breit-Wigner, or Reich-Moore resonance parameters, the ERRORR code branches to the ERRORJ logic. The resonance cross sections are computed on a special grid and group averaged. Then each parameter is changed in turn to generate new cross sections. Sensitivity parameters are computed from the differences. These sensitivity coefficients are then combined with the covariances of the parameters to obtain covariances on the multigroup cross sections. The matrices are very large in many cases, and the folding is time consuming. There are two formats in use for the parameter covariances. One gives each value in a full 11-digit ENDF field. With many resonances, this can become very bulky. The other uses a more compact representation based on the correlations. Just a few digits are used for each correlation, which allows several values to be packed into each 11-digit field, thus reducing the size of the evaluation. The coding must reconstruct the actual covariance values before folding them with the sensitivities, so this economization doesn't help with the memory needed or the time required to fold the data in the final cross section covariances.

If Reich-Moore-Limited resonances are found (`LRF=7`, a different branch is taken. In this case, it is possible to use analytic formulas to compute the cross section sensitivities to the parameters. These formulas are given in detail in the SAMMY reference [16]. These sensitivities are computed on a special energy grid and group averaged. They are then folded with the parameter covariances as described above to obtain the cross section covariances.

L. Processing of Lumped-Partial Covariances

The lumped-partial covariance format, allows the evaluator to specify a group of nuclear reactions and to give the uncertainty only in the sum of the cross sections for that group of reactions. One can, for example, replace 30 or 40 discrete-level inelastic cross sections with 5 or 6 lumped cross sections when constructing the covariance files. Because the volume of the covariance data varies, in general, as the square of the number of reactions, this lumping can greatly reduce the size of the files.

The first ENDF evaluation to employ the lumped-partial format was P. Young's evaluation [53, 54] for ^{7}Li (ENDF/B-V, Rev. 2). All covariance data for this evaluation have been successfully processed into multigroup form using ERRORR. The covariances for `MT854` (a single real level with an excitation energy of 4.63 MeV) with `MT855` (6 lumped pseudo-levels, with excitation energies ranging from 4.75 MeV to 6.75 MeV) have been plotted in Fig. 62. The large negative correlations along the diagonal result from the fact that, below 10 MeV, these inelastic reactions are the major contributors to the relatively well-known tritium production cross section. An upward variation in one reaction at a given energy must be accompanied by a downward change in the other reaction. As shown in the plot, the magnitude of this negative correlation diminishes at higher energies, as other reactions begin to contribute significantly to the tritium-production cross section. Plots of this type, prepared using ERRORR and COVR, have proved to be useful tools in the validation of the covariance files of new evaluations [55].

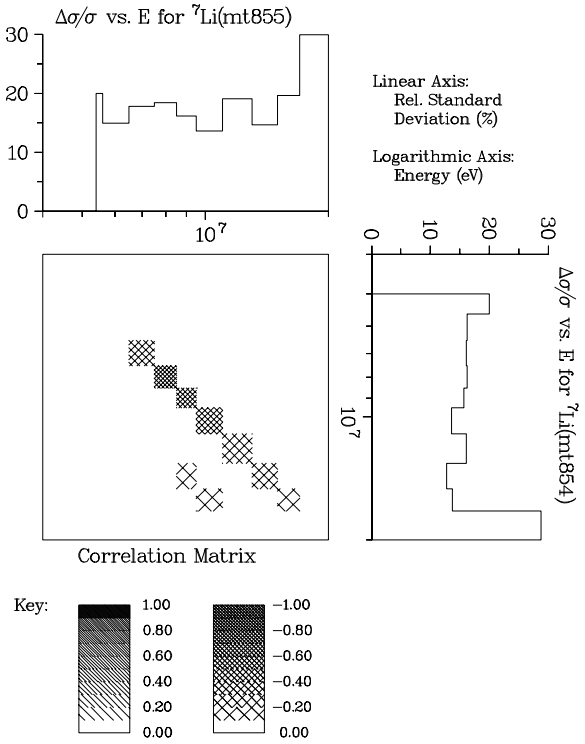

FIG. 62: Covariance data for ^{7}Li (MT854) with ^{7}Li (MT855).

M. Running ERRORR

As usual, the user input specifications for ERRORR can be found in the source code, the NJOY manuals, or on the web. Details on many of the input parameters are provided in the text below. Actual NJOY input for a few example problems is provided for further clarification. The first example is the simple input used to produce Fig. 55.

```
 1. errorr
 2. 31 32 0 33/
 3. 525 3 9 1 1/
 4. 0 0./
 5. 0 33/
 6. covr
 7. 33 0 34/
 8. 1/
 9. /
10. /
11. 525/
12. viewr
13. 34 35/
14. stop
```

The line numbers are for reference and are not part of the input. Identical versions of the ENDF/B-VII file for B-10 can be copied to `tape31` and `tape32`. The ERRORR output will appear on `tape33`. In this input, that file is passed on to COVR to generate the plot information. Line 3 defines the material to be processed (525, or B-10), the group structure, the weight function, the listing option, and requests relative covariances. Line 4 turns off printing for the group averaging and requests zero temperature. Line 5 asks the program to calculate the MT numbers to be processed and to work with File 33. The COVR and VIEWR input in lines 6 through 13 act to prepare the graph as seen in Fig. 55.

The following paragraphs give some details on how the input parameters are used.

`npend, ngout` ... The user must supply either a PENDF file on unit `npend` or a GENDF file on unit `ngout`. If present, the `npend` tape should contain pointwise (resonance-reconstructed, but not multigrouped) data and is normally produced with the RECONR module. If the resonance region is of no interest, a simple copy of the ENDF tape will suffice for this purpose. The GENDF tape, if present, should contain multigroup cross sections and/or $\overline{\nu}$ values, produced by the GROUPR module, for all reactions for which multigroup covariances are needed. Group cross sections need not be in the same group structure as the requested output co-

variances, although if the input group structure is much coarser than the output structure, rather crude approximations will be made in deriving an effective set of fine-group cross sections and fluxes from the coarse input data. Certain types of covariance calculations related to fission data or the computation of cross-material covariances require the user to supply a GENDF tape on `ngout` (that is, `npend` must be zero).

`nin, nout` ... Input and output covariance tapes in the user's group structure may either be both formatted or both binary. Although ERRORR lacks an explicit multimaterial loop, the same effect can be achieved by executing a series of ERRORR runs, with the output of one run becoming the input of the next. Only two unit numbers need be employed, with the data going back and forth between them until the multimaterial library is complete. To save time, binary files should be used for this purpose. MODER can be used to convert the final output tape from binary to formatted form, if desired.

`nstan` ... This tape is needed if covariances are requested for a reaction pair that is related by ratio measurements. The ENDF subsection for such a pair will contain an NC-type sub-subsection with `LTY`=1, 2, or 3.

`iread` ... The list of reactions for which covariances are produced is constructed in one of several ways, depending on the value of `iread`. If `iread=0`, which is the generally recommended choice, the reaction list is assumed to be identical to the list of sections contained in the ENDF/B covariance file (see `mfcov` below). Although `iread=0` is the most convenient option for constructing the list of covariance reactions, it leads to a problem if some covariance reactions have thresholds above the highest group boundary of the user's structure. These reactions will have zero cross sections in all groups and thus will be omitted from `ngout`. To avoid this difficulty, ERRORR automatically resets the user's highest group boundary to 20 MeV whenever `iread=0`. By judicious choice of weighting functions, one can minimize the effect of this resetting on the other cross sections and their covariances.

If `iread=1`, only the covariance reactions specifically named by the user will be included in the reaction list. For certain applications, this option can save considerable execution time. However, to take advantage of this option, the user must examine the ENDF evaluation visually to determine which reactions are derived from other reactions in each energy region. This information is contained in NC-type sub-subsections with `LTY=0`. Incorrect results will be obtained if one requests processing for a given reaction without processing all of the reactions from which the given reaction is derived in the energy region spanned by the output group structure. The inclusion of those reactions may require, in turn, additional inclusions as well. In addition, the user must extract from the file (and enter into the input) the appropriate derivation coefficients and the energy ranges over which they apply. An example of the required input is presented below in the discussion of the parameters `nmt`, `nek`, `mts`, `ek`, and `akxy`.

If `iread=0` or `1`, a complete set of covariance matrices is written. That is, covariance matrices are produced for every reaction-pair combination that can be formed from the given reaction list. For those reaction pairs where the evaluator has not specified the covariances, the output matrix contains only zeros.

In the evaluation containing cross-material covariances, subsections involving the other materials will be ignored if `iread=0` or `1`. These subsections can be selectively processed by specifying `iread=2`. If `iread=2`, the reaction list is initially constructed in the same way as for `iread=0`. This list is then supplemented with a list of extra reactions ($\mathrm{MAT1}_k$, $\mathrm{MT1}_k$) (where ($\mathrm{MAT1}_k \neq \mathrm{MATD}$), specified by the user. The output in this case contains matrices for all of the (MATD, MT_i; MATD, MT_j) combinations, as before, plus matrices for all (MATD, MT_i; $\mathrm{MAT1}_k$, $\mathrm{MT1}_k$) combinations. However, covariances among the extra reactions, for example ($\mathrm{MAT1}_k$, $\mathrm{MT1}_k$; $\mathrm{MAT1}_m$, $\mathrm{MT1}_m$), are not computed. Input for an example problem that illustrates the use of `iread=2` is given below in the discussion of `mfcov`. That discussion also includes an illustration of the set of MAT/MT combinations that are processed when `iread=2`.

`mfcov` ... This parameter is used to specify whether covariances of fission $\overline{\nu}$ (`mfcov=31`), cross sections (`mfcov=33 or 40`), angular distributions (`mfcov=34`), or energy distributions (`mfcov=35`) are needed. If `mfcov=31`, `35`, or `40`, it is necessary to supply multigrouped $\overline{\nu}$ values, cross sections, or secondary energy distributions on a GENDF file (that is, `npend` must be 0); the sample problem below shows the calculational sequence required. `mfcov=32` is reserved for a planned future capability to compute uncertainties in self-shielded cross sections. If `mfcov=33`, the contribution of the uncertainty in individual resonance parameters (File 32) is included in the calculated uncertainty in infinite-dilution group cross sections. Given below is the complete NJOY input for a calculation of a multigroup $\overline{\nu}$ covariance library for ^{238}U, including five cross-material reactions.

```
moder / mount 515, 516, and 555 on units 20, 21, and 22.
1 -23
'ENDF/B-V NUBAR COVARIANCE MATERIALS'/
20 1380
20 1381
21 1390
22 1395
22 1398
20 1399
0/
moder / copy ENDF for use as a PENDF.
-23 -24
groupr / prepare GENDF with multigrouped nubars.
-23 -24 0 25
1380 3 0 3 0 1 1 0
'BIG3 + 2 NUBAR'/
0.
1.e10
3 452 'TOTAL NUBAR'/
```

TABLE VII: Reaction Pairs in ^{238}U Library.

	a	b	c	d	e	f	g	h	i
a = (1398,452)	X	X	X	X	X	X	X	X	X
b = (1398,455)		X	X	X	X	X	X	X	X
c = (1398,456)			X	X	X	X	X	X	X
d = (1380,452)				–	–	–	–	–	–
e = (1381,452)					–	–	–	–	–
f = (1390,452)						–	–	–	–
g = (1395,452)							–	–	–
h = (1398,452)								–	–
i = (1399,452)									–

```
0/
1381
3 452 'TOTAL NUBAR'/
0/
1390
3 452 'TOTAL NUBAR'/
0/
1395
3 452 'TOTAL NUBAR'/
0/
1398
3 452 'TOTAL NUBAR'/
3 455 'DELAYED NUBAR'/
3 456 'PROMPT NUBAR'/
0/
1399
3 452 'TOTAL NUBAR'/
0/
0/
errorr / prepare multigroup nubar covariance library.
-23 0 25 26/
1398 19 1 1
2 31
1380 452
1381 452
1390 452
1395 452
1399 452
0/
1
1.e7 1.7e7
stop
```

Note the use of the MODER module to prepare a special ENDF tape containing selected evaluations from other ENDF tapes. Because only high-energy $\overline{\nu}$ data are requested (10–17 MeV), it is unnecessary to use RECONR to prepare a PENDF file for GROUPR. A copy of the ENDF file is used instead.

Following the prescription given in the discussion of the `iread=2` option, with this particular input, covariance matrices will be produced for those reaction pairs marked with an X in Table VII. The remaining non-redundant possibilities (dashes) can be filled in with successive ERRORR runs with `MATD=1380`, `1381`, etc.

`nwt`, `nek`, `mts`, `ek`, `akxy` ... For Version 5, if `iread=1`, the user specifies a subset of the evaluator's covariance reactions (that is, a subset of the sections of `mfcov`), as the particular set of reactions for which processing is requested. These `nmt` desired reactions are entered in the array `mts`. As mentioned in the discussion of `iread` above, one must also include in the `mts` array all reactions from which the desired reactions are derived.

Another requirement of the `iread=1` option is that the actual derivation coefficients [called `akxy` in the code, but called $C_I(a,c)$ in the discussion above] must be entered in the ERRORR input for `nek` energy ranges spanning the output group structure. Adjacent ENDF derivation ranges may be merged into a single range if the derivation-coefficient matrix for the `nmt` explicitly requested reactions is the same in each of the adjacent ranges. The ordering of the $C_I(a,c)$ data is as follows: on one line of input the coefficients are specified for a fixed a-value and for all c-values ranging from 1 to `nmt`. One such line is given for each a-value. Finally, there is an outer loop over the `k` energy ranges. An example of the input that is required for `iread=1` is given below for the case of ENDF/B-V carbon `MAT1306`. It will be necessary to examine File 33 of the evaluation (which is available on the ENDF/B-V standards file, Tape 511), in order to understand the details of this example.

```
moder
20- 21
moder / copy ENDF for use as a PENDF.
-21 -22/
errorr
-21 -22 0 -23/
1306 3 1/
2 0 0
1 33
7 3
1 2 4 102 103 104 107
1.e-5 2e6 4.812e6 2e7
0 1 0 1 0 0 0
0 1 0 0 0 0 0
0 0 1 0 0 0 0
0 0 0 1 0 0 0
0 0 0 0 1 0 0
0 0 0 0 0 1 0
0 0 0 0 0 0 1
1 0 0 0 0 0 0
0 1 0 0 0 0 0
0 0 1 0 0 0 0
0 0 0 1 0 0 0
0 0 0 0 1 0 0
0 0 0 0 0 1 0
0 0 0 0 0 0 1
1 0 0 0 0 0 0
0 1 0 0 0 0 0
1 -1 0 -1 -1 -1 -1
0 0 0 1 0 0 0
0 0 0 0 1 0 0
0 0 0 0 0 1 0
0 0 0 0 0 0 1
stop
```

Note that here, even if the user wanted to process only MT=2 and MT=4, for example, it is nevertheless necessary to include MT=1, MT=102, MT=103, MT=104, and MT=107, because the `ign=3` group structure extends to 17.0 MeV, and above 4.812 MeV, MT=4 is derived

from the relation

$$\sigma_4 = \sigma_1 - \sigma_2 - \sigma_{102} - \sigma_{104} - \sigma_{107} \ . \qquad (330)$$

See the corresponding $C_I(a,c)$ matrix in the input above (the last 7 lines before `STOP`).

`matb`, `mtb`, `matc`, `mtc` ... Card 11 provides a capability to remap all references to a given standards reaction `(MATB,MTB)` appearing in NC-type sub-subsections with `LTY=1`, `2`, or `3` into references to a different reaction `(MATC,MTC)`. This facility is useful if the evaluation `(MATB,MTB)` is for some reason unavailable. Up to 5 such standards can be redefined. As explained in the note on the input instructions at the beginning of this section, Card 11 is also used [if `MATB(1)` and `MTB(1)` are negative] to process covariances between two distinct reactions, both measured relative to a common standards reaction. This was referred to as the `LTY=4` case above. A separate ERRORR run is required for each requested `LTY=4` reaction pair.

This second use of Card 11 is illustrated in the sample NJOY input listed below, which is the input used to generate and plot the data shown in Fig. 61. The input to the COVR module contained in the sample is explained in the following section.

```
mount T562 on tape30
mount T563 on tape40
mount T560 on tape50

moder
1 -31
'U235 FROM T562'/
30 1395
0/
moder
1 -21
'AM241 FROM T560 AND PU239 FROM T563'/
50 1361/ AM241
40 1399/ PU239
0/
reconr
-21 -22
'10 PERCENT PENDF FOR AM241 AND PU239'/
1361/
.1/
1399/
.1/
0/
GROUPR
-21 -22 0 -24
1361 3 0 2 0 1 1 0
'30 GROUP XSECS FOR AM241 AND PU239'/
0
1e10
3 1/
3 2/
3 4/
3 16/
3 17/
3 18/
3 102/
0/
1399
3 18/
3 102/
0/
0/
errorr
-21 0 -24 -25 0 -31
1361 3 1 1
0 33
0/
errorr
-21 0 -24 -28 -25 -31
1399 3 1 1
2 33
1395 18
0/
-1395 -18 1361 18
0/
covr
-28/
0 0 0 8e4
1 1 0 0 2
1399 18 1361 18
stop
```

The input for a File 34 case preparing covariances for an angular distribution is shown below. This is the input that made Fig. 57 shown above. This deck runs a job sequence that includes MODER (ASCII-to-binary conversion), RECONR and BROADR (cross section reconstruction to 300K), GROUPR (unweighted, infinitely dilute multigroup cross sections plus average $\bar{\mu}$), ERRORR (specifying `mfcov=34` for $\bar{\mu}$ covariance processing), and COVR and VIEWR (for plot generation and conversion to Postscript format). The input on `tape20` must be the JENDL-3.3 evaluation for U-238.

```
moder
 20 -21 /
reconr
 -21 -22 /
 'processing jendl-3.3 u-238.'
 9237 0 0 /
 0.001 /
 0 /
broadr
 -21 -22 -23 /
 9237 1 0 0 0 /
 0.001 /
 300. /
 0 /
groupr
 -21 -23 0 91 /
 9237 3 0 2 1 1 1 0 /
 'test'
 300. /
 1.0e10 /
 3 /
 3 251 'mubar' /
 0 /
 0 /
--
-- process mf34
errorr
 -21 0 91 27 0 0 /
 9237 3 2 1 1 /
 0 34 1 1 -1 /
--
-- make mf34 plot file.
covr
 27 0 37 /
```

```
 1 /
 /
 /
 9237 /
--
-- make mf34 postscript file.
viewr
 37 47 /
stop
```

The input for the File 35 case that produced the plot of secondary energy covariances seen in Fig. 58 is given below. This is a problem taken from the NJOY test suite that uses a fictitious input file for ^{252}Cf. The GROUPR and ERRORR inputs use a user-defined group structure that corresponds to the energies listed in File 1 of the evaluation with the uncertainties provided by the evaluator. This makes it easy to compare the results with the evaluator's intent.

```
--
-- Copy ascii input to binary.
moder
 20 -21 /
--
-- Resonance reconstruction, to 0.1%.
reconr
 -21 -22 /
 'processing e70 252Cf with decay mf5/mt18 & mf35/mt18'
 9999 0 0 /
 0.001 /
 0 /
--
-- Doppler broaden to 300K.
broadr
 -21 -22 -23 /
 9999 1 0 0 0 /
 0.001 /
 300. /
 0 /
--
-- Group average, 300K with mf35 group structure.
--  - All file 3 cross sections plus fission spectrum.
groupr
 -21 -23 0 91 /
 9999 1 0 2 1 1 1 1 /
 'test'
 300. /
 1.0e10 /
 71 / # of groups, energy boundaries follow:
 1.000000-5 1.500000+4 3.500000+4 5.500000+4 7.500000+4
 9.500000+4 1.150000+5 1.350000+5 1.650000+5 1.950000+5
 2.250000+5 2.550000+5 3.050000+5 3.550000+5 4.050000+5
 4.550000+5 5.050000+5 5.550000+5 6.050000+5 6.550000+5
 7.050000+5 7.550000+5 8.050000+5 8.550000+5 9.050000+5
 9.550000+5 1.050000+6 1.150000+6 1.250000+6 1.350000+6
 1.450000+6 1.550000+6 1.650000+6 1.750000+6 1.850000+6
 1.950000+6 2.150000+6 2.350000+6 2.550000+6 2.750000+6
 2.950000+6 3.250000+6 3.550000+6 3.850000+6 4.150000+6
 4.450000+6 4.750000+6 5.050000+6 5.550000+6 6.050000+6
 6.550000+6 7.050000+6 7.550000+6 8.050000+6 8.550000+6
 9.050000+6 9.550000+6 1.005000+7 1.055000+7 1.105000+7
 1.155000+7 1.205000+7 1.255000+7 1.305000+7 1.355000+7
 1.405000+7 1.460000+7 1.590000+7 1.690000+7 1.790000+7
 1.910000+7 2.000000+7
 3 /
 5 18  'chi' /
 0 /
 0 /
--
-- ERRORJ, mf35
errorr
 20 0 91 28 0 0 /
 9999 1 2 1 1 /
 0 35 1 1 -1 1.23e6 /
 71 / # of groups, energy boundaries follow:
 1.000000-5 1.500000+4 3.500000+4 5.500000+4 7.500000+4
 9.500000+4 1.150000+5 1.350000+5 1.650000+5 1.950000+5
 2.250000+5 2.550000+5 3.050000+5 3.550000+5 4.050000+5
 4.550000+5 5.050000+5 5.550000+5 6.050000+5 6.550000+5
 7.050000+5 7.550000+5 8.050000+5 8.550000+5 9.050000+5
 9.550000+5 1.050000+6 1.150000+6 1.250000+6 1.350000+6
 1.450000+6 1.550000+6 1.650000+6 1.750000+6 1.850000+6
 1.950000+6 2.150000+6 2.350000+6 2.550000+6 2.750000+6
 2.950000+6 3.250000+6 3.550000+6 3.850000+6 4.150000+6
 4.450000+6 4.750000+6 5.050000+6 5.550000+6 6.050000+6
 6.550000+6 7.050000+6 7.550000+6 8.050000+6 8.550000+6
 9.050000+6 9.550000+6 1.005000+7 1.055000+7 1.105000+7
 1.155000+7 1.205000+7 1.255000+7 1.305000+7 1.355000+7
 1.405000+7 1.460000+7 1.590000+7 1.690000+7 1.790000+7
 1.910000+7 2.000000+7
--
-- make plot file.
covr
 28 0 38 /
 1 /
 1.e3 /
 /
 9999 /
--
-- make postscript file.
viewr
 38 39 /
stop
```

The final example gives the input used to prepare Fig. 59 showing the uncertainties in radionuclide production for ^{93}Nb. This is a fairly elaborate input deck, as it include GROUPR/ERRORR/COVR/VIEWR processing for both File 3 and File 10.

```
--
-- Extract/convert neutron evaluated data
moder
 1 -21
 '41-Nb-93 from e70'/
 20 4125
 0/
--
-- Make an ASCII copy
moder
 -21 41 /
--
-- Reconstruct xs
reconr
 41 42
 'pendf for 41-Nb-93'/
 4125 2/
 0.001 0. 0.003/
 '41-Nb-93 from e70'/
 'Processed with NJOY'/
 0/
--
-- Doppler broaden xs
broadr
 41 42 43
 4125 1 0 0 0./
 0.001 2.0e6 0.003/
```

```
 300.
 0/
--
-- groupr, process all file 3 reactions:
groupr
 41 43 0 44 /
 4125 1 0 6 0 1 1 1 /
 '41-Nb-93 from e70 with NJOY'/
 300. /
 1.e10 /
 21 /
 1.e-5 0.625 10.7 100.0 1.e3 1.e4 1.e5 1.e6
 2.e6 3.e6 4.e6 5.e6 6.e6 7.e6 8.e6 9.e6 10.e6
 12.e6 14.e6 16.e6 18.e6 20.e6 /
 3 /
 0/
 0/
--
-- errorj, process all file 3 reaction uncertainties:
errorr
 41 43 44 45/
 4125 1 6 1 1 /
 0 33 /
 21 /
 1.e-5 0.625 10.7 100.0 1.e3 1.e4 1.e5 1.e6
 2.e6 3.e6 4.e6 5.e6 6.e6 7.e6 8.e6 9.e6 10.e6
 12.e6 14.e6 16.e6 18.e6 20.e6 /
--
-- create plot of mf33 uncertainty data
covr
 45 0 55 /
 1 /
 1.e-1 /
 /
 4125 /
viewr
 55 65 /
--
-- groupr, process all file 10 reactions:
groupr
 41 43 0 46 /
 4125 1 0 6 0 1 1 1 /
 '41-Nb-93 from e70 with NJOY'/
 300. /
 1.e10 /
 21 /
 1.e-5 0.625 10.7 100.0 1.e3 1.e4 1.e5 1.e6
 2.e6 3.e6 4.e6 5.e6 6.e6 7.e6 8.e6 9.e6 10.e6
 12.e6 14.e6 16.e6 18.e6 20.e6 /
 10 /
 0/
 0/
--
-- errorj, process all file 10 reaction uncertainties:
errorr
 41 43 46 47/
 4125 1 6 1 1 /
 0 40 /
 21 /
 1.e-5 0.625 10.7 100.0 1.e3 1.e4 1.e5 1.e6
 2.e6 3.e6 4.e6 5.e6 6.e6 7.e6 8.e6 9.e6 10.e6
 12.e6 14.e6 16.e6 18.e6 20.e6 /
--
-- create plot of mf40 uncertainty data
covr
 47 0 57 /
 1 /
 1.e3 /
 /
 4125 /
viewr
 57 67 /
stop
```

N. ERRORR Output File Specification

The results from ERRORR are written to an output file on unit `nout`, provided that the input parameter `nout` is nonzero. This file contains the user's group structure, the multigroup cross sections, and either absolute (`irelco=0`) or relative (`irelco=1`) covariances. If `nout` is positive, then a formatted (card-image) file is written, and, if negative, a binary file is written. In either case, the output is written with the standard NJOY I/O utilities and the result is an 80-column ENDF-like structure with six 11-character data fields plus 14 characters to define the 4-digit MATN number, the 2-digit MF number, the 3-digit MT number and a one to five digit sequence number per line. As is the case for ENDF, PENDF, and GENDF files, ERRORR output files can be converted from binary to formatted form (and vice versa) by using the MODER module. A detailed description of the ERRORR output file format follows.

The initial record is a "tapeid" record, consisting of a 66-character comment field, unity for the tape number,which is written in the MATN field, and zero for the remaining fields. The TAPEID comment is copied from the GROUPR input tape, when available, or is blank. This is followed by an MF=1, MT=451 section that contains a CONT record plus a single LIST record. The CONT record contains the standard values of 1000*Z+A and AWR in the C1 and C2 fields. In addition the N1 field is set to -11, -12 or -14. We use -11 when the subsequent covariance data are from MF=31, 33 or 34; we use -12 when the data are from MF=35; and we use -14 when they are from MF40. In general, this flag identifies this as an ERRORR output file, similar to GROUPR's output file containing -1 in this position. The LIST record contains the temperature in degrees Kelvin in the C1 field (taken from the input GROUPR tape or the user input), the number of multigroups in the L1 field, and the number of energies in the N1 field. This is followed by the N1 energy values. As with GROUPR, ERRORR continues to follow the ENDF convention of ordering the energy boundaries from low to high energy. SEND and FEND records end this portion of the file.

This is followed by one or more LIST records with multigroup data from File 3 or File 5. The generic form of the LIST record includes ZA in the C1 field and the number of multigroups in the N1 field, followed by N1 data values. This differs from the GROUPR format in that here NJOY writes values for all multigroups, including zero for multgroups whose energy boundaries fall below cross section threshold energies. For ERRORR jobs with `mfcov=31` the specific MT fields include any of 452 (total $\bar{\nu}$), 455 (delayed $\bar{\nu}$) and 456 (prompt $\bar{\nu}$) from the original ENDF input tape. When MFCOV=33 the MT fields in-

clude all those found in File 3 of the ENDF input tape. For `mfcov=34` the average $\bar{\mu}$ data, MF=3 MT=251, are written; for `mfcov=35` the MF=5, MT=18 prompt fission spectrum multigroup values are written. For `mfcov=40`, the multigroup radioactive production cross sections derived from the underlying MF8 and MF10 data on the input ENDF tape are written to the appropriate MT section of MF3. Multiple MT sections are separated by a SEND record and both SEND and FEND records follow the data from the last MT.

The next portion of the ERRORR output file, with MF=MFCOV, contains the covariance data, now collapsed to the user's multigroup structure. For each unique MT value, the generic output file structure includes two CONT records and a series of LIST records that define the specific covariance matrix elements. Specifically, for each MT, the initial CONT record contains ZA in the C1 field and an integer number in the N2 field. This N2 value identifies how many sets of covariance data (MT-MTi pairs) follow for this MT. A second CONT record follows containing MATNi in the L1 field and MTi in the L2 field plus an integer flag in the N2 field for the number of LIST records that follow. A non-zero value for MATNi indicates that this is a cross-material covariance, a rarely used but perfectly legal representation within the ENDF format. It is much more common for MATNi to equal MATN and so the subsequent covariance matrix is given for a specific MATN, MTi-MT reaction pair. In this case, MATNi in the L1 field is set to zero and the equalite MATNi=MATN is understood. The value of the N2 integer flag must be at least one and less than or equal to the number of multigroups. Each list record contains a portion of the complete multigroup covariance matrix for this MT-MTi reaction pair. The L1, L2, N1 and N2 values in the initial line of the LIST record define the location of these data within the matrix. The specific values in the L1, L2, N1 and N2 fields define (i) the number of matrix elements being read in this LIST record; (ii) the starting column in the covariance matrix for these L1 values; (iii) the standard ENDF parameter that specifies the number of values in a LIST record (which is redundant with the L1 datum); and (iv) the covariance matrix row, respectively. All other elements in the matrix are zero. This triad of CONT, CONT and multiple LISTs is repeated for each MTi-MT pair. As usual, each MT section is terminated with a SEND record.

This pattern is repeated for the next MT value. Note that it is an unnecessary redundancy to write the MTi-MT and MT-MTi matrices. Therefore, within a given MT section, matrices are only defined for MTi≥MT. A standard FEND record is written after the final MT SEND record. Further discussion of ERRORR's output format is provided in the NJOY manuals.

XII. COVR

The COVR module of NJOY is an editing module that post-processes the output of ERRORR in a manner analogous to the way MATXSR and DTFR post-process the output of GROUPR. COVR performs two quite separate functions using the multigroup covariance file from ERRORR as input. First, it can prepare a new covariance library in a highly compressed card-image format, which is suitable for use as input to sensitivity analysis programs [56, 57]. (Data in this form can also be copied to the system output file to obtain a compact printed summary of an ERRORR run without using the sometimes bulky long-print option in ERRORR.) Such a summary can also provide information on standard deviations and correlation coefficients, neither of which are printed by ERRORR.) The second main function of COVR is to produce publication-quality plots [55, 58] of the multigroup covariance information.

A. Production of Boxer-Format Libraries

As discussed in the ERRORR section, the output file of that module contains the group structure, cross sections, and either absolute `(irelco=0)` or relative `(irelco=1)` covariances for one or more materials. If the COVR user-input parameter `nout` is greater than zero, COVR reads an ERRORR output file from unit `nin` and produces a new multigroup covariance library on unit `nout`. COVR performs only sorting and reformatting operations on the ERRORR data.

In COVR, as well as ERRORR, the ENDF energy ordering is followed. That is, low group indices correspond to low energies, high indices to high energies.

The material and reaction coverage of COVR post-processing is determined by a set of reaction pairs `(mat,mt;mat1,mt1)` supplied by the code user (see input Card 4 in the input instructions and the corresponding discussion). At the beginning of the output library, the group structure is given. Then, for each specified reaction pair, the output library contains either a covariance matrix or a correlation matrix, depending on the output option `(matype)` selected. In the case that covariances are requested `(matype=3)`, the type of data on `nout` (absolute *vs* relative) is governed by the covariance type present on `nin`. In addition, whenever `mat1=mat` and `mt1=mt`, the group-cross-section vector and the (absolute/relative) standard-deviation vector for that particular reaction are written to `nout` just before the matrix itself. All data (group structure, cross sections, standard deviations, and matrices) are written to `nout` using a highly compressed, card-image format.

The design of this format, called the Boxer format, proceeds from a simple fact: as discussed in the ERRORR section of this manual, most of the ENDF covariance formats define certain rectangular regions (boxes) in energy "space," over which the relative covariance is con-

stant. (The ENDF format allowing a constant absolute covariance is only rarely used.) The coordinate axes of the two-dimensional energy space in question are E_x and E_y, where x and y indicate the particular reaction pair to which the ENDF covariances apply. Because of this feature of the basic evaluations, one expects that an element of a multigroup relative covariance matrix, derived from the ENDF data for a given reaction pair, frequently will be identical either to the element before it in the same row (E_x constant, E_y varying) or to the element above it in the same column (E_y constant, E_x varying). Thus, the Boxer format allows a combination of horizontal and vertical repeat operations.

Even though the ERRORR output format already suppresses zero covariances, very large data compression factors can be achieved in transforming data from the ERRORR format to the Boxer format. As one example, the ERRORR output file for a particular 137-group reactor-dosimetry library [59] contained 38 000 card images, while the corresponding COVR output file contained fewer than 1000 card images.

In the Boxer format, data are stored as a list of numerical data values (*e.g.*, relative covariances), together with a list of integers that control the loading of these data into the reconstructed array $C(i,j)$. A negative control integer, $-n$, indicates that the next value in the data list is to be loaded into the next n columns (j-locations) of the current row of $C(i,j)$. A positive integer m, on the other hand, means that, for the next m j-values, the value to be loaded is to be carried down from the row above,

$$C(i,j) = C(i-1,j)\,. \tag{331}$$

For the first row ($i = 1$), the row above is defined to be a row containing all zeros. As an additional compression feature of the format, one may indicate by a "flag" that the matrix $C(i,j)$ is symmetric; hence, explicit instructions are provided in the compressed data library for the reconstruction of only the upper right triangle of $C(i,j)$.

B. Generation of Plots

The COVR plot mode, which is requested by specifying `nout=0`, is used to generate publication-quality plots of multigroup covariance data from a card-image or binary ERRORR output file. Examples of plots produced by COVR can be seen in the ERRORR section. In addition to their usefulness in preparing publications, the plots have proved to be a useful tool for checking the reasonableness and mechanical correctness of new covariance evaluations. One can, for example, execute ERRORR and COVR in tandem, using the evaluator's energy grid as the user group structure (ERRORR input option `ign=19`). The output of such a run is a series of plots showing all important features of the covariance evaluation.

As can be seen in the examples mentioned above, each plot contains a shaded contour map of the correlation matrix. If black&white plots are needed, positive-correlation regions are shaded with parallel straight lines (hatching), while negative correlations are indicated by cross-hatching. When color is allowed, the positive correlations are represented by shades of green, and the negative correlations are represented by shades of red. The plots also contain two additional inset graphs giving the energy dependence of the associated standard deviation vectors. One of the vector plots is rotated by 90 degrees, so that the logarithmic energy grids for the vector plots can be aligned with the corresponding grids for the matrix plot. When `MAT=MAT1` and `MT=MT1`, we plot the cross section rather than repeating the standard deviation plot.

Just as in the library mode, the material and reaction coverage of the sequence of plots generated in the plot mode is determined by the reaction pairs specified by the user on input Card 4. Other input, specific to the generation of plots, is described in Sect.XII C.

C. Running COVR

The detailed input instructions for COVR can be found at the beginning of the module in the source code, in the NJOY manuals, or on the NJOY web site. Some examples of COVR input have already appeared in the section on ERRORR. For example, the following input prepares graphs of the output for the B-10 File 33 covariances. The results on `tape34` can be passed to VIEWR for conversion into Postscript format.

```
1. covr
2. 33 0 34/
3. 1/
4. /
5. /
6. 525/
```

Line 3 selects the color option for the plots. Line 4 defines the lowest energy of interest for the plots. The default of zero was taken—that is, no truncation of the energy range was wanted. Line 5 takes the default values for input card 3a; for example, relative covariances are desired, one case is to be run, legends are to appear on all plots, and so on. The final line specifies the MAT number for the material to be run (525 for B-10).

The following paragraphs provide some discussion of the input parameters.

`epmin` ... This parameter is used to eliminate uninteresting energy regions from the correlation and standard deviation plots, or to display high-energy regions with greater resolution.

`irelco` ... This parameter must match the value used in the ERRORR run that produced the covariances to be plotted.

`ncase` ... This is the number of cases, or occurrences of Card 4. See the discussion of input parameters `mat`,

`mt`, `mat1`, and `mt1` below. Presently, `ncase` is limited to 40. Problems larger than this can be run as a series of COVR jobs, each processing a batch of up to 40 cases. See also the parameter `nstart` below.

`noleg` ... "Legend" here refers to the gray-shading scale (key) as well as the figure caption. `noleg=1` is used as a rough-draft mode to display plots quickly.

`nstart` ... Unless `nstart=0`, the plots are assigned a sequential figure number, beginning at `nstart`, and a list of figures is drawn on the final plot frame.

`ndiv` ... One gray-shade step equals 0.20 in correlation magnitude if `ndiv=1`. Finer gradations are possible with `ndiv` greater than 1. The plots appearing in the ERRORR section were generated with `ndiv=2`.

`hlibdD` ... This is a 6-character string, normally containing the name of the output covariance library. It is written on the header records present at the beginning of each output data block.

`hdescr` ... This contains additional information, for example, on where and when the library was produced; it is also written on the data header records.

`mat, mt, mat1, mt1` ... The information contained in the output library or plot file is controlled by means of these parameters on input Card 4. If `mt` is positive, then a single covariance matrix for reaction (`mat,mt`) with reaction (`mat1,mt1`) will be read from `nin` and processed. On unit `nin` (and, in the library option, on unit `nout`), the rapidly varying, or column, index is the group index of (`mat1,mt1`), and the slowly varying, or row, index is the group index of (`mat,mt`). If `mat1` is different from `mat`, COVR will expect to find separate materials (produced by separate ERRORR runs) for both `mat` and `mat1` on `nin`. The `mat` numbers must occur on `nin` in ascending order. In the case of positive `mt`, the entries `mat1=0` and `mt1=0` are shorthand for `mat1=mat` and `mt1=mt`, respectively. If, on the other hand, `mt` is zero, negative, or defaulted, Card 4 becomes a kind of macroinstruction that is expanded by subroutine `expndo` into a request for many `mt-mt1` pairs, all with `mat1=mat`. If, for example, Card 4 contains the entry

```
MAT/
```

or, equivalently,

```
MAT 0 0 0
```

first the cross-section file, `MF=3`, for material `MAT` is read from the input covariance tape on unit `nin` to obtain the list of reactions present. Then, all possible reaction combinations `mt-mt1` are formed in ENDF order. Thus, for example, if the reactions present are `MT=1, 2, 3`, and `16`, then the behavior of the code is the same as if the following input were specified:

```
MAT 1 0 1
MAT 1 0 2
MAT 1 0 3
MAT 1 0 16
MAT 2 0 2
MAT 2 0 3
MAT 2 0 16
MAT 3 0 3
MAT 3 0 16
MAT 16 0 16  .
```

Because of the clear labor-saving advantage of this feature, especially if there are many reactions, enhancements have been added to permit its use in the additional situation in which many, but not all, combinations are desired. As discussed in some detail in the input instructions, it is possible to strip selected reactions out of the list before the `mt-mt1` combinations are formed. For example, if the reactions present are once again 1, 2, 3, and 16, then a Card 4 containing

```
MAT -3/
```

would produce the same output as the following three cards:

```
MAT 2 0 2
MAT 2 0 16
MAT 16 0 16.
```

The stripping of the higher inelastic levels, mentioned in the input instructions, is useful because ERRORR automatically resets the highest user-group energy to 20 MeV whenever `IREAD=0`, and this can result in the inclusion of unwanted high-threshold reactions on the ERRORR output file.

For one of several reasons, a requested reaction pair (`mat,mt; mat1,mt1`) may be absent from the COVR output. The usual reason that this occurs is that the normal `iread` option for the ERRORR module is `iread=0`, and this results in the generation of an output matrix for every possible reaction-pair combination. For those combinations that do not occur in the ENDF evaluation, the ERRORR output matrix contains only zeroes. These null matrices are omitted at the COVR output stage, in both the library and plot modes.

Additionally, in the plot mode, the correlations may be nonzero, but everywhere less than `0.2/ndiv` in absolute magnitude. In this case, the entire correlation plot would consist of a single blank region. These rather uninteresting plots having small correlations are also omitted from the plot file.

A final, similar category is the class of "empty" plots. Because parallel lines to achieve a gray effect when color is not used, there is clearly some lower limit, for a given value of the correlation magnitude, on the physical size of a region that can be sensibly shaded. COVR does not attempt to shade regions that are smaller than this limit. If a correlation matrix does contain some plottable data (magnitudes exceeding `0.2/ndiv`), but all plottable regions are smaller than the size limit discussed above, then an empty plot will be generated on the plot file. As

a convenience in discarding these empty plots at the time a report is produced, no caption is written on such plots and the figure number is not advanced. Also, these plots are omitted from the list of figures prepared at the end of a plot run. (See the discussion of user-input parameter `nstart`.)

In the library mode, each omission of a requested, but null, matrix is noted on the `output` file with an informative diagnostic. In the plot mode, a summary table is printed at the end of the run to identify all requested matrices that were omitted because they were null or small, as well as those that were plotted, but empty.

We next discuss the production of a particular COVR output library, both to illustrate the input and as a supplement to the general discussion of Boxer-format libraries. In the ERRORR section, we gave the complete NJOY input for producing a 7-reaction, 30-group covariance library in ERRORR output format for carbon. By appending the following lines to that input (just before the `stop` card), one can produce, in addition, a 2-reaction COVR output library in Boxer format.

```
covr
-23 24
3 3
'  LIB'
'MAT1306 COVR EXAMPLE'/
1306 2 0 2
1306 2 0 4
1306 4 0 4
```

The resulting library occupies only 49 lines and is listed in Fig. 63.

Note that the header card at the start of each data block contains an integer `itype`, specifying the type of data contained in the current block, a 12-character library name ("`LIB-A- 30`" in this case), 21 characters of user-supplied descriptive information, `mat, mt, mat1, mt1`, and a set of 7 integers. The meaning of the various values of `itype` is as follows: 0 = group boundaries, 1 = cross sections, 2 = standard deviations, 3 = covariances, and 4 = correlations. The library name is generated within COVR by adding either "-A-" (for a covariance library) or "-B-" (for a correlation library), together with the number of energy groups, to the user-supplied library name `hlibid`. (The ERRORR input option `ign=3` used in this example specifies a built-in 30-group structure). The final seven integers on the header card indicate the number and format of the data values, the number and format of the control integers *"m"* and *"-n"*, a data paging flag, and the dimensions of the reconstructed data array, $C(i, j)$.

XIII. LEAPR

This module is used to prepare the scattering law $S(\alpha, \beta)$, which describes thermal scattering from bound moderators, in the ENDF-6 format used by THERMR. The original ENDF thermal scattering data [29] were prepared by General Atomics (GA) using the GASKET code [60]. LEAPR is based on the British code LEAP+ADDELT originally written by McLatchie at Harwell [61], then implemented by Butland at Winfrith [62], and finally modified to work better for cold moderators as part of the Ph.D. Thesis of D. J. Picton [63]. A number of modifications and enhancements were made to the coding and the methods for preparing modern ENDF thermal evaluations [64]. Some of the cold-moderator work has been described elsewhere [65, 66].

A. Theory

The following discussion of the theories used in the the code is based on the original British documentation and the presentation in a standard reference [67].

a. Coherent and Incoherent Scattering. In practice, the scattering of neutrons from a system of N particles with a random distribution of spins or isotope types can be expressed as the sum of a coherent part and an incoherent part. The coherent scattering includes the effects from waves that are able to interfere with each other, and the incoherent part depends on a simple sum of noninterfering waves from all the N particles. (The spin correlations in ortho and para hydrogen violate the assumption of randomness, so liquid hydrogen does not fit into the model described here. A method for treating them will be described below.) The cross sections for coherent and incoherent scattering can be considered to be characteristic properties of the materials. As examples, the scattering from hydrogen is almost completely incoherent, and the scattering from carbon and oxygen is almost completely coherent.

Furthermore, the coherent and incoherent scattering include both elastic and inelastic parts. The elastic scattering takes place with no energy change. It should not be confused with the elastic scattering from a single particle that is familiar for higher neutron energies where the neutron loses energy; thermal elastic scattering can be considered to be scattering from the entire lattice, thus the effective mass of the target is very large, and the neutron does not lose energy in the scattering process. Thermal inelastic scattering results in an energy loss (gain) for the neutron with a corresponding excitation (deexcitation) of the target. The excitation may correspond to the production of one or more phonons in a crystalline material, to the production of rotations or vibrations in molecules, or to the initiation of atomic or molecular recoil motions in a liquid or gas.

In addition, the coherent inelastic part of the scattering contains both interference effects between waves scattered by different particles and direct terms. It turns out that the direct part for gases, liquids, and solids consisting of randomly oriented crystallites has approximately the same form as the incoherent term. The interference is usually neglected.

```
0    LIB-A- 30 MAT1306 COVR EXAMPLE  1306   2 1306   2  31 10  31  3   0  31   1
 1.390E-04 1.520E-01 4.140E-01 1.130E+00 3.060E+00 8.320E+00 2.260E+01 6.140E+01
 1.670E+02 4.540E+02 1.235E+03 3.350E+03 9.120E+03 2.480E+04 6.760E+04 1.840E+05
 3.030E+05 5.000E+05 8.230E+05 1.353E+06 1.738E+06 2.232E+06 2.865E+06 3.680E+06
 6.070E+06 7.790E+06 1.000E+07 1.200E+07 1.350E+07 1.500E+07 1.700E+07
 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
 -1 -1 -1 -1 -1
1    LIB-A- 30 MAT1306 COVR EXAMPLE  1306   2 1306   2  23 10  23  3   0  30   1
 4.739E+00 4.738E+00 4.735E+00 4.729E+00 4.712E+00 4.676E+00 4.579E+00 4.332E+00
 4.002E+00 3.619E+00 3.103E+00 2.477E+00 1.998E+00 1.820E+00 1.710E+00 2.256E+00
 1.447E+00 9.921E-01 8.223E-01 7.627E-01 8.631E-01 8.228E-01 8.845E-01
 -8 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
2    LIB-A- 30 MAT1306 COVR EXAMPLE  1306   2 1306   2  18 10  18  3   0  30   1
 2.000E-03 1.964E-03 4.583E-03 3.822E-03 4.583E-03 4.288E-03 3.730E-03 4.141E-03
 6.501E-03 1.011E-02 9.558E-03 9.981E-03 1.806E-02 1.854E-02 3.296E-02 3.760E-02
 4.255E-02 7.632E-02
 -9 -1 -5 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
3    LIB-A- 30 MAT1306 COVR EXAMPLE  1306   2 1306   2  43 10  58  4   0  30   0
 4.000E-06 2.797E-06 3.856E-06 6.317E-06 4.613E-06 2.407E-06 1.229E-06 2.100E-05
 1.534E-05 8.000E-06 4.087E-06 1.461E-05 1.366E-05 2.100E-05 1.838E-05 9.004E-06
 1.391E-05 8.934E-06 1.715E-05 4.447E-06 4.227E-05 4.373E-05 2.446E-05 1.980E-05
 1.223E-05 1.022E-04 6.116E-05 4.048E-05 2.500E-05 9.136E-05 4.572E-05 9.962E-05
 6.641E-05 6.308E-05 3.260E-04 1.614E-04 3.437E-04 1.086E-03 4.250E-04 1.413E-03
 7.052E-04 1.811E-03 5.825E-03
  -9  -1 224  -1  -5  -1  -4  -1   9  -5  -1  -4  -1  79  -1  -1  13  -1  13  -1
  -1  11  -1  -1  10  -1  -1   9  -1  -1  -1  -1  -6  -1  -1  -1  -6  -1  -1   6
  -1  -1  -1   4  -1  -1   4  -1   4  -1  -2   1  -1  -1   1  -1   1  -1
3    LIB-A- 30 MAT1306 COVR EXAMPLE  1306   2 1306   4  38 10  47  4   0  30  30
-1.106E-04-4.897E-05-3.047E-05-2.164E-05-2.630E-05-2.337E-05-2.192E-05-2.261E-04
-1.001E-04-6.228E-05-4.423E-05-5.376E-05-4.777E-05-4.481E-05-1.441E-03-2.659E-04
-1.571E-04-1.210E-03-1.305E-03-4.021E-04-1.131E-03-6.462E-04-8.562E-04-2.261E-04
-1.001E-04-6.228E-05-1.922E-03-9.140E-04-8.121E-04-7.520E-04-3.040E-03-1.348E-03
-1.517E-03-3.460E-03-4.423E-05-5.376E-05-4.777E-05-1.044E-02
 623  -1  -1  -1  -1  -1  -1  -1  23  -1  -1  -1  -1  -1  -1  -1  53  -1  -1  -1
  27  -1  -1  -1  27  -1  -1  -1  27  -1  -1  -1  -1  -1  -1  27  -1  -1  -1  28
  -1  -1  27  -1  -1  -1  -1
1    LIB-A- 30 MAT1306 COVR EXAMPLE  1306   4 1306   4   7 10   8  3   0  30   1
 6.091E-02 2.478E-01 3.301E-01 4.310E-01 4.013E-01 4.306E-01 4.935E-01
 23 -1 -1 -1 -1 -1 -1 -1
2    LIB-A- 30 MAT1306 COVR EXAMPLE  1306   4 1306   4   7 10   8  3   0  30   1
 2.499E-01 1.085E-01 9.724E-02 9.806E-02 1.358E-01 1.263E-01 2.177E-01
 23 -1 -1 -1 -1 -1 -1 -1
3    LIB-A- 30 MAT1306 COVR EXAMPLE  1306   4 1306   4  28 10  29  4   0  30   0
 6.245E-02 7.530E-03 3.823E-03 1.154E-03 1.338E-03 1.211E-03 1.155E-03 1.176E-02
 2.311E-03 5.938E-04 6.451E-04 5.843E-04 5.578E-04 9.455E-03 1.887E-03 4.735E-04
 4.295E-04 4.106E-04 9.617E-03 1.872E-03 1.670E-03 3.033E-04 1.844E-02 5.214E-03
 3.493E-04 1.596E-02 5.253E-03 4.741E-02
 437  -1  -1  -1  -1  -1  -1  -1  -1  -1  -1  -1  -1  -1  -1  -1  -1  -1  -1  -1
  -1  -1  -1  -1  -1  -1  -1  -1  -1
```

FIG. 63: Example of BOXER output.

Therefore, we can usually divide the thermal scattering cross section into three different parts:

- *Coherent elastic.* Important for crystalline solids like graphite or beryllium.
- *Incoherent elastic.* Important for hydrogenous solids like solid methane, polyethylene, and zirconium hydride.
- *Inelastic.* Important for all materials (this category includes both incoherent and coherent inelastic).

The absence of interference in incoherent scattering and the neglect of interference in coherent inelastic scattering allows us to construct thermal scattering laws for "hydrogen in water" or "hydrogen in solid methane" or "oxygen in beryllium oxide". However, this simplification is not possible in general for coherent elastic scattering in materials with more that one type of atom in the unit cell; for coherent elastic scattering, beryllium oxide must be considered as a unit.

b. Inelastic Scattering. It is shown in the standard references [67] that the double differential scattering cross section for thermal neutrons for gases, liquids, or solids consisting of randomly ordered microcrystals can be written as

$$\sigma(E\rightarrow E',\mu) = \frac{\sigma_b}{2kT}\sqrt{\frac{E'}{E}}\,\mathcal{S}(\alpha,\beta)\,, \tag{332}$$

where E and E' are the incident and secondary neutron energies in the laboratory system, μ is the cosine of the scattering angle in the laboratory, σ_b is the characteristic bound scattering cross section for the material, kT is the thermal energy in eV, and $\mathcal{S}$ is the asymmetric form of the scattering law. The scattering law depends on only two variables: the momentum transfer

$$\alpha = \frac{E' + E - 2\sqrt{E'E}\mu}{AkT}\,, \tag{333}$$

where A is the ratio of the scatterer mass to the neutron mass, and the energy transfer

$$\beta = \frac{E' - E}{kT}\,. \tag{334}$$

Note that β is positive for energy gain and negative for energy loss. Working in the incoherent approximation and the Gaussian approximation, the scattering law can be calculated using the "phonon expansion" method starting with the frequency spectrum of excitations in the system expressed as a function of β. For systems in thermal equilibrium, there is a relationship between upscatter and downscatter called "detail balance" that is a consequence of microscopic reversibility. It requires that

$$\mathcal{S}(\alpha,\beta) = \mathrm{e}^{-\beta}\mathcal{S}(\alpha,-\beta)\,. \tag{335}$$

Liquid hydrogen and deuterium violate this condition, as will be described below. In addition, the scattering law satisfies two other important constraints; namely, normalization,

$$\int \mathcal{S}(\alpha,\beta)\,d\beta = 1\,, \tag{336}$$

and the sum rule

$$\int \beta\,\mathcal{S}(\alpha,\beta)\,d\beta = -\alpha\,. \tag{337}$$

Actually, ENDF works with the so-called "symmetric" $S(\alpha,\beta)$,

$$S(\alpha,\beta) = \beta/2\,\mathcal{S}(\alpha,\beta) \tag{338}$$

which (except for liquid hydrogen or deuterium) satisfies the condition

$$S(\alpha,\beta) = S(\alpha,-\beta)\,. \tag{339}$$

Note that $\mathcal{S}(\alpha,-\beta)$ for positive β describes the downscatter side of the function, and because it is basically proportional to the cross section, it can be represented by reasonable numbers (say 10^{-8} to 1) for all β. The symmetric $S(\alpha,\beta)$, on the other hand, can easily be smaller than $\mathcal{S}(\alpha,-\beta)$ by factors like $\mathrm{e}^{-\beta/2} \sim \exp -80 \sim 10^{-35}$, which can cause trouble on short-word machines. This kind of numerical problem is even more severe for cold moderators, where dynamic ranges on the order of 10^{100} occur for $S(\alpha,\beta)$. (The user will have to use some caution reading this report, because the typographic symbols for S and script-S are very similar.) LEAPR works with $\mathcal{S}(\alpha,-\beta)$.

The next step is to decompose the frequency spectrum into a sum of simple spectra

$$\rho(\beta) = \sum_{j=1}^{K} \rho_j(\beta), \tag{340}$$

where the following possibilities are allowed:

$$\rho_j(\beta) = w_j\delta(\beta_j) \quad \text{discrete oscillator} \tag{341}$$
$$\rho_j(\beta) = \rho_s(\beta) \quad \text{solid-type spectrum} \tag{342}$$
$$\rho_j(\beta) = \rho_t(\beta) \quad \text{translational spectrum} \tag{343}$$

The solid-type spectrum must vary as β^2 as β goes to zero, and it must integrate to w_s, the weight for the solid-type law. The translational spectrum can be either a free-gas law or a diffusion-type spectrum represented with the approximation of Egelstaff and Schofield that will be discussed later. In either case, the spectrum must integrate to w_t, the translational weight. The sum of all the weights of the partial spectra must equal 1. Defining $\gamma_j(\hat{t})$ to be the value of γ appropriate for ρ_j, and $\mathcal{S}_j$ to be the corresponding partial scattering law, and using the convolution theorem for Fourier transforms, leads to a recursive formula for the scattering law:

$$\mathcal{S}(\alpha,\beta) = \mathcal{S}^{(K)}(\alpha,\beta), \tag{344}$$

where

$$\begin{aligned}\mathcal{S}^{(J)}(\alpha,\beta) &= \frac{1}{2\pi}\int \mathrm{e}^{i\beta\hat{t}}\prod_{j=1}^{J}\mathrm{e}^{-\gamma_j(\hat{t})}\,d\hat{t} \\ &= \int \mathcal{S}_J(\alpha,\beta')\,\mathcal{S}^{(J-1)}(\alpha,\beta-\beta')\,d\beta'\,.\end{aligned} \tag{345}$$

As an example of the use of this recursive procedure, consider a case where the desired frequency spectrum is a combination of ρ_s and two discrete oscillators. First, calculate $\mathcal{S}^{(1)}$=$\mathcal{S}_1$ using ρ_s. Then calculate $\mathcal{S}_2$ using $\rho(\beta_1)$, the distribution for the first discrete oscillator, and convolve $\mathcal{S}_2$ with $\mathcal{S}^{(1)}$ to obtain $\mathcal{S}^{(2)}$, the composite scattering law for the first two partial distributions. Repeat the process with the second discrete oscillator to obtain $\mathcal{S}^{(3)}$, which is equal to $\mathcal{S}(\alpha,\beta)$ for the full distribution.

c. The Phonon Expansion. Consider first $\gamma_s(\hat{t})$, the Gaussian function for solid-type frequency spectra. Expanding the time-dependent part of the exponential gives

$$\begin{aligned} e^{-\gamma_s(\hat{t})} &= e^{-\alpha\lambda_s} \sum_{n=0}^{\infty} \frac{1}{n!} \\ &\times \left[\alpha \int_{-\infty}^{\infty} P_s(\beta)\, e^{-\beta/2} e^{-i\beta\hat{t}}\, d\beta \right]^n , \quad (346) \end{aligned}$$

where λ_s is the Debye-Waller coefficient

$$\lambda_s = \int_{-\infty}^{\infty} P_s(\beta)\, e^{-\beta/2}\, d\beta\,. \quad (347)$$

The scattering function becomes

$$\begin{aligned} \mathcal{S}_s(\alpha,\beta) &= e^{-\alpha\lambda_s} \sum_{n=0}^{\infty} \frac{1}{n!} \alpha^n \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{i\beta t} \\ &\times \left[\int_{-\infty}^{\infty} P_s(\beta')\, e^{-\beta'/2}\, e^{-i\beta'\hat{t}}\, d\beta' \right]^n d\hat{t}\,. \quad (348) \end{aligned}$$

For convenience, define the quantity in the second line of this equation to be $\lambda_s^n \mathcal{T}_n(\beta)$. Then clearly,

$$\mathcal{S}_s(\alpha,\beta) = e^{-\alpha\lambda_s} \sum_{n=0}^{\infty} \frac{1}{n!} [\alpha\lambda_s]^n\, \mathcal{T}_n(\beta)\,, \quad (349)$$

where

$$\mathcal{T}_0(\beta) = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{i\beta\hat{t}}\, d\hat{t} = \delta(\beta)\,, \quad (350)$$

and

$$\begin{aligned} \mathcal{T}_1(\beta) &= \int_{-\infty}^{\infty} \frac{P_s(\beta')\, e^{-\beta'/2}}{\lambda_s} \\ &\quad \left\{ \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{i(\beta-\beta')\hat{t}}\, d\hat{t} \right\} d\beta' \\ &= \frac{P_s(\beta)\, e^{-\beta/2}}{\lambda_s}\,, \quad (351) \end{aligned}$$

In general,

$$\mathcal{T}_n(\beta) = \int_{-\infty}^{\infty} \mathcal{T}_1(\beta')\, \mathcal{T}_{n-1}(\beta-\beta')\, d\beta'\,. \quad (352)$$

The script-T functions obey the relationship $\mathcal{T}_n(\beta) = e^{-\beta}\mathcal{T}_n(-\beta)$. In addition, each of the $\mathcal{T}_n$ functions obeys the following normalization condition:

$$\int_{-\infty}^{\infty} \mathcal{T}_n(\beta)\, d\beta = 1\,. \quad (353)$$

It guarantees that Eq.(336) will be satisfied by the sum in Eq.(349). In LEAPR, the $\mathcal{T}_n(-\beta)$ functions are precomputed on the input β grid for n up to some specified maximum value, typically 100. It is then easy to compute the smooth part of $\mathcal{S}_s(\alpha,-\beta)$ for any sufficiently small value of α using Eq.(349). The corresponding values of $\mathcal{S}_s(\alpha,\beta)$ can then be obtained by multiplying by $e^{-\beta}$. The delta function arising from the "zero-phonon" term is carried forward separately. The normalization in Eq.(349) has better numerical properties than the one used in LEAP.

d. The Short-Collision-Time Approximation. For larger values of α, the phonon expansion requires too many terms. LEAPR uses the simple Short-Collision-Time (SCT) approximation from ENDF to extend the solid-type scattering law.

$$\begin{aligned} &\mathcal{S}_s(\alpha,-\beta) \\ &= \frac{1}{\sqrt{4\pi w_s \alpha \overline{T}_s/T}} \exp\left[-\frac{(w_s\alpha-\beta)^2}{w_s\alpha\overline{T}_s/T} \right], \quad (354) \end{aligned}$$

and

$$\mathcal{S}_s(\alpha,\beta) = \exp -\beta \mathcal{S}_s(\alpha,-\beta) \quad (355)$$

where β is positive, and where the effective temperature is given by

$$\overline{T}_s = \frac{T}{2w_s} \int \beta^2 P_s(\beta)\, \exp -\beta\, d\beta\,. \quad (356)$$

As above, w_s is the weight for the solid-type spectrum.

e. Diffusion and Free-Gas Translation. The neutron scattering from many important liquids, including water and liquid methane, can be represented using a solid-type spectrum of rotational and vibrational modes combined with a diffusion term. Egelstaff and Schofield have proposed an especially simple model for diffusion called the "effective width model". It has the advantage of having analytic forms for both $S(\alpha,\beta)$ and the associated frequency spectrum $\rho(\beta)$:

$$\begin{aligned} \mathcal{S}_t(\alpha,\beta) &= \frac{2cw_t\alpha}{\pi} \exp\left[2c^2 w_t \alpha - \beta/2\right] \\ &\quad \frac{\sqrt{c^2+.25}}{\sqrt{\beta^2+4c^2w_t^2\alpha^2}} \\ &\quad K_1\left\{ \sqrt{c^2+.25}\sqrt{\beta^2+4c^2w_t^2\alpha^2} \right\}, \quad (357) \end{aligned}$$

and

$$\begin{aligned} \rho(\beta) &= w_t \frac{4c}{\pi\beta} \sqrt{c^2+.25} \\ &\quad \sinh(\beta/2)\, K_1\left\{ \sqrt{c^2+.25}\,\beta \right\}. \quad (358) \end{aligned}$$

In these equations, $K_1(x)$ is a modified Bessel function of the second kind, and the translational weight w_t and the diffusion constant c are provided as inputs.

An alternative for the translational part of the distribution is the free-gas law. It is clearly appropriate for a gas of molecules, but it has also been used to represent

the translational component for liquid moderators like water [29]. The scattering law is given by

$$\mathcal{S}_t(\alpha,-\beta) = \frac{1}{\sqrt{4\pi w_t\alpha}}\,\exp\left[-\frac{(w_t\alpha-\beta)^2}{4w_t\alpha}\right], \qquad (359)$$

and

$$\mathcal{S}_t(\alpha,\beta) = \exp-\beta\mathcal{S}_t(\alpha,-\beta)\,, \qquad (360)$$

with β positive. The free-gas law is used in LEAPR if the diffusion coefficient c is input as zero.

In LEAPR, $\mathcal{S}_s(\alpha,\beta)$, the scattering law for the solid-type modes, is calculated using the phonon expansion as described above. The translational contribution $\mathcal{S}_t(\alpha,\beta)$ is then calculated using one of the formulas above on a β grid chosen to represent its shape fairly well. The combined scattering law is then obtained by convolution as follows:

$$\begin{aligned}\mathcal{S}(\alpha,\beta) &= \mathcal{S}_t(\alpha,\beta)\,\exp-\alpha\lambda_s \\ &+ \int \mathcal{S}_t(\alpha,\beta')\,\mathcal{S}_s(\alpha,\beta-\beta')\,d\beta'. \qquad (361)\end{aligned}$$

The first term arises from the delta function in Eq.(349), which isn't included in the numerical results for the phonon series calculation. The values for $\mathcal{S}_t(\beta)$ and $\mathcal{S}_s(\beta-\beta')$ are obtained from the precomputed functions by interpolation. The effective temperature for a combination of solid-type and translation modes is computed using

$$\overline{T}_s = \frac{w_tT + w_s\overline{T}_s}{w_t+w_s}\,. \qquad (362)$$

f. Discrete Oscillators. Polyatomic molecules normally contain a number of vibrational modes that can be represented as discrete oscillators. The distribution function for one oscillator is given by $w_i\delta(\beta_i)$, where w_i is the fractional weight for mode i, and β_i is the energy-transfer parameter computed from the mode's vibrational frequency. The corresponding scattering law is given by

$$\begin{aligned}\mathcal{S}_i(\alpha,\beta) &= \exp-\alpha\lambda_i \sum_{n=-\infty}^{\infty}\delta(\beta-n\beta_i) \\ &\quad I_n\left[\frac{\alpha w_i}{\beta_i\sinh(\beta_i/2)}\right]\exp-n\beta_i/2 \\ &= \sum_{n=-\infty}^{\infty} A_{in}(\alpha)\,\delta(\beta-n\beta_i)\,, \qquad (363)\end{aligned}$$

where

$$\lambda_i = w_i\frac{\coth(\beta_i/2)}{\beta_i}\,. \qquad (364)$$

The combination of a solid-type mode (s) with discrete oscillators (1) and (2) would give

$$\begin{aligned}\mathcal{S}^{(0)}(\alpha,\beta) &= \mathcal{S}_s(\alpha,\beta)\,, \qquad (365)\\ \mathcal{S}^{(1)}(\alpha,\beta) &= \int \mathcal{S}_1(\alpha,\beta')\,\mathcal{S}^{(0)}(\alpha,\beta-\beta')\,d\beta' \\ &= \sum_{n=-\infty}^{\infty} A_{1n}(\alpha)\,\mathcal{S}^{(0)}(\alpha,\beta-n\beta_1)\,, \qquad (366)\\ \mathcal{S}^{(2)}(\alpha,\beta) &= \int \mathcal{S}_2(\alpha,\beta')\,\mathcal{S}^{(1)}(\alpha,\beta-\beta')\,d\beta' \\ &= \sum_{m=-\infty}^{\infty} A_{2m}(\alpha)\sum_{n=-\infty}^{\infty} A_{1n}(\alpha) \\ &\quad \mathcal{S}^{(0)}(\alpha,\beta-n\beta_1-m\beta_2)\,. \qquad (367)\end{aligned}$$

This process can be continued through $\mathcal{S}^{(3)}(\alpha,\beta)$, $\mathcal{S}^{(4)}(\alpha,\beta)$, etc., until all the discrete oscillators have been included. The result has the form

$$\mathcal{S}(\alpha,\beta) = \sum_k W_k(\alpha)\,\mathcal{S}_s(\alpha,\beta-\beta_k)\,, \qquad (368)$$

where the β_k and the associated weights W_k are easily generated recursively using a procedure that throws out small weights at each step. The Debye-Waller λ for the combined modes is computed using

$$\lambda = \lambda_s + \sum_{i=1}^{N}\lambda_i\,. \qquad (369)$$

The effective temperature for the combined modes is given by

$$\overline{T}_s = w_tT + w_s\overline{T}_s + \sum_{i=1}^{N} w_i\frac{\beta_i}{2}\coth\left(\frac{\beta_i}{2}\right)T\,. \qquad (370)$$

If the starting-point scattering law $\mathcal{S}^{(0)}$ does not contain a translational contribution (true for hydrogenous solids like polyethylene and frozen methane), it is important to remember to include the effects of the "zero-phonon" term $\exp(-\alpha\lambda_s)\delta(\beta)$. The code does this by adding in triangular peaks with the proper areas and with their apexes at the β value closest to the β_k values. One of these peaks is at β=0. This peak is not put into the scattering law as a sharp triangle; instead, it is handled as "incoherent elastic" scattering in order to take full advantage of the analytic properties of $\delta(\beta)$.

g. Incoherent Elastic Scattering. In hydrogenous solids, there is an elastic (no energy loss) component of scattering arising from the "zero-phonon" term. In ENDF terminology, this is called the "incoherent elastic" term. The corresponding differential scattering cross section is

$$\sigma(E,\mu) = \frac{\sigma_b}{2}\,\exp(-2WE(1-\mu))\,, \qquad (371)$$

and the integrated cross section is

$$\sigma(E) = \frac{\sigma_b}{2}\left\{\frac{1-\exp(-4WE)}{2WE}\right\}. \quad (372)$$

In these equations, the Debye-Waller coefficient is given by

$$W = \frac{\lambda}{AkT}, \quad (373)$$

where λ is computed from the input frequency spectrum as shown by Eq.(347) and modified by the presence of discrete oscillators (if any) as shown above. LEAPR writes the bound scattering cross section σ_b and the Debye-Waller coefficient W as a function of temperature into a section of the ENDF-6 output with MF=7 and MT=2.

h. Coherent Elastic Scattering. In solids consisting of coherent scatterers—for example, graphite—the zero-phonon term leads to interference scattering from the various planes of atoms of the crystals making up the solid. Once again, there is no energy loss, and the ENDF term for the process is "coherent elastic scattering". The differential scattering cross section is given by

$$\sigma_{\rm coh}(E,\mu) = \frac{\sigma_c}{E}\sum_{E_i<E} f_i\,\exp(-4WE_i\delta(\mu-\mu_i)), \quad (374)$$

where

$$\mu_i = 1 - E_i/E, \quad (375)$$

and the integrated cross section is given by

$$\sigma_{\rm coh} = \frac{\sigma_c}{E}\sum_{E_i<E} f_i\,\exp(-4WE_i). \quad (376)$$

In these equations, σ_c is the effective bound coherent scattering cross section for the material, W is the effective Debye-Waller coefficient, E_i are the so-called "Bragg Edges", and the f_i are related to the crystallographic structure factors.

It can be seen from Eq.(376) that the coherent elastic cross section is zero below the first Bragg Edge, E_1 (typically about 2 to 5 meV). It then jumps sharply to a value determined by f_1 and the Debye-Waller term. At higher energies, the cross section drops off as $1/E$ until E=E_2. It then takes another jump, and resumes its $1/E$ drop off. The sizes of the steps in the cross section gradually get smaller, and at high energies there is nothing left but an asymptotic $1/E$ decrease (typically above 1 to 2 eV). LEAPR stores the quantity $E\sigma_{\rm coh}(E)$ as a function of energy and temperature in a section of the ENDF-6 output with MF=7 and MT=2. The cross section is easily recovered from this representation by dividing by E (this is done in the THERMR module of NJOY). The angular distribution of scattered neutrons can be calculated by extracting the f_i from the steps in $\sigma_{\rm coh}(E)$ file by subtraction. This process is carried out automatically in the GROUPR module of NJOY.

The calculation of the E_i and f_i depends on a knowledge of the crystal structure of the scattering material. The methods used are borrowed from HEXSCAT [30]. In general, the energies of the Bragg edges are given by

$$E_i = \frac{\hbar^2\tau_i^2}{8m}, \quad (377)$$

where τ_i is the length of the vectors of one particular "shell" of the reciprocal lattice, and m is the neutron mass. The f_i factors for a material containing a single atomic species are given by

$$f_i = \frac{2\pi\hbar^2}{4mNV}\sum_{\tau_i}|F(\tau)|^2, \quad (378)$$

where the sum extends over all reciprocal lattice vectors of the given length, and the crystallographic structure factor is given by

$$|F(\tau)|^2 = \left|\sum_{j=1}^{N}\exp(i2\pi\phi_j)\right|^2, \quad (379)$$

where N is the number of atoms in the unit cell, $\phi_j = \vec{\tau}\cdot\vec{\rho}_j$ are the phases for the atoms, and the $\vec{\rho}_j$ are their positions. The situation is more complicated for materials containing different atomic species, such as beryllium oxide. In these cases,

$$\sigma_c\exp -2WE_if_i = \left|\sum_{j=1}^{N}\sqrt{\sigma_j}\,\exp -W_jE_i\exp(i2\pi\phi_j)\right|^2, \quad (380)$$

where the coherent cross section and Debye-Waller factor can be different for each site in the unit cell. The effective coherent cross section is clearly given by

$$\sigma_c = \sum_{j=1}^{N}\sigma_j. \quad (381)$$

Since LEAPR only works with one material at a time, it doesn't have access to different values of W_j for the atoms in the unit cell. Therefore, it assumes that either W_jE_i is small, or the W_j doesn't vary much from site to site. This allows it to calculate the f_i using

$$|F|^2 = \left|\sum_{j=1}^{N}\frac{\sqrt{\sigma_j}}{\sqrt{\sigma_c}}\,\exp(-2\pi\phi_j)\right|^2. \quad (382)$$

For hexagonal materials, the lattice is described by the two constants a and c. The reciprocal lattice vector lengths are given by

$$\left(\frac{\tau}{2\pi}\right) = \frac{4}{3a^2}\left(\ell_1^2+\ell_2^2+\ell_1\ell_2\right)+\frac{1}{c^2}\ell_3^2. \quad (383)$$

where ℓ_1, ℓ_2, and ℓ_3 run over all the positive and negative integers, including zero. The volume of the unit cell is

$$V = \sqrt{3}a_2c/2. \quad (384)$$

For graphite, there are four atoms in the unit cell at positions [68]

$$(0,0,0),\ (-\frac{1}{3},\frac{1}{3},0),\ (-\frac{2}{3},-\frac{1}{3},\frac{1}{2}),\ (-\frac{1}{3},\frac{1}{3},\frac{1}{2})\,.$$

These positions give the following phases:

$$\phi_1 = 0\,, \tag{385}$$
$$\phi_2 = (-\ell_1+\ell_2)/3\,, \tag{386}$$
$$\phi_3 = -(2/3)\ell_1-(1/3)\ell_2+(1/2)\ell_3\,, \tag{387}$$
$$\phi_4 = -(1/3)\ell_1+(1/3)\ell_2+(1/2)\ell_3\,. \tag{388}$$

The form factor for graphite becomes

$$|F|^2 = \begin{cases} 6+10\cos[2\pi(\ell_1-\ell_2)/3] & \ell_3 \text{ even} \\ 4\sin^2[\pi(\ell_1-\ell_2)/3] & \ell_3 \text{ odd} \end{cases} \tag{389}$$

For the hexagonal close packed structure (hcp), which includes beryllium, there are two atoms per unit cell at

$$(0,0,0),\ (\frac{1}{3},\frac{2}{3},\frac{1}{2})\,,$$

and the form factor for hcp lattices like beryllium becomes

$$|F|^2 = 2+2\cos\Big[2\pi(2\ell_1+4\ell_2+3\ell_3)/6\Big]\,. \tag{390}$$

The beryllium oxide lattice consists of two interpenetrating hcp lattices, one for the beryllium atoms, and one for the oxygen. There are four atoms per unit cell with positions

$$(0,0,0),\ (\frac{1}{3},\frac{2}{3},\frac{1}{2}),\ (0,0,u),\ (\frac{1}{3},\frac{2}{3},u+\frac{1}{2})\,,$$

Using the approximation that the Debye-Waller factor doesn't vary from position to position in the unit cell gives the following expression for the structure factor:

$$\begin{aligned} |F|^2 &= \Big(1+\cos[2\pi(2\ell_1+4\ell_2+3\ell_3)/6]\Big) \\ &\quad \Big(r_1^2+2r_1r_2\cos(3\pi\ell_3/4)\Big)\,, \end{aligned} \tag{391}$$

where r_1^2 and r_2^2 are the bound coherent cross sections for beryllium and oxygen, respectively, and the effective coherent cross section, σ_c, is to be taken as 1.

LEAPR can also handle FCC lattices (Al, Pb) and BCC lattices (Fe). More formulas for the structure factor can be added to the code when needed.

i. Liquid Hydrogen and Deuterium. Materials containing hydrogen (H) or deuterium (D) molecules violate the assumption that spins are distributed randomly that underlies the incoherent approximation used for Eq.(349), and an explicitly quantum-mechanical formula is required to take account of the correlations between the spins of two atoms in the same molecule. This problem was considered by Young and Koppel [69]. Changing to our notation, the formulas for the hydrogen molecule (neglecting vibrations) become

$$\begin{aligned} \mathcal{S}_{\rm para}(\alpha,\beta) &= \sum_{J=0,2,4,\ldots} P_J \\ &\times \frac{4\pi}{\sigma_b}\Big[A_{\rm para}\sum_{J'=0,2,4,\ldots} + B_{\rm para}\sum_{J'=1,3,5,\ldots}\Big] \\ &\times (2J'+1)\mathcal{S}_f(w\alpha,\beta+\beta_{JJ'}) \\ &\times \sum_{\ell=|J'-J|}^{J'+J} 4\,j_\ell^2(y)\,C^2(JJ'\ell;00)\,, \end{aligned} \tag{392}$$

and

$$\begin{aligned} \mathcal{S}_{\rm ortho}(\alpha,\beta) &= \sum_{J=1,3,5,\ldots} P_J \\ &\times \frac{4\pi}{\sigma_b}\Big[A_{\rm ortho}\sum_{J'=0,2,4,\ldots} + B_{\rm ortho}\sum_{J'=1,3,5,\ldots}\Big] \\ &\times (2J'+1)\mathcal{S}_f(w\alpha,\beta+\beta_{JJ'}) \\ &\times \sum_{\ell=|J'-J|}^{J'+J} 4\,j_\ell^2(y)\,C^2(JJ'\ell;00)\,. \end{aligned} \tag{393}$$

The coefficients for the even and odd sums are given in the following table:

Type	A(even)	B(odd)
H para	a_c^2	a_i^2
H ortho	$a_c^2/3$	$a_c^2+2a_i^2/3$
D para	$3a_i^2/4$	$a_c^2+a_i^2/4$
D ortho	$a_c^2+5a_i^2/8$	$3a_i^2/8$

Here a_c and a_i are the coherent and incoherent scattering lengths (note that the characteristic bound cross section σ_b=$4\pi[a_c^2$+$a_i^2]$), P_J is the statistical weight factor, $\beta_{JJ'}$=$(E'_J-E_J)/kT$ is the energy transfer for a rotational transition, $j_\ell(x)$ is a spherical Bessel function of order ℓ, and $C(JJ'\ell;00)$ is a Clebsch-Gordan coefficient. The parameter y is given by $\kappa a/2$=$(a/2)\sqrt{4MkT\alpha/2}$, where a is the interatomic distance in the molecule. The translational weight w is 1/2 for H_2 and 1/4 for D_2. The sums over J' are treated as operators into order to keep the notation compact.

Young and Koppel assumed that the molecular translations were free, so the equations contain

$$\mathcal{S}_f(\alpha,-\beta) = \frac{1}{\sqrt{4\pi\alpha}}\,\exp\left[-\frac{(\alpha-\beta)^2}{4\alpha}\right]\,, \tag{394}$$

and

$$\mathcal{S}_f(\alpha,\beta) = \exp-\beta\mathcal{S}_f(\alpha,-\beta)\,, \tag{395}$$

the free-atom scattering function (with β positive). Note that α is multiplied by a translational weight of 0.5 or 0.25 when this equation is used in order to make the formula apply to a molecule with mass ratio 2 or 4, respectively.

These formulas as stated are appropriate for a gas of hydrogen or deuterium molecules. In a liquid, there are two additional effects to be considered: interference between the neutron waves scattered from different molecules, and the fact that the recoil of the hydrogen molecule is not really free. First, we will consider the latter effect. Experiments by Egelstaff, Haywood, and Webb at Harwell [70] and Schott at Karlsruhe [71] showed appreciable broadening of the quasi-elastic scattering peak for liquid hydrogen, and both groups ascribed this to diffusive effects. Later, Utsuro of Kyoto University constructed a simple analytic model [72] that included both diffusion and intermolecular vibrations and showed good agreement with experiment. More recently, Keinert and Sax of the University of Stuttgart proposed the model [73] that we follow here.

They suggested that the free translation term in the Young and Koppel formulas be replaced by the superposition of a solid-state like motion and a diffusive law. One can picture a hydrogen molecule bound in a cluster of about 20 molecules and undergoing vibrations similar to those of a hydrogen molecule in a solid. These clumps then diffuse through the liquid (hindered translations) according to the Egelstaff-Schofield effective width model discussed above.

As mentioned earlier, waves scattered from different molecules can also interfere. Inter-molecular coherence results when there is a correlation between the positions of nearby molecules. This kind of coherence is described by the "static structure factor" $S(\kappa)$. This quantity can be used in an approximation due to Vineyard as follows:

$$\frac{d^2\sigma}{d\Omega d\epsilon} = \frac{d^2\sigma_{\rm coh}}{d\Omega d\epsilon}\, S(\kappa) + \frac{d^2\sigma_{\rm incoh}}{d\Omega d\epsilon}\ . \qquad (396)$$

This is equivalent to using Eqs.(392) and (393) with a_c^2 replaced by $S(\kappa)a_c^2$ in the calculation of the coefficients A and B. The effects of this procedure will be shown below.

j. Mixed Moderators. In some cases, thermal evaluations give the scattering for a principal scatterer as bound in a moderator; for example, H in H_2O, or Zr in ZrH. The other atoms in the molecule are represented by an analytic law (free-gas O), or by another detailed scattering law (H in ZrH). In other cases, the scattering from the entire molecule is represented in one file. Examples from ENDF are BeO and methane. The molecular scattering is renormalized to be used with the principal scatterer (Be or H for these cases); the secondary scatterer is assumed to have zero scattering in the thermal range.

Taking BeO as an example, the thermal cross section can be represented as follows:

$$\sigma_{\rm BeO} = \frac{\sigma_{b,\rm Be}}{2kT}\sqrt{\frac{E'}{E}}\ \exp -\beta/2 \left\{ S_{\rm Be}(\alpha_{\rm Be},\beta) + \frac{\sigma_{b,\rm O}}{\sigma_{b,\rm Be}}\, S_{\rm O}(\frac{A_{\rm Be}}{A_{\rm O}}\alpha_{\rm Be},\beta)\right\}\ , \quad (397)$$

where $\alpha_{\rm Be}$ stands for α computed with the atomic mass ratio for Be, $A_{\rm Be}$. In practice, LEAPR first computes $S_{\rm Be}$ using the input α grid, and then it computes $S_{\rm O}$ on a new α grid obtained by transforming the input grid with the indicated mass ratio. The two parts can now be added up by weighting with the indicated ratio of the bound cross sections. The method for preparing a mixed $S(\alpha,\beta)$ for BeO is shown in detail below.

B. Running LEAPR

The formal input instructions for LEAPR can be found in the comment cards at the beginning of the source listing, in the NJOY manual [5], or on the NJOY web site [6]. In order to illustrate their use, we will analyze an example of running LEAPR for graphite. The basic physics for graphite is taken from the original GA evaluation [29]. A concise account appears in the new ENDF File 1 comment cards included in the input deck shown in Figs. 64 and 65.

The line numbers are for reference only and are not part of the input. The first card tells the code to write the $S(\alpha,\beta)$ output on unit 20. After the comment card, which is just for the user's convenience and does not go into the output file, line 4 gives the global options for the run. Here we see that 10 temperatures and a mid-size listing are desired. The default size for the phonon expansion of 100 is taken. The next card gives the ENDF MAT number to be used and a substitute for ZA. Here is a table of the current standard values from ENDF-102 [2]:

Compound	MAT	Compound	MAT
Water	1	Be in BeO	29
Para Hydrogen	2	Graphite	31
Ortho Hydrogen	3	Liquid Methane	33
H in ZrH	7	Solid Methane	34
Heavy Water	11	Polyethylene	37
Para Deuterium	12	Benzine	40
Ortho Deuterium	13	O in BeO	46
Be	26	Zr in ZrH	58
BeO	27	UO2	75
Be_2C	28	UC	76

We are currently using non-standard numbers for Al, Fe, U in UO2, and O in UO2. There are two additional defaulted parameters given on this input line. The parameter `isym` controls whether the output ENDF tape contains the symmetric $S(\alpha,\beta)$ or the asymmetric $\hat{S}(\alpha,\beta)$. The default is symmetric. The asymmetric option gives much better numerics, especially on short-word machines, but it is not sanctioned by the ENDF-6 format. Next, `ilog` controls whether the output file contains S or $\log_{10} S$. Giving the log of S is the ENDF-sanctioned way of handling very small numbers in File 7.

Line 6 gives information for the primary scatterer. In this case, there is only a primary; its ZA is 11.898 and

```
 1.  leapr
 2.  20
 3.  'graphite, endf model (extended) '/
 4.  10 1/
 5.  31 131./
 6.  11.898 4.7392 1 1/
 7.  0/
 8.  72 96 1/
 9.  .01008 .015 .0252 .033 .0504 .0756 .1008 .15
10.  2.52030e-1 .33 5.040600-1 7.560900-1 1.008120+0 1.260150+0 1.512180+0
11.  1.76421e+0 2.016240+0 2.273310+0 2.535520+0 2.802970+0 3.075770+0
12.  3.35401e+0 3.637900+0 3.927330+0 4.222710+0 4.523830+0 4.831110+0
13.  5.14443e+0 5.464110+0 5.790130+0 6.122610+0 6.461850+0 6.807830+0
14.  7.16077e+0 7.520670+0 7.887830+0 8.262340+0 8.644320+0 9.033960+0
15.  9.43136e+0 9.836730+0 1.025060+1 1.067190+1 1.110240+1 1.154090+1
16.  1.19886e+1 1.244520+1 1.291100+1 1.338580+1 14. 15. 16. 17. 18.
17.   19. 20. 22. 24. 26. 28. 30. 32.5 35. 37.5 40. 42.5 45. 47.5
18.   50. 52.5 55. 60. /
19.  0.000000+0 1.008120-1 2.016240-1 3.024360-1 4.032480-1 5.040600-1
20.  6.048720-1 7.056840-1 8.064960-1 9.073070-1 1.008120+0 1.108930+0
21.  1.209740+0 1.310550+0 1.411370+0 1.512180+0 1.612990+0 1.713800+0
22.  1.814610+0 1.915430+0 2.016240+0 2.117050+0 2.217860+0 2.318670+0
23.  2.419490+0 2.520300+0 2.621110+0 2.721920+0 2.822730+0 2.923540+0
24. 3.024360+0 3.125170+0 3.225980+0 3.326790+0 3.427600+0 3.528420+0
25.  3.629230+0 3.730040+0 3.830850+0 3.931670+0 4.032480+0 4.133290+0
26.  4.243780+0 4.364850+0 4.497620+0 4.643090+0 4.802480+0 4.977190+0
27.  5.168730+0 5.378620+0 5.608670+0 5.73473 5.860800+0 5.99896
28.  6.137130+0 6.28855 6.439970+0 6.60591
29.  6.771840+0 6.95376 7.135670+0 7.33502 7.534380+0 7.75289
30.  7.971400+0 8.21088 8.450360+0 8.975290+0
31.  9.550520+0 1.018100+1 1.087260+1 1.162970+1 1.245930+1 1.336970+1
32.  1.436670+1 1.545950+1 1.665710+1 1.796970+1 1.940930+1 2.098600+1
33.  2.271390+1 2.460820+1 2.668490+1 2.896020+1 3.145330+1 3.418730+1
34.  3.718250+1 4.046590+1 45. 50. 55. 60. 65. 70. 75. 80. /
35.  293.6/
36.  .005485 40/
37.  0. .346613 1.4135 3.03321 3.25901 3.38468 3.48269
38.  3.76397 4.05025 4.84696 7.35744 5.88224 4.63255
39.  4.48287 5.80642 4.63802 4.28503 3.92079 4.91352
40.  5.53836 7.51076 5.31651 5.40525 5.20376 5.3276
41.  7.17251 3.31813 4.50126 5.04663 4.2089 2.91985
42.  4.65109 13.1324 7.25016 6.5662 5.47181 5.06137
43.  5.19813 .457086 0./
44.  0. 0. 1. 0./
45.  0/
46. -400/
47. -500/
48. -600/
49. -700/
50. -800/
51. -1000/
52. -1200/
53. -1600/
54. -2000/
55. ' graphite  lanl       eval-may05 macfarlane '/
56. ' ref. 4               dist- '/
57. ' ---- endf/b-6        material 31 '/
58. ' ----- thermal neutron scattering data '/
59. ' ------ endf-6 '/
60. '  '/
61. '  temperatures = 293.6, 400, 500, 600, 700, 800, 1000,  '/
62. '                     1200, 1600, 2000 deg k. '/
63. '  '/
```

FIG. 64: Sample LEAPR input for graphite, part 1.

```
 64. ' history '/
 65. ' ------- '/
 66. ' '/
 67. ' changed temperatures may05'/
 68. ' '/
 69. ' this evaluation was generated at the los alamos national '/
 70. ' laboratory (apr 1993) using the leapr code.  the physical '/
 71. ' model is very similar to the one used at general atomic '/
 72. ' in 1969 to produce the original endf/b-iii evaluations '/
 73. ' (see ref. 1).  tighter grids and extended ranges for alpha '/
 74. ' and beta were used.  a slightly more detailed calculation '/
 75. ' of the coherent inelastic scattering was generated.  of '/
 76. ' course, the various constants were updated to agree with '/
 77. ' the endf/b-vi evaluation of natural carbon. '/
 78. '  '/
 79. ' theory '/
 80. ' ------ '/
 81. ' graphite has an hexagonal close-packed crystal structure.  the '/
 82. ' lattice dynamics is represented using a model with four force '/
 83. ' constants (refs.2,3).  one force constant is used to describe a '/
 84. ' nearest-neighbor central force that binds two hexagonal planes '/
 85. ' together, another describes a bond-bending force in an hexagonal '/
 86. ' plane, the third is for bond-stretching between nearest neighbors '/
 87. ' in a plane, and the fourth corresponds to a restoring force '/
 88. ' against bending of the hexagonal plane.  the force constants '/
 89. ' were evaluated numerically using a very precise fit to the '/
 90. ' high and low temperature specific heat and compressibility '/
 91. ' of reactor grade graphite.  the phonon spectrum was computed '/
 92. ' from this model using the root sampling method, and then used '/
 93. ' to compute s(alpha,beta). the coherent elastic scattering '/
 94. ' cross section was computed using the known lattice '/
 95. ' structure and the debye-waller integrals from the lattice '/
 96. ' dynamics model. '/
 97. '  '/
 98. ' references '/
 99. ' ---------- '/
100. ' 1. j.u.koppel and d.h.houston, reference manual for endf thermal '/
101. '    neutron scattering data, general atomic report ga-8774 '/
102. '    revised and reissued as endf-269 by the national nuclear '/
103. '    data center, july 1978. '/
104. ' 2. j.a.young, n.f.wilkner, and d.e.parks, nukleonik '/
105. '    band 1, 295(1965). '/
106. ' 3. j.a.young and j.u.koppel, j.chem.phys. 42, 357(1965). '/
107. ' 4. r.e.macfarlane, new thermal neutron scattering files for '/
108. '    endf/b-vi release 2, los alamos national laboratory report '/
109. '    la-12639-ms (march 1994). '/
110. '  '/
111. /
stop
```

FIG. 65: Graphite input, continued.

its free scattering cross section is 4.7392 barns. The next field indicates that there is only one of the primary scatterers in the compound. The coherent elastic option is set to 1, which means that the code obtains its information from the ENDF file. For ENDF/B-V evaluations, the user can select from several internal options for this. There are two additional defaulted parameters, the cold hydrogen option and the structure factor option. Both are defaulted to "none" for this case. The "Skold" structure factor option is used in ENDF/B-VII to provide inter-molecular correlation for D in D2O.

The "principal scatterer" may be hard to select for some compounds. For water, it is H. But for ZrH, it would be H for `mat=7` and Zr for `mat=58`. For mixed moderators like BeO, it is usually the light material. The value of `spr` should be chosen by looking at the low-energy limit for MF=3,MT=2 (elastic scattering) on the neutron file to be used with the new evaluation. The value for `npr` would be 2 for H_2O, or 1 for BeO.

The "secondary scatterer" card (line 7) is just "0/" for simple materials like graphite or beryllium. The behavior of LEAPR for molecular moderators is determined by the value of `B7`. The choice `B7=0.` is for mixed moderators like BeO and Benzine. In these cases, the entire $S(\alpha, \beta)$ for the molecule is given in MF=7,MT=4, and it is intended to be used with the neutron file for the pri-

mary scatterer. The secondary scatterer's cross section, atomic weight ratio, and effective temperature are only used for extending $S(\alpha,\beta)$ with the SCT approximation (see THERMR). When `B7>0.`, only the scattering law for the primary scatterer is given. The effects of the secondary scatterer are to be included later by using an analytic law. For example, in the water evaluation, the $S(\alpha,\beta)$ is for H in H_2O; the oxygen is included later by using a free-gas cross section using the given `sps` and `aws`.

The following lines are used to define the α and β grids for the LEAPR run. Each list is terminated by a "/0" character. The graphite case uses 72 α values and 96 β values. The next field gives the `LAT` parameter. In the ENDF thermal format, the values for $S(\alpha,\beta)$ for the higher temperatures are given on the same α and β grids as for the base temperature. But since α and β are inversely proportional to T, only the smaller α and β values would be seen at higher temperatures. The results for the higher values would normally be zero. This is a waste of space in the fields on the ENDF evaluation. Using `LAT=1` will spread the scattering law values for the higher temperatures out, thereby giving a more accurate representation.

The range of β helps determine the high-energy limit for the evaluation. For example, if T is 296K, a value of $\beta_{\max}$ of 160 will allow downscatter events of 4 eV to be represented without recourse to the SCT approximation. This implies that pretty good results would be obtained for incident neutrons with energies of 4 eV. If incident energies are limited to $\beta_{\max}kT$, the range of α values that can be obtained using Eq.(333) is limited to $\alpha_{\max} = 4\beta_{\max}/A$. The specific points in the α and β grids are hard to choose. Too many points makes the evaluation expensive to use; too few lead to inaccurate interpolated results. The low β grid should probably have about the same detail as the input $\rho(\beta)$ in order to reflect all the structure in the frequency distribution. Because of the smoothing effect of the convolutions, the grid can gradually get coarser as β increases. GA traditionally used a log spacing in this higher region. If translational modes are to be included, a finer β grid for small β may be required to get good results at small α. If discrete oscillators are included, additional β values might be needed near the values $\pm n\beta_i$ and their various sums and differences, especially for n=1. After running LEAPR, the user should examine the results *vs* β printed out on the listing and the results *vs* α printed out on the ENDF file to see whether the features of $S(\alpha,\beta)$ are being represented well enough. If the normalization and sum rule checks are not being satisfied well, this may be an indication that the grids are too coarse.

Line 35 starts the temperature loop with a first value of 296.3 K (0.0253 eV). At this point, we give the energy distribution $\rho(\epsilon)$. In this example, the energy grid spacing is .005485 eV, and there are 40 points given to describe the function. The list of probability values is terminated with a "/" character. See Fig. 66 for the graphite distribution. Line 44 following the frequency distribution says that there is no translational component, no diffusion component, and the weight for the continuous distribution is one. The input line after that is "0," meaning that there are no discrete oscillators for this material.

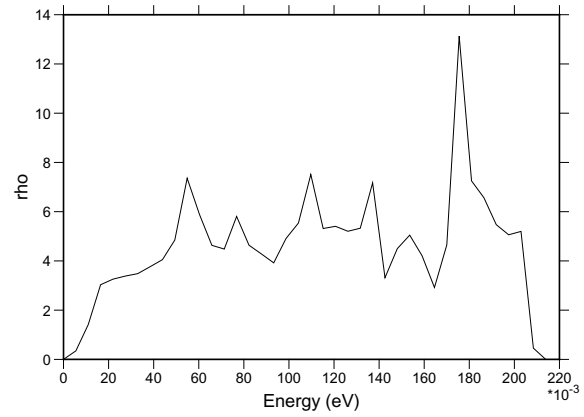

FIG. 66: The phonon frequency spectrum $\rho(\epsilon)$ used for graphite.

LEAPR allows you to change the input frequency distribution for every temperature, but this has not been done very often. The new ENDF/B-VII evaluation for D in D2O is an exception. In this case, the distribution is not temperature dependent, and the temperature values are given on lines 46 through 54. The minus sign in front of each subsequent temperature indicates that no additional parameters are to be read for that temperature.

The rest of the sample input for graphite gives comment cards to be entered into MF1/MT451 in the new evaluation. Fig. 67 and Fig. 68 show the results of the graphite calculation.

This graphite example doesn't demonstrate all the kinds of evaluations that can be produced using LEAPR. See reference [64] for additional examples.

XIV. PLOTR

The PLOTR module provides a general-purpose plotting capability for ENDF, PENDF, and GENDF files by generating files that VIEWR can use to generate high-quality Postscript plots. The following kinds of plots are produced:

- conventional 2-D plots (for example, cross section *vs* energy) of ENDF, PENDF, or GENDF data with the normal combinations of linear and log scales, automatic or user-specified ranges and labels, an optional alternate right-hand axis, and with one or two title lines;
- a set of experimental data by itself or superimposed on ENDF, PENDF, or GENDF curves;

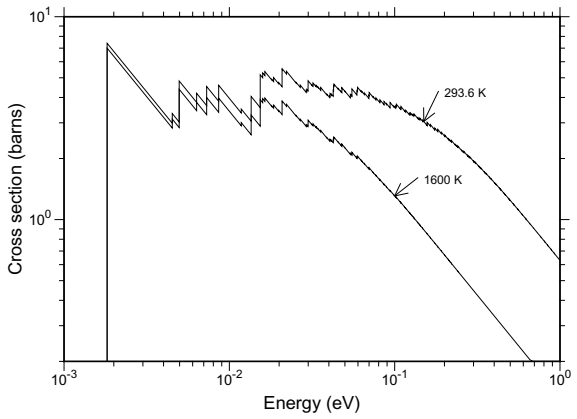

FIG. 67: The coherent elastic cross section for graphite at two temperatures showing the Bragg peaks.

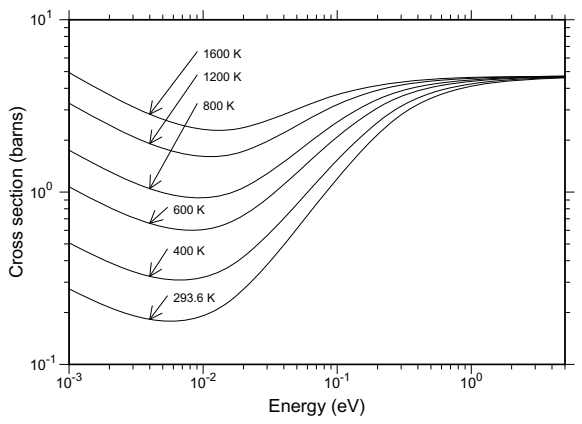

FIG. 68: The incoherent inelastic cross section for graphite at temperatures from 293.6 K to 1600 K.

- curves of various patterns, labeled with tags and arrows or described in a legend block;
- data points given with a variety of symbols with error bars (they can be identified in a legend block);
- detailed 3-D perspective plots of File 4 or File 6 angular distributions with a choice of a linear or a log axes for incident energy and a choice of energy range and viewpoint;
- selected 2-D plots of File 5 and File 15 emission spectra for specified incident energies, and selected 2-D emission spectra for given energies and particle types for File 6 data;
- detailed 3-D perspective plots of File 5, 6, or 15 energy distributions with a choice of log or linear axes and viewpoint (both $EE'\theta$ and $E\theta E'$ laws are supported);
- 3-D plots of GENDF data; and
- various 2-D plots for File 7 data, including both symmetric and asymmetric $S(\alpha, \beta)$ *vs* either α or β.

Methods for generating these types of plots will be given in the following subsections. An attempt has been made to keep the input as simple as possible by moving the less common options to the right-hand side of each input line so that they can be easily defaulted.

A. Simple 2-D Plots

The simplest kind of 2-D plot is for a single reaction from an ENDF, PENDF, or GENDF file using automatic scales and default labels. For example, to plot the total cross section of carbon from ENDF/B-VII, use the following input (don't type the line numbers; they are inserted here for reference):

```
 1.  plotr
 2.  31/
 3.  /
 4.  1/
 5.  '<endf/b-vii carbon'/
 6.  '<t>otal <c>ross <s>ection'/
 7.  4/
 8.  /
 9.  /
10.  /
11.  /
12.  6 20 600 3 1/ tape20 is carbon
13.  /
14.  99/
15.  stop
```

The data to be plotted are selected in line 12 using the normal MAT, MF, MT notation of ENDF. The "slash" at the end of the line hides several defaults, the first of which is the temperature, which defaults to 0 K. The "4" in line 7 selects log-log axes. The other choices for this parameter are given in the following fragment from the PLOTR input instructions:

```
type for primary axes
  1 = linear x - linear y
  2 = linear x - log y
  3 = log x - linear y
  4 = log x - log y
  set negative for 3d axes
  default=4
```

Lines 8 through 11 are blank, resulting in the choice of automatically defined ranges and default labels. Two title lines are given on lines 5 and 6. Note the use of special shift characters to change between lowercase and uppercase. The file on tape31 should be sent through VIEWR to generate the Postscript plot. The result is shown in Fig. 69.

In many cases, the default scales will give reasonable plots. However, in this case, the low-energy portion of the plot is approximately constant. It makes sense to change the lower limit of the x axis in order to expand

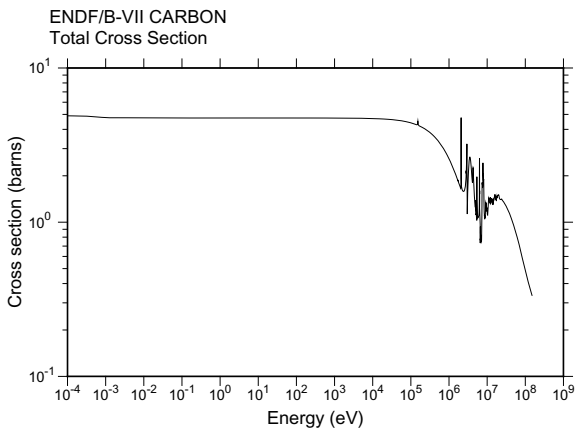

FIG. 69: Simple 2-D plot of the total cross section of ENDF/B-VII carbon using automatic log-log axes.

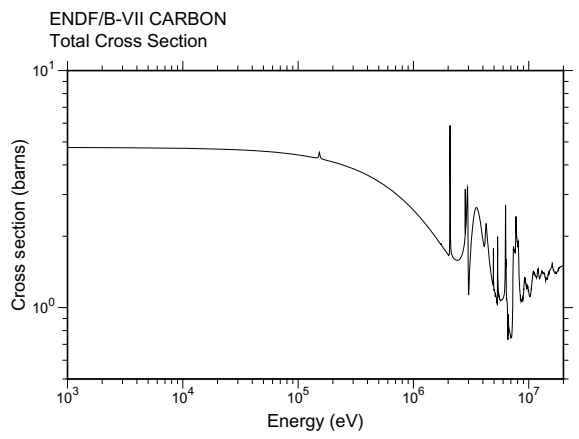

FIG. 70: Simple 2-D plot of the total cross section of ENDF/B-VII carbon using log-log axes with user-selected ranges.

the amount of detail shown at higher energies. A slight change in the lower limit of the y axis would also be beneficial. It is only necessary to change two lines as follows:

```
 8.  1e3 2e7/
10.  .5 10/
```

Note that the third parameter on these axes cards should always be defaulted for log scales. The results are shown in Fig. 70. This is a better balanced plot.

If the user needs to emphasize the high-energy region, linear scales are more appropriate. In addition, some people may prefer a different font. The following input gives the results shown in Fig. 71. Note how the new font is specified in line 3. The two font options currently available are a serif style "roman," implemented by the Postscript Times-Roman font, or a sans-serif style "swiss," implemented by the Postscript Helvetica font.

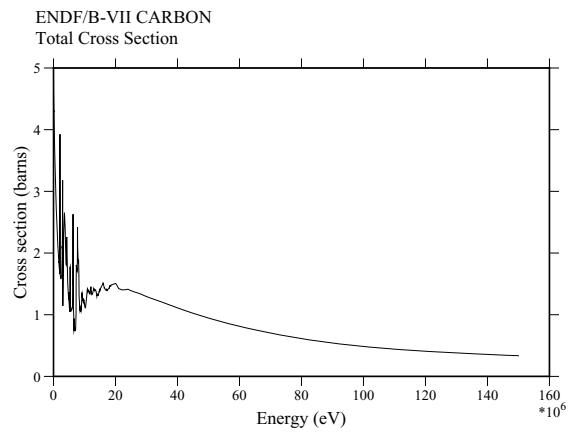

FIG. 71: Simple 2-D plot of the total cross section of ENDF/B-V carbon using linear axes to emphasize the high-energy region and a more elaborate font.

The default is the swiss style. The linear-linear axes option is selected in line 7.

```
 1.  plotr
 2.  31/
 3.  1 1/
 4.  1/
 5.  '<endf/b-vii carbon'/
 6.  '<t>otal <c>ross <s>ection'/
 7.  1/
 8.  /
 9.  /
10.  /
11.  /
12.  6 20 600 3 1/ tape20 is carbon
13.  /
14.  99/
15.  stop
```

The limits for the linear x axis in this example could have been specified by the user with a card of the form

```
 8.  0 1e7 2e6/  .
```

The general rule for linear axes is either give all three parameters explicitly, or default all three parameters. For log axes, either give the first two parameters and default the third, or default all three parameters.

Most ENDF or PENDF reactions will have many more energy points than can be shown on a graph like those in these figures. Therefore, PLOTR "thins" the grid down until there are fewer than `max` points on the plot (`max` is currently 10000). On the other hand, at some energies some ENDF reactions are described on fairly coarse energy grids using interpolation laws like "lin-lin" or "log-log". These representations will look as the evaluator intended if they are plotted using corresponding scales (for example, log-log interpolation on log-log scales, or log-lin interpolation on log-lin scales), but if they were to be plotted on a different set of axes, the cross sections between the grid points would be different from those intended. Therefore, PLOTR "thickens" the energy grid by

adding additional energy points between the grid points of the evaluation and computing the cross section at each of these points from the given interpolation law. The resulting curves will be faithful to the evaluation, but they may exhibit unphysical bumps and cusps in certain modes of presentation.

B. Multicurve and Multigroup Plots

Several curves can be drawn on each set of axes, and each curve can be taken from a different data source. The following input deck demonstrates how GENDF data can be compared with PENDF data by over plotting:

```
 1.  plotr
 2.  31/
 3.  /
 4.  1/
 5.  'ENDF/B-VII #EH.5>241#EXHX>Pu'/
 6.  /
 7.  4 0 2 1 5e3 2000/
 8.  .01 2e7/
 9.  /
10.  1 1e4/
11.  /
12.  6 23 9443 3 18 296.3/
13.  /
14.  'pointwise fission'/
15.  2/
16.  1 24 9443 3 18 296.3 1 1 1/
17.  0 0 1/
18.  'multigroup fission'/
19.  99/
20.  stop
```

The result is shown in Fig. 72. The PENDF data are requested on Card 12, and the GENDF data are requested on Card 16. Note the use of `ivers`=1 to denote GENDF data, and also note the settings for `nth`, `ntp`, and `nkh` necessary to select the P_0 infinitely dilute cross section for plotting. The GENDF-format data are automatically converted into histogram form for plotting. Here are the various options for `nth`, `ntp`, and `nkh` when requesting GENDF data:

```
special meanings for nth,ntp,nkh for gendf mf=3 data
     nth=0 for flux per unit lethargy
     nth=1 for cross section (default)
     ntp=1 for infinite dilution (default)
     ntp=2 for next lowest sigma-zero values, etc.
     nkh=1 for p0 weighting (default)
     nkh=2 for p1 weighting (total only)

special meaning for nth for gendf mf=6 data
     nth=1 plot 2-d spectrum for group 1
     nth=2 plot 2-d spectrum for group 2
       etc.
 no special flags are needed for mf=6 3d plots
```

A solid curve is requested for the PENDF data on line 13 (the default), and a dashed curve is selected for the GENDF data on line 17. Here are the available options for curve type `itype`:

```
type of line to plot
  0 = solid
  1 = dashed
  2 = chain dash
  3 = chain dot
  4 = dot
  default=0
```

This example also demonstrates using a "legend" block to identify the two curves. The position for the legend is given on Card 7. These values are normally determined by trial and error. The legend strings are give on lines 14 and 18. Note also the presence of a superscript in the title. The superscript depends on the VIEWR "instruction" mode, which is described in the VIEWR below.

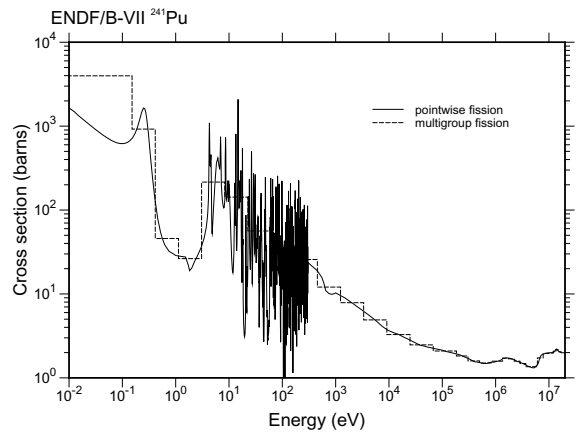

FIG. 72: Comparison of the multigroup fission cross section of ENDF/B-VII ^{241}Pu (dashed curve) with the corresponding pointwise cross section from the PENDF file (solid curve).

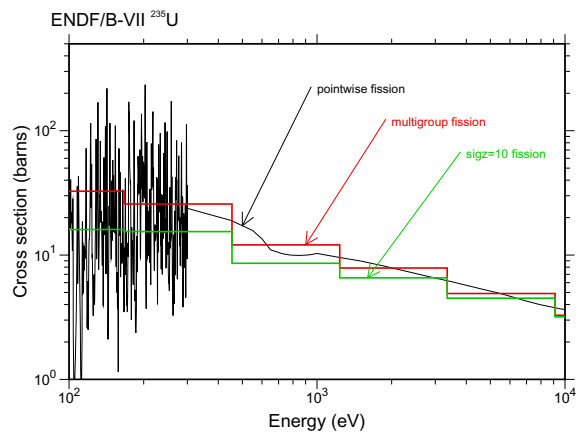

FIG. 73: Multigroup fission cross sections of ENDF/B-VII ^{238}U for infinite dilution and a 10-barn background compared with the corresponding pointwise cross section.

As another example of a plot of multigroup data, Fig. 73 shows both infinitely dilute and self-shielded cross sections, and the plot also compares them with the point-

wise cross section. In order to distinguish better between the different curves, color and double width are used. Here are the curve colors currently allowed:

```
curve color (def=black)
  0=black
  1=red
  2=green
  3=blue
  4=magenta
  5=cyan
  6=brown
  7=purple
  8=orange
```

In addition, this plot uses "tags" and arrows to identify the different curves. The position of the tags and the x location for the arrowhead usually must be determined by trial and error. See lines 15, 20 and 25, which give the x and y coordinates of the tag and the x coordinate where the arrow meets the curve. The input for this example follows:

```
 1.  plotr
 2.  31/
 3.  /
 4.  1/
 5.  'ENDF/B-VII #EH.5>235#EXHX>U'/
 6.  /
 7.  4 0 2 2/
 8.  100 10000/
 9.  /
10.  1 500/
11.  /
12.  6 23 9443 3 18 293,6/
13.  /
14.  'pointwise fission'/
15.  1000 200 500/
16.  2/
17.  1 24 9443 3 18 293,6 1 1 1/
18.  0 0 0 1 2/
19.  'multigroup fission'/
20.  2000 110 900/
21.  3/
22.  1 24 9443 3 18 300 1 3 1/
23.  0 0 0 2 2/
24.  ']S#LH.5>0#LXHX>=10 fission'/
25.  4000 60 1600/
26.  99/
27.  stop
```

There are some additional curve options, including closed curves filled with levels of grey, cross hatching, or colors. See `ishade` in the PLOTR or VIEWR input instructions. Color-filled curves can be seen in the examples in ERRORR and COVR sections. A grey region around a curve to show its confidence limits gives a nice touch.

These two examples require 293.6 K PENDF data for ^{241}Pu on the input file `tape23` and 293.6 K multigroup data for $\sigma_0 = \infty$, $\sigma_0 = 100$ barns, and $\sigma_0 = 10$ barns on `tape24`. These files can be generated with the following NJOY input deck:

```
 1.  reconr
 2.  20 21/ tape20 is Pu-241
 3.  /
 4.  9443/
 5.  .1/
 6.  0/
 7.  broadr
 8.  20 21 22
 9.  9443 1/
10.  .1/
11.  293.6
12.  0/
13.  purr
14.  20 22 23
15.  9443 1 3 20 8/
16.  293.6/
17.  1e10 100 10/
18.  0/
19.  groupr
20.  20 23 0 24
21.  9443 3 0 3 1 1 3 1
22.  /
23.  293.6
24.  1e10 100 10
25.  3 1/
26.  3 18/
27.  3 102/
28.  0/
29.  0/
30.  stop
```

C. Right-Hand Axes

PLOTR supports the capability to have different ordinate scales on the left and right sides of a graph. One place where this can be used is plotting a data curve and the ratio of another data curve to the first curve. An example of this is shown in Fig. 74. The input file used to make this plot is shown below. The RECONR module is run twice to make zero K PENDF files for plotting. Note the minus signs on lines 28 and 35—this selects the right-hand scale for those sections of the input. In line 36, the value of `nth` of 3 request the ratio calculation, and the additional input card in line 37 is read to access the second material for the ratio.

```
 1. reconr
 2. 20 22/ tape20 ENDF/B-V U-235
 3. /
 4. 1395/
 5. .02/
 6. 0/
 7. reconr
 8. 21 23/ tape20 ENDF/B-VII U-235
 9. /
10. 9228/
11. .02/
12. 0/
13. plotr
14. 31/
15. /
16. 1/
17. 'U-235 Capture Comparison'/
18. 'ENDF/B-VII to ENDF/B-V ratio'/
```

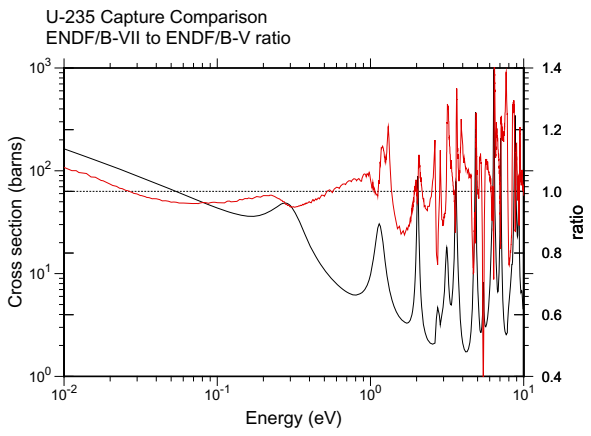

FIG. 74: Using a right-hand axis to plot the ratio of the ENDF/B-VII capture cross section for U-235 to the ENDF/B-V value. The black curve is the ENDF/B-V cross section, and the red curve is the ratio (right-hand axis).

```
19. 4 1/
20. 1e-2 10/
21. /
22. /
23. /
24. 0.4 1.4 .2/
25. 'ratio'/
26. 5 22 1395 3 102/ tape20 is carbon from V
27. /
28. -2/
29. 0/
20. 0 0 4/
31. 0/
32. 1e-2 1./
33. 10. 1./
34. /
35. -3/
36. 5 22 1395 3 102 0. 1 3/ tape20 is carbon from V
37. 6 23 9228 3 102 0./
38. 0 0 0 1/
39. 99/
40. stop
```

D. Plotting Input Data

PLOTR allows the user to insert data directly into the input deck. The main use of this is to superimpose experimental data over curves obtained from ENDF, PENDF, or GENDF tapes, but reading data directly from the input deck can also be used to add precalculated curves or eye guides to plots, or to add special features such as vertical lines to separate regions on plots. Experimental data points can be plotted with a variety of symbols, and x and/or y error bars can be included if desired. The error bars can be either symmetric or asymmetric. For experimental data, the curves and various sets of data points are normally identified using a legend block. The following input produces a typical example of this type of plot:

```
 1.  plotr
 2.  31/
 3.  /
 4.  1/
 5.  'ENDF/B-VII CARBON'/
 6.  '(n,]a>) with fake data'/
 7.  1 0 2 1 13e6 .25/
 8.  6e6 18e6/
 9.  /
10.  /
11.  /
12.  6 20 600 3 107/
13.  /
14.  'ENDF/B-VII MAT 600'/
15.  2/
16.  0/
17.  -1 0/
18.  '<s>mith & <s>mith 1914'/
19.  0/
20.  1.1e7 .08 .05 .05/
21.  1.2e7 .10 .05 .05/
22.  1.3e7 .09 .04 .04/
23.  1.4e7 .08 .03 .03/
24.  /
25.  3/
26.  0/
27.  -1 2/
28.  'Black & Blue 2008'/
29.  0/
30.  1.15e7 .07 .02 0. .2e6 0./
31.  1.25e7 .11 .02 0. .2e6 0./
32.  1.35e7 .08 .015 0. .2e6 0./
33.  1.45e7 .075 .01 0. .2e6 0./
34.  /
35.  99/
36.  stop
```

The results are shown in Fig. 75. The error bars for both of these simulated data sets are symmetric, as indicated explicitly in Cards 20 through 23, or by the zeroes in Cards 30 through 33. They can also be asymmetric if the lower and upper (or right and left) values are nonzero and different. Here is a list of the plotting symbols currently allowed by PLOTR and VIEWR drawn from the user input instructions in the source code:

```
no. of symbol to be used
  0 = square
  1 = octagon
  2 = triangle
  3 = cross
  4 = ex
  5 = diamond
  6 = inverted triangle
  7 = exed square
  8 = crossed ex
  9 = crossed diamond
 10 = crossed octagon
 11 = double triangle
 12 = crossed square
 13 = exed octagon
 14 = triangle and square
 15 = filled circle
 16 = open circle
 17 = open square
 18 = filled square
 default=0
```

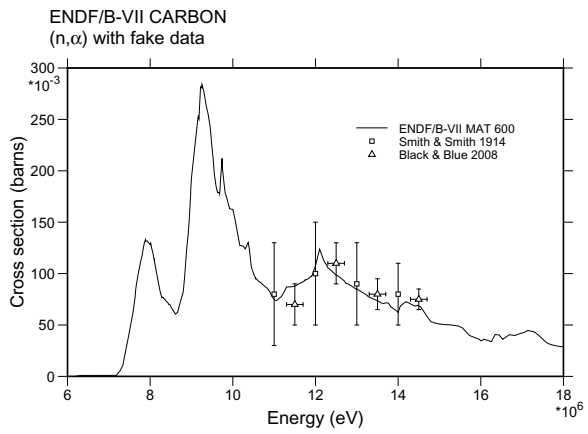

FIG. 75: Carbon (n,α) cross section compared with two sets of simulated experimental data represented with two types of error bars.

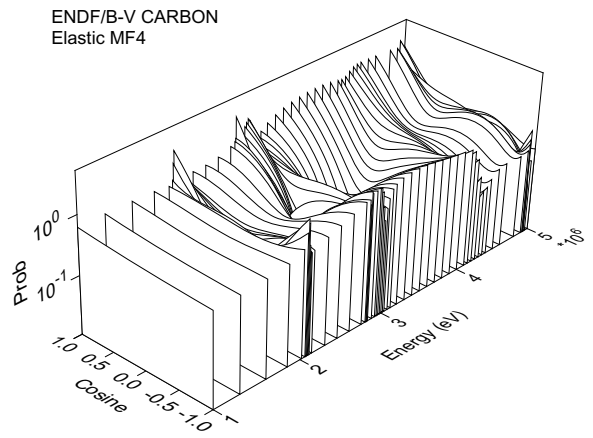

FIG. 76: Perspective view of carbon elastic scattering angular distribution.

E. Three-D Plots of Angular Distributions

ENDF angular distribution data, whether given in File 4 or File 6, can be very bulky. Therefore, it is useful to present them in the form of a perspective plot showing a family of angular distribution curves for each value of incident particle energy. An input deck to make such a perspective plot follows:

```
 1.  plotr
 2.  31/
 3.  /
 4.  1/
 5.  'ENDF/B-V CARBON'/
 6.  'Elastic MF4'/
 7.  -1 2/
 8.  /
 9.  /
10.  1e6 5e6 1e6/
11.  /
12.  /
13.  /
14.  5 20 1306 4 2/
15.  /
16.  99/
17.  stop
```

Fig. 76 shows the result using a linear axis for incident energy. This axis is the y axis, and its range has been limited to expand a particular part of the distribution. A linear scale emphasizes the high-energy region.

F. Three-D Plots of Energy Distributions

Three-D perspective plots are also useful for energy distributions. For the ENDF-5 and earlier formats, neutron secondary-energy distributions are given in File 5. The following input deck shows how to request a 3-D plot for File 5:

```
 1.  plotr
 2.  31/
 3.  /
 4.  1/
 5.  'ENDF/B-V Li-6'/
 6.  '(n,2n)]a neutron distribution'/
 7.  -1 2 1/
 8.  /
 9.  /
10.  4e6 20e6 2e6/
11.  /
12.  5 20 1303 5 24/
13.  /
14.  99/
15.  stop
```

Grids lines have been requested in Card 7. The result is shown in Fig. 77. Similar methods can be used to plot photon emission distributions from File 15.

The ENDF-6 format also provides for giving distributions for other emitted particles, such a protons, alphas, photons, and even recoil nuclei. This complicates the task of selecting which distribution is to be extracted from File 6. The user must specify the index for the particular outgoing particle to be considered (see `nkh`). Line 12 might become, for example,

```
12.  6 20 2437 103 0. 0 0 1/  ,
```

which would request a plot of the proton distribution for the (n,p) reaction of ^{54}Cr from ENDF/B-VI. Future versions of PLOTR should allow the particle to be selected by giving a ZA identifier.

For some evaluations, File 6 uses Law 7, where the energy-angle distribution is represented as $E \rightarrow E'$ distributions for several emission cosines μ. In these cases, the `ntp` parameter may be used to select one of the emission angles, and the 3-D plot shows the distribution for that angle.

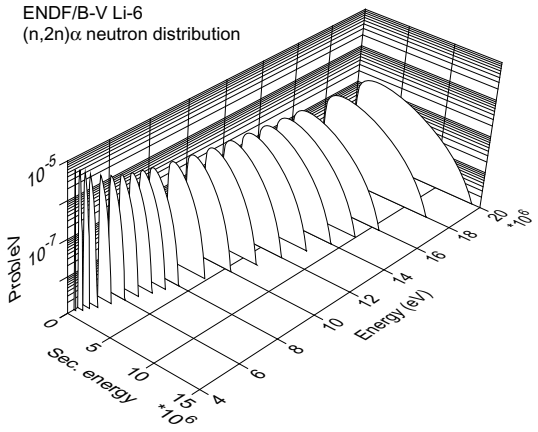

FIG. 77: Perspective plot of Li-6 (n,2n)α neutron secondary-energy distributions.

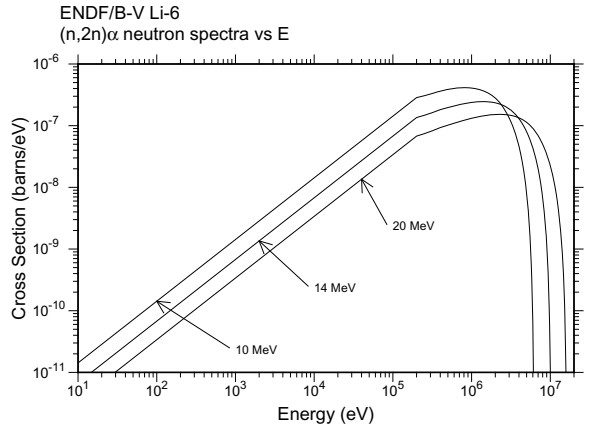

FIG. 78: Two-D plot of selected secondary neutron spectra for the Li-6 (n,2n)α reaction.

G. Two-D Spectra Plots from Files 5 and 6

It is difficult to see real detail on 3-D plots, and the emission spectra cannot be compared with measurements. Therefore, PLOTR has the capability to extract the spectrum for a particular particle and incident energy. The following input deck shows how several such spectra can be plotted on one graph.

```
 1.  plotr
 2.  31/
 3.  /
 4.  1/
 5.  'ENDF/B-V Li-6'/
 6.  '(n,2n)]a >neutron spectra vs <E>'/
 7.  4 0 2 2/
 8.  10. 2.e7/
 9.  /
10.  1e-11 1e-6/
11.  '<c>ross <s>ection (barns/e<V>)'/
12.  5 20 1303 5 24 0. 12/
13.  /
14.  '10 <m>e<v>'/
15.  1e3 2e-11 1e2/
16.  2/
17.  5 20 1303 5 24 0. 16/
18.  /
19.  '14 <m>e<v>'/
20.  1e4 2e-10 2e3/
21.  3/
22.  5 20 1303 5 24 0. 20/
23.  /
24.  '20 <m<e<v>'/
25.  1e5 2e-9 4e4/
26.  99/
27.  stop
```

The results are shown in Fig. 78. "Tags" are used to distinguish between the different incident energy values.

The method used for selecting which curve is to be plotted is awkward in the current version of PLOTR. The user must give the index number of the incident energy desired (like the value 12 in line 12 of this example).

When an ENDF-6 format File 6 is available, emission spectra will normally be given for several emitted particles, photons, and recoil nuclei. In addition, angular data may be given for various $E \rightarrow E'$ transfers using several different representations. This complicates the task of selecting which curve is to be extracted from File 6. For Law 1 data, the user must specify the particular outgoing particle to be considered (see `nkh`), the index for the incident energy desired (see `nth`), and the dependent variable to plot (see `ntp`). The result depends on the representation used. In every case, `ntp`=1 gives the cross section *vs* E', but for Legendre polynomials, `ntp`=2 gives the P_1 component *vs* E', and for Kalbach-Mann, `ntp`=2 gives the preequilibrium ratio *vs* E'. For Law 7, `ntp` specifies the emission angle. The graph will show a spectrum *vs* E' for the specified angle, specified incident energy E, and specified particle.

XV. VIEWR

The VIEWR module provides a general-purpose plotting capability for NJOY. It reads in user commands that define a variety of 2-D and 3-D graphs, and then it writes a Postscript file to display the graphs with high quality. Some of the capabilities of VIEWR include

- conventional 2-D plots (for example, cross section *vs* energy) with the normal combinations of linear and log scales, automatic or user-specified ranges and labels, an optional alternate right-hand axis, and with one or two title lines;
- curves of various line patterns, labeled with tags and arrows or described in a legend block;
- data points given with a variety of symbols with error bars (they can be identified in a legend block);
- the superposition of several plots on a given set of axes;

- detailed 3-D perspective plots for things like angular distributions or energy distributions, with a choice of linear or log axes, with a choice of working area and viewpoint, and with one or two title lines;
- color can be used to distinguish between different curves, and the graphics window area and page background can also have selected colors;
- closed regions can be displayed with various kinds of cross hatching for filled with selected colors; and
- multiple plots can be put on each page.

Many examples of graphs generated by VIEWR can be found elsewhere in this report.

A. Running VIEWR

The VIEWR module can be used as a general-purpose plotting engine by constructing input for it by hand. However, it is more commonly used to display data assembled by other NJOY modules. For example, PLOTR reads data in ENDF, PENDF, or GENDF formats and constructs an output file in VIEWR format. COVR constructs a fancy set of plots of cross-section covariance data using color contour maps and rotated subplots. HEATR can produce graphs showing the heating value, the photon energy production, and their associated kinematic limits. DTFR and ACER both are capable of generating VIEWR files that show the cross sections and distributions contained in the libraries that they produce. See the write ups for these various modules for examples of these kinds of plots.

There is actually only one line of input for the VIEWR module itself. It gives the input unit for plotting commands and the output Postscript unit. Here is an example. The lines on `tape40` can be prepared by hand if desired in order to use VIEWR as a manual plotting program. The Postscript results appear on `tape41`.

```
 1.  40 41
```

The following simple example illustrates plotting a $1/v$ curve between 1 meV and 10 eV. The line numbers are for reference only and are not part of the input.

```
 1.  /
 2.  1/
 3.  'Example of a 1/v cross section'/
 4.  /
 5.  4/
 6.  1e-3 10/
 7.  'Energy (eV)'/
 8.  1 100/
 9.  'Cross Section (barns)'/
10.  0/
11.  /
12.  0/
13.  1e-3 100/
14.  10  1/
15.  /
16.  99/
```

The input instructions for VIEWR and PLOTR are almost identical, except for the PLOTR commands to access ENDF, PENDF, and GENDF data. The first input card looks like this:

```
card 1
   lori          page orientation (def=1)
                  0  portrait (7.5x10in)
                  1  landscape (10x7.5in)
   istyle        character style (def=2)
                   1  roman
                   2  swiss
   size          character size option
                   pos = height in page units
                   neg = height as fraction of
                     subplot size (default=.30)
   ipcol         page color (def=white)
                  0=white
                  1=navajo white
                  2=blanched almond
                  3=antique white
                  4=very pale yellow
                  5=very pale rose
                  6=very pale green
                  7=very pale blue
```

It sets up the specifications for the graphics page. The orientation can be either portrait or landscape. Characters can use a serif style "roman," implemented by the Postscript Times-Roman font, or sans-serif "swiss," implemented by the Postscript Helvetica font. The character size can be specified in inches or as a fraction of the subplot size. The latter is handy when subplots are shrunk down to just a fraction of the page. The default character size is 0.30 inches. The page color set here is the color of the overall page. See below for the color inside the frame of a subplot. We have provided a set of pale colors here for these backgrounds. In line 1 or the example above, everything is defaulted.

The second input card on the VIEWR input file starts a new plot, subplot, or curve on an existing plot. It has the following format:

```
iplot      plot index
             99 = terminate plotting job
              1 = new axes, new page
             -1 = new axes, existing page
              n = nth additional plot on existing axes
             -n = start a new set of curves using
                  the alternate y axis
             default = 1
iwcol      window color (def=white)
            color list same as for ipcol above
factx      factor for energies (default=1.)
facty      factor for cross-sections (default=1.)
xll,yll    lower-left corner of plot area
ww,wh,wr   window width, height, and rotation angle
             (plot area defaults to one plot per page)
```

It also provides the flag indicating the the plotting job is finished; namely, `iplot=99/`. The value for `n` when its magnitude is greater than one is arbitrary, but we often use 2, 3, *etc.*, to indicate the second, third, *etc.*, curves on a set of axes. Users could just use the value `n=2` for every additional curve, and it would still work. The window

color `ipcol` refers to the color inside the axes frame for 2-D plots and to the color used for the various slices of the function for 3-D plots. The same set of light colors described for the page background is used here, and the pages look good if different colors are selected for the page color and the window color. The parameters `factx` and `facty` are handy for changing the x or y units for a plot or for scaling curves on a multi-curve graph to be closer to each other. The last few parameters on this card are only needed when several small subplots are put onto one page. They define the position, size, and rotation of the subplot. The covariance plots from COVR make use of this feature. The example just gives `iplot=1` in line 2 to start a default plot.

The next few lines of input are only given for new plots or subplots. There can be zero, one, or two lines of titles provided above the graph. The size of the plot readjusts to use the space not used by titles. The titles (and all the other labels in VIEWR) can use some special characters to trigger effects like subscript, superscript, symbol characters, size changes, or font changes. The rules for this are defined in the source listing as shown in Fig. 79.

Normal characters are just entered using mixed upper and lower case as desired. If desired, upper case can be forced using the trigger character. For example,

```
<p>u-239 from <endf/b-vi>
```

will have the indicated characters displayed in upper case. Similarly, Greek letters can be entered using mixed upper and lower case, or they can be given in all lower case and then forced to upper case with the trigger character:

```
Display ]a> and ]b>, then ]G> and ]D>.
Display ]a< and ]b>, then [g> and [d>.
```

Subscripts and superscripts use the "instruction" trigger character. For example,

```
ENDF/B-VI #EH<3#HXEX<h>e
```

displays the isotope name for ^{3}He correctly with a subscript one-half the size of the main characters. Subscripts lowered by one-quarter of the character size and with height equal to three-quarters of the character size could be obtained using

```
X#L.25H.75<min#HXLX<
```

For the mapping of the symbols and the other special characters available by shifting to the upper 128 characters in each font, check the Postscript Red Book [74]. The example just has one simple title line.

Returning to the input data, Card 4 (line 5 of the example) sets up the axes for the plot using `itype` and `jtype`. Log-log axes are chosen in this case. Several choices are allowed for adding grid lines to the axes. There are two ways available to label the curves in multi-curve graphs, namely, legend blocks, and curve tags. The default location for the legend block is the upper left corner of the graph area, but it can be positioned wherever desired using `xtag` and `ytag`. These values use coordinate units. The legend block contains a line for each curve giving the plotting symbol, curve style or color, and an identifying string entered below (see `aleg`). Curve tags are defined below.

Lines 6 through 9 correspond to Cards 5, 5a, 6, 6a, 7, and 7a in the input instructions. They define the axis limits, step sizes, and labels. The limits and step sizes are defined automatically based on the data, but they can also be defined manually be better control the display. This is often necessary for multi-curve plots, because the axes are chosen based on the extent of the first curve, which may not be suitable for all the curves in the graph. The graph scales are sometimes scaled with a power of ten, and these powers are always chosen to be divisible by three. Thus, you might have a scale labeled barns with a factor of 10^{-3} indicated; the numbers on the scale would then by milli-barns. Note that when no label is given, the numbering for the axis is also omitted.

Card 8 (line 10) is a dummy card that is only used to make the VIEWR input more compatible with the PLOTR input. As indicated, it should always contain `0/`. This is the only thing in VIEWR input that is different from PLOTR input.

Cards 9, 10, and 10a are used for curves on 2-D plots. These are represented by line 11 in the example above. You an use the `icon` connection parameter to choose simple curves, sets of symbols, or symbols connected by curves. A variety of symbols and curve types are provided, and curves can be rendered in color if desired. The colors provided are all darker than the light backgrounds provided. We do not allow for the light curves on dark backgrounds that some people like. Another way to distinguish between different curves is with the weight of the curve (see `ithick`). If the curve being plotted is closed, it can be filled with hatching patterns, shades of gray, or colors (shades of green, red, brown, or blue). The shades of gray could be used to show a variance band around a curve. The colors are used for the contour plots of covariances produced by COVR, and many other uses are possible. See Fig. 80 for examples of fills.

Card 10 gives `aleg`, the identifying string for the curve that is used if legend blocks or curve tags were requested. The location for the curve tag and the point where the arrow point touches the curve are defined on Card 10a. The arrow from the tag to the curve can be omitted, if desired.

Card 11 is only given for 3-D plots. It defines the imaginary work box that contains the perspective view of the 3-D function, and it defines the view point outside of the box where your eye goes. The perspective distortion will be greater for close viewpoints and less for distant ones. The 3-D coding is not very sophisticated, and it is best to keep the view point in the octant defined by the defaults x,y,z=(15,-15,15). That is, you are looking in along the x and z axes and the y axis is running away from you. You have the option of flipping the sense of

```
!
! Character specifications are similar to DISSPLA, except that
! the default case is lower instead of upper.  This allows
! mixed-case strings to be used in Postscript mode.  The
! following shift characters are used:
!     < = upper-case standard
!     > = lower-case or mixed-case standard
!     [ = upper-case greek
!     ] = lower-case or mixed-case greek
!     # = instructions
! Give one of the shift characters twice to get it instead of
! its action.  The following instructions are supported:
!     Ev = elevate by v as a fraction of the height
!          if v is missing or D is given, use .5
!     Lv = lower by v as a fraction of the height
!          if v is missing or D is given, use .5
!     Hv = change height by v as a fraction of the height
!          if v is missing or D is given, use .5
!     Fi = change to font number i
!     Mi = change mode number, where mode 0 is the lower 128
!          postscript characters and mode 1 is the upper 128
!     X  = reset E, L, or H.  Font and Mode must be
!          reset explicitly.
!     c is a real number, i is an integer.
!
```

FIG. 79: Rules for constructing labels with special characters.

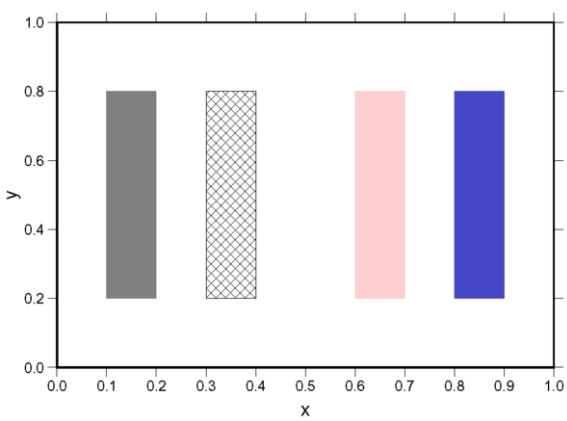

FIG. 80: Examples of a region filled with a mid grey, a region filled with cross hatching and outlined, a region filled with a light red, and a region filled with a darker blue.

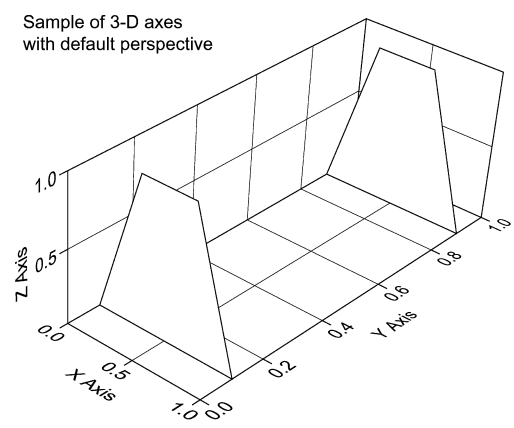

FIG. 81: The 3-D coordinate system using the default work box and view point.

the x axis, which is along the front of the work box as seen from the view point. This makes it possible to see features on either side of the slices of the function that are given for different discrete values of y. Figure 81 may help to clarify the coordinate system used.

At this point, all the specifications for the 2-D or 3-D graph have been entered, and it is only necessary to enter the actual values to be plotted. The format to be used is controlled by `nform`, which currently has only two options. Option `nform=0` is used for 2-D curves with optional error bars. In order to make the input simple for simple cases, the sequence of the possible x and y values is arranged to be defaulted at various points. For a simple curve, you can just give x and y, and then terminate the line with /. For the most common case of error bars (symmetric y errors), just give x, y, delta-y, and the terminating slash. The next step allows for asymmetric y error bars, then symmetric x error bars, and finally asymmetric x error bars. Terminate the list with an empty line (only /). See lines 11 through 15 in the example.

For 3-D views, the function is represented by giving a set of curves $z(x)$ for given values of y. The y values are given in increasing order, and each y value is followed by lines of x and z values terminated by /. The set of lines for each y is terminated by an empty card (only /). The y list is also terminated by an empty line, so there are two lines with only / given at the end of the `nform=1`

structure.

After the input data are given, the code loops back to search for another `iplot` value. If `iplot=99`, the job is finished.

XVI. MODER

The MODER module can be used to convert ENDF-format files between binary and text formats, and it can also be used to select materials from a multi-material ENDF "tape" or to merge several materials into a new multi-material tape. It works for ENDF, PENDF, GENDF, and ERRORR files. Running NJOY using files in binary mode can speed up jobs, but it doesn't have any effect on the accuracy of the results. One test on a typical PENDF job showed a 25% speedup.

A. Running MODER

The simplest MODER run converts an ENDF, PENDF, or GENDF file from text mode (ASCII) into the special blocked-binary mode used in NJOY. Positive unit numbers are used for text files and negative values are used for binary files. With the text-format file on unit 20, use the following input:

```
moder
20 -21
```

A run with reversed signs at the end of a job will convert the final result back into readable text format. Some examples of this use of MODER will be found in the various input examples in this report.

MODER can also be used to make a copy of an ENDF-format file. Just use two sequential MODER runs as follows:

```
moder
20 -21
moder
-21 -22
```

Now `tape20` is in text format and `tape21` and `itape22` are identical files in binary format.

A more advanced application of MODER is to combine several materials onto a single file. An example of this will be found in the ERRORR section where the ENDF/B-V tapes 515, 516, and 555 were mounted on units 20, 21, and 22 respectively. MODER was run with the input

```
1.  1 -23
2.  'ENDF/B-V NUBAR COVARIANCE MATERIALS'/
2.  20 1380
4.  20 1381
5.  21 1390
6.  22 1395
7.  22 1398
8.  20 1399
9.  0/
```

The line numbers are for reference and aren't part of the input. The result of this run is a binary `tape23` containing six materials extracted from the three different input files. The output on `tape23` in in binary mode. The value "1" on line 1 indicates that the input files are in ENDF format. Other values will select GENDF or ERRORR-output formats. The "0/" on line 9 terminates the list of input materials.

As usual, the full input instructions will be found in the code manuals, as comments in the source code, or on the NJOY web site.

XVII. DTFR

This module is used to prepare libraries for discrete-ordinates transport codes that accept the format designed for the S_N code DTF-IV [75]. Since this code predates many of the newer S_N and diffusion codes, the newer codes often allow DTF format as an input option. DTFR also contains a simple plotting capability for providing a quick look at its output. DTFR was the first output module written for the NJOY system, and it has been largely superseded by the MATXS/TRANSX system. But DTFR is still useful for some purposes because of its simplicity.

A. Transport Tables

Transport tables in DTF format are organized to mirror the structure of the data inside a discrete-ordinates transport code. These codes start with the highest energy group and work downward. (The conventional order for group indices is to increase as energy decreases.) Therefore, for each group g, the data required are the reaction cross sections for group g and the scattering cross sections to g from other groups g'. Each data element is said to occupy a "position" in the table for group g. The organization is shown in Table VIII.

The basic table consists of the three standard edits, namely, particle balance absorption (σ_a), fission neutron production cross section ($\bar{\nu}\sigma_f$), and total cross section (σ_t). These standard edits are followed by the group-to-group scattering cross sections. If desired, the standard edits can be preceded by `iptotl-3` special edits, which can be used in the transport code to calculate various responses of the system (for example, heating, activation, or gas production). If `iptotl` is the position of the total cross section, the positions of the three standard edits will be `iptotl-2`, `ipingp-1`, and `iptotl`, respectively. The positions of the special edits (if `iptotl>3`) will be 1, 2, ... `iptotl-3`.

Most transport tables describe only downscatter. In such cases, the position of the ingroup element is

TABLE VIII: Organization of data for one group in a transport table.

Position	Meaning
1	edit cross sections
···	(if any)
`iptotl-3`	last special edit
`iptotl-2`	particle-balance absorption
`iptotl-1`	fission neutron production
`iptotl`	total cross section
`iptotl+1`	upscatter $g \leftarrow g'$ ($g'>g$)
···	(if any)
`ipingp`	ingroup scattering $g \leftarrow g$
···	downscatter $g \leftarrow g'$ ($g' < g$)
`itabl`	end of table

`ipingp=iptotl+1`. Position `iptotl+1` would contain $\sigma_{g\leftarrow(g-1)}$, and so on. If thermal upscatter is present, the `nup` upscatter cross sections are between the total cross section element and the ingroup element. Therefore, position `iptotl-1` contains $\sigma_{g\leftarrow(g+1)}$, and so on. The position parameters must satisfy the following conditions: `iptotl` ≥ 3 , and `ipingp` = `iptotl` + `nup` + 1 . The number of positions for a group is called the table length. A full table will have the table length given by `itabl` = `iptotl-3` + 3 + `nup` + `ng` , where `ng` is the total number of groups in the set. Table lengths can be truncated in some cases. If this is done correctly, the important cross sections will be conserved, and valid results can still be obtained.

An example of a transport table is given in Fig. XVII A. This table was generated with DTFR for ENDF/B-VI ^{235}U (MAT=9228). It contains three special edits: the (n,2n) cross section, the fission cross section, and the radiative capture cross section. These are followed by the three standard edits, and then by the ingroup scattering for group 1. Since this is the highest energy group, there is no downscatter to this group from groups above, and the rest of the positions are filled with zeros. Group 2 starts on line 8 with the six edit cross sections. The seventh number is the ingroup cross section, and the eighth is the scattering from group 1 to group 2. Continuing to lines 14 and 15, there are now two downscatter elements: two to three and one to three. Note that there is an entire table for each Legendre order, and that each table has a header card that describes its contents. The ellipses were added to mark removed lines.

The last table in this example describes the photon production matrix. There are 30 neutron groups and 12 photon groups. The photon group index replaces the position index used in the neutron tables. Therefore, the first 12 numbers in this table correspond to photon production from neutron group 1. The photon groups are arranged in order of decreasing energy.

The header lines at the start of each table in this example give the Legendre order, number of groups, number of positions, MAT number, σ_0 number, and temperature.

DTFR will also produce a variant of the transport tables that was used for Los Alamos libraries in years past. These are sometimes called "CLAW Libraries" after the CLAW-III and CLAW-IV libraries available from the Radiation Safety Information Computational Center (RSICC) at the Oak Ridge National Laboratory. Although CLAW-IV uses a version of this format, it was actually generated using MATXS cross sections and the TRANSX code [9, 76]. They key feature of the CLAW tables is that the edits are removed from their normal position at the beginning of each group in the transport table and written out on separate lines. The format specified a particular list of reactions (see Table IX) that is defined using data statements in DTFR. In addition, thermal upscatter is not allowed in this format. An example of this style of transport table is given in Fig. XVII A. Note that eye-readable identifiers were added to the right-hand edge of each card by DTFR. The card labels contain the first two letters of the material name, the reaction name or the Legendre order, and a sequence number. The format of the header lines at the start of each table is different from the last example. The quantity in parentheses on the "EDIT XSEC" card is groups by number of edit reactions. For the neutron tables, it is table length by groups. For the gamma table, it is gamma groups by neutron groups.

B. Data Representations

The data stored into the transport tables are obtained from GROUPR. Since all the scattering matrix reactions are kept separate in the multigroup processing, it is necessary to add them up to compute the DTF scattering matrix. They are obtained from the sections with MF=6 on the input GENDF tape. Similarly, all the photon production matrices have to be added up to obtain the final DTF photon production table. These sections have MF=16.

The standard edit called particle-balance absorption is used in discrete-ordinates transport codes to calculate the balance table. The most fundamental definition for this quantity is

$$\sigma_{a,g} = \sigma_{t,g} - \sum_{g'} \sigma_{s,g'\leftarrow g} \ . \tag{398}$$

The total cross section is obtained from MF=3,MT=1 on the input GENDF tape. The scattering matrix is obtained by adding all the matrix reactions found in File 6 on the input tape. The absorption edit is often written in the form

$$\sigma_{a,g} = \sigma_{\gamma,g} + \sigma_{f,g} - \sigma_{n2n,g} \ , \tag{399}$$

```
IL= 1 TABLE 30 GP 36 POS, MAT= 9228 IZ= 1 TEMP= 3.00000E+02
 3.2597E-01  2.0823E+00  6.1629E-04  1.3965E+00  9.7755E+00  5.9989E+00
 3.1734E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 4.9950E-01  2.0532E+00  1.1756E-03  1.4681E+00  9.1321E+00  5.8659E+00
 3.0495E+00  1.1523E-01  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 6.8192E-01  1.8830E+00  1.5668E-03  1.1928E+00  8.0521E+00  5.7877E+00
 2.9838E+00  8.2484E-02  3.1214E-02  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
...
IL= 2 TABLE 30 GP 36 POS, MAT= 9228 IZ= 1 TEMP= 3.00000E+02
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 2.8003E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 2.5972E+00  3.8535E-02  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
 2.4730E+00  2.3851E-02  7.4855E-03  0.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
...
IP= 1 TABLE 30 GP 12 POS, MAT= 9228 IZ= 1 TEMP= 3.00000E+02
 0.0000E+00  0.0000E+00  1.1808E-02  2.0992E-02  2.4928E-02  5.7726E-02
 6.0219E-01  1.8328E+00  5.5365E+00  7.5176E+00  1.0018E+01  2.0404E+00
 0.0000E+00  0.0000E+00  1.0938E-02  1.9445E-02  2.4306E-02  5.8998E-02
 5.7158E-01  1.7441E+00  5.2857E+00  7.2063E+00  9.5514E+00  1.9537E+00
 0.0000E+00  0.0000E+00  8.6352E-03  1.5352E-02  2.6314E-02  7.8974E-02
 5.3190E-01  1.6485E+00  5.0937E+00  7.1114E+00  9.1307E+00  1.9158E+00
 0.0000E+00  0.0000E+00  5.0549E-03  8.9865E-03  2.9371E-02  1.0974E-01
 4.6950E-01  1.4973E+00  4.7871E+00  6.9512E+00  8.4624E+00  1.8534E+00
....
```

FIG. 82: Example of a transport table with internal edits.

which is good up to the threshold for other multiparticle reactions. Because of the presence of the (n,2n) term, the σ_a parameter is not equal to the real neutron absorption for high-energy groups. In fact, it is often negative.

The next standard edit is used to compute the fission neutron production rate when constructing a fission source. It is used together with a fission spectrum, which can be included in the specials edits (see below). The fission contributions from the GENDF tape are complicated. First, the prompt fission matrix may be given in MT=18 (total fission), or in the partial fission reactions MT=19, MT=20, MT=21, and MT=38, which stand for (n,f), (n,n′f), (n,2nf), and (n,3nf). Second, each of these matrices may take advantage of the fact that the shape of the fission spectrum is constant at low energies. Thus, fission can be represented using single fission χ_g^{LE} vector with this shape at low energies, with an accompanying fission neutron production cross section σ_{Pfg}^{LE} at low energies, and with a rectangular fission matrix $\sigma_{g,g\to g'}^{HE}$ at high energies. The third complication is the presence of delayed fission neutron emission. The delayed neutron yield $\bar{\nu}_g^D$ is retrieved from MF=3,MT=455, and the delayed neutron spectrum χ_g^D is obtained by adding up the spectra for the six time groups in MF=5,MT=455. The following equation shows how these separate terms are combined into the steady-state fission neutron production edit:

$$\bar{\nu}_g^{SS}\sigma_{fg} = \sum_{g'} \sigma_{fg\to g'}^{HE} + \sigma_{Pfg}^{LE} + \bar{\nu}_g^D \sigma_{fg} \ . \tag{400}$$

The associated steady-state fission spectrum is given by

$$\chi_{g'}^{SS} = \frac{\sum_g \sigma_{fg\to g'}^{HE}\phi_g + \chi_{g'}^{LE}\sum_g \sigma_{Pfg}^{LE}\phi_g + \chi_{g'}^{D}\sum_g \bar{\nu}_g^D \sigma_{fg}\phi_g}{\text{NORM}} \ , \tag{401}$$

```
U235       EDIT XSEC ( 30X 48) PROC BY NJOY1  ON 05/03/90
 3.03110E+00 2.85141E+00 2.75464E+00 2.66464E+00 2.87622E+00 3.56158E+00U2  ELS1
 4.43195E+00 4.38571E+00 3.94779E+00 3.53864E+00 3.34261E+00 3.61868E+00U2  ELS2
 4.74566E+00 6.13164E+00 7.65700E+00 9.34710E+00 1.08466E+01 1.16438E+01U2  ELS3
 1.19400E+01 1.28024E+01 1.32190E+01 1.31520E+01 1.29202E+01 1.30545E+01U2  ELS4
 1.27990E+01 1.19229E+01 1.32489E+01 1.41192E+01 1.47783E+01 1.54225E+01U2  ELS5
 3.78514E-01 4.17166E-01 4.61638E-01 5.44816E-01 8.11127E-01 1.37822E+00U2  INS1
 2.20958E+00 2.30405E+00 2.34199E+00 2.28404E+00 2.14540E+00 1.90879E+00U2  INS2
 1.60448E+00 1.29695E+00 9.63151E-01 4.00064E-01 4.54497E-02 3.03043E-03U2  INS3
 1.87467E-06 7.26072E-08 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00U2  INS4
 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00U2  INS5
 3.25966E-01 4.99502E-01 6.81925E-01 8.22064E-01 6.05949E-01 3.47278E-01U2  N2N1
 8.86188E-03 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00U2  N2N2
 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00U2  N2N3
 ...
U235   L=0 N-N TABLE ( 33X 30)
 1.39646E+00 9.77549E+00 5.99892E+00 3.17345E+00 0.00000E+00 0.00000E+00U2 0   1
 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00U2 0   2
 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00U2 0   3
 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00U2 0   4
 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00U2 0   5
 0.00000E+00 0.00000E+00 0.00000E+00 1.46808E+00 9.13206E+00 5.86594E+00U2 0   6
 3.04950E+00 1.15234E-01 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00U2 0   7
 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00U2 0   8
 ...
U235   L=1 N-N TABLE ( 33X 30)
 0.00000E+00 0.00000E+00 0.00000E+00 2.80033E+00 0.00000E+00 0.00000E+00U2 1   1
 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00U2 1   2
 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00U2 1   3
 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00U2 1   4
 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00U2 1   5
 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00U2 1   6
 2.59724E+00 3.85346E-02 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00U2 1   7
 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00 0.00000E+00U2 1   8
 ...
U235   L=0 N-P TABLE ( 12X 30)
 0.00000E+00 0.00000E+00 1.18077E-02 2.09916E-02 2.49277E-02 5.77264E-02U2 0   1
 6.02189E-01 1.83282E+00 5.53653E+00 7.51757E+00 1.00182E+01 2.04037E+00U2 0   2
 0.00000E+00 0.00000E+00 1.09379E-02 1.94453E-02 2.43062E-02 5.89981E-02U2 0   3
 5.71583E-01 1.74411E+00 5.28569E+00 7.20630E+00 9.55138E+00 1.95373E+00U2 0   4
 0.00000E+00 0.00000E+00 8.63521E-03 1.53515E-02 2.63136E-02 7.89737E-02U2 0   5
 5.31904E-01 1.64847E+00 5.09371E+00 7.11139E+00 9.13067E+00 1.91577E+00U2 0   6
 0.00000E+00 0.00000E+00 5.05490E-03 8.98652E-03 2.93715E-02 1.09745E-01U2 0   7
 ....
```

FIG. 83: Example of DTFR transport tables using separate edits.

where NORM is the quantity that will normalize the spectrum, namely, the sum of the numerator over all g'.

The total cross section is read directly from MF=3,MT=1. No transport corrections are made. The normal convention for libraries made with DTFR in the past was to supply P_4 tables and let the application code construct transport-corrected P_3 tables if it wanted them.

A flexible scheme is provided for constructing special edit cross sections. Each position can contain any of the cross sections available in File 3 of the GENDF tape, or a position can contain a combination of several cross sections weighted by multiplicities. For example, the ENDF/B-IV evaluation for ^{12}C contains the two reactions (n,α) (in MF=3,MT=107) and (n,n$'$)3α (in MF=3,MT=91). To obtain the helium production cross section, it is only necessary to request an edit made up as follows: $1 \times$ MT107 $+ 3 \times$ MT91 .

Most of the reactions for these special edits are requested by giving their ENDF MT numbers. The following special MT values are used to request some special quantities:

MT	Meaning
300	weighting flux (from MF=3,MT=1)
455	delayed neutron yield (MF=3,MT=455)
470	steady-state spectrum (see Eq. 401)
471	delayed neutron spectrum (MF=5,MT=455)

See the section on user input for more details.

DTFR can also construct transport tables suitable for calculations in the thermal range. These tables allow for upscatter, and the user can specify that bound scattering cross sections be used in the thermal range instead of the normal static elastic scattering reaction. The NJOY thermal capabilities are described in more detail in the

TABLE IX: Predefined edits for DTFR.

Reaction Name	Position	Reaction	Multiplicity
`els`	1	2	1
`ins`	2	4	1
`n2n`	3	16	1
	3	6	1
	3	7	1
	3	8	1
	3	9	1
`n3n`	4	17	1
`ngm`	5	102	1
`nal`	6	107	1
`np`	7	103	1
`fdir`	8	19	1
`nnf`	9	20	1
`n2nf`	10	21	1
`ftot`	11	18	1
`nd`	12	104	1
`nt`	13	105	1
`nhe3`	14	106	1
`n2p`	15	111	1
`npa`	16	112	1
`nt2a`	17	113	1
`nd2a`	18	114	1
`n2a`	19	108	1
`n3a`	20	109	1
`nnd`	21	32	1
`nnd2a`	22	35	1
`nnhe3`	23	34	1
`nna`	24	22	1
`n2na`	25	24	1
`nn3a`	26	23	1
`n3na`	27	25	1
`nnp`	28	28	1
`nn2a`	29	29	1
`n2n2a`	30	30	1
`nnt`	31	33	1
`nnt2a`	32	36	1
`n4n`	33	37	1
`n3nf`	34	38	1
`chi`	35	470	1
`chid`	36	471	1
`nud`	37	455	1
`phi`	38	300	1
`theat`	39	301	1
	39	443	0
`kerma`	40	443	1
`tdame`	41	444	1
`nusf`	42	1	1
`totl`	43	452	1

THERMR and GROUPR sections of this report. In brief, NJOY can provide thermal scattering cross sections and matrices for any material treated as a free gas in thermal equilibrium (a Maxwell-Boltzmann distribution), or it can provide data for a number of important moderator materials whose scattering laws $S(\alpha, \beta)$ are available in the ENDF libraries. These thermal "materials" look like different reactions for the dominant scattering isotope after being processed by GROUPR. Table X lists the different binding states available and the MT numbers used to request them. Materials like "H in H_2O" give the scattering for the principal scattering isotope only; the other isotope should be treated using free-gas scattering. There are two exceptions: benzene and BeO were evaluated as complete molecules, and the results were renormalized to be used with the cross section for the dominant isotope (H and Be, respectively). Therefore, these two materials should be used with the density corresponding to the dominant isotope, and no thermal contribution from the other isotope should be included at all (set both `mti` and `mtc` to zero).

TABLE X: Thermal reactions available to DTFR when using the normal ENDF/B thermal evaluations.

MTI	MTC	Thermal Binding Condition
221		free gas
222		H in H_2O
223	224	H in polyethylene (CH_2)
225	226	H in ZrH
227		Benzene
228		D in D_2O
229	230	C in graphite
231	232	Be metal
233	234	Be(BeO)
235	236	Zr in ZrH
237	238	O(BeO)
239	240	O(UO2)
241	242	U(UO2)
243	244	Al
245	246	Fe

As described in more detail in the subsection on user input, DTFR has an input parameter called `ntherm` that specifies the number of incident-energy groups to be treated using thermal cross sections; this parameter determines the breakpoint between the thermal treatment and the normal static treatment. DTFR subtracts the static elastic cross section (MT=2) from both the total and absorption edits in this group range, and it adds the `mti` and `mtc` cross sections to these positions. Similarly, it omits the contributions from the MT=2 matrix for these final-energy groups, and it adds in the contributions from `mti` and `mtc`. The user is free to use a number of upscatter positions less than `ntherm`. The code will truncate the table in a way that preserves the total thermal scattering cross section.

C. Plotting

Plots of the output data from a formatting program like DTFR are useful in two ways: first, they provide a nice summary of the library and help its users to understand the trends in the data easily, and second, they are helpful in quality control as an aid to finding errors in processing.

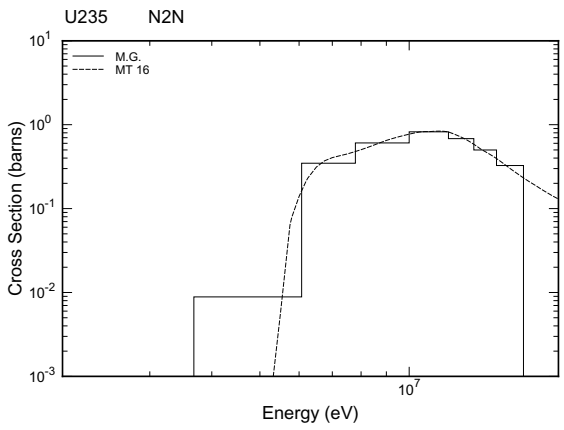

FIG. 84: An example of DTFR plotting for the (n,2n) reaction of ENDF/B-VI U-235. This plot was prepared using IFILM=1 and in-table edits. All the threshold reactions are shown using the same energy range to make comparisons easy.

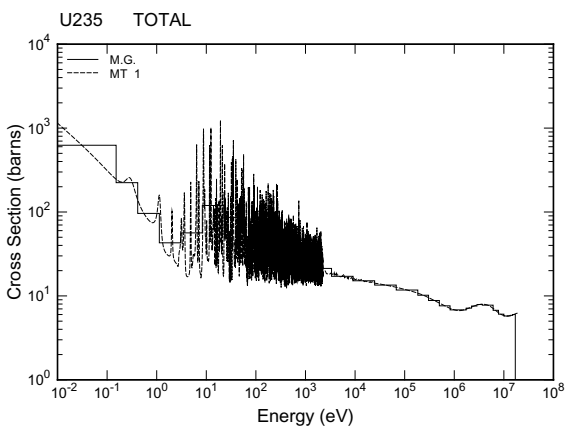

FIG. 85: This is the total cross section for ENDF/B-VI U-235 from the same run as the previous plot. The energy range from 1×10^{-5} eV to 1×10^{-2} eV was removed to expand the rest of the energy scale.

DTFR automatically makes two-dimensional log-log graphs for all the special edit positions and the three standard edit positions. If available (see `npend`), pointwise cross sections are plotted on the same frame as the group cross sections. If the multigroup cross section is a combination of several reactions, the pointwise cross sections for all of the components are plotted. An example of this will be found below. DTFR also prepares three-dimensional isometric projections of the P_0 scattering matrix and the P_0 photon-production table. The user can request that one reaction be plotted per page, or that four reactions be drawn on the same page. Examples of these plots are given in Fig. 84–88.

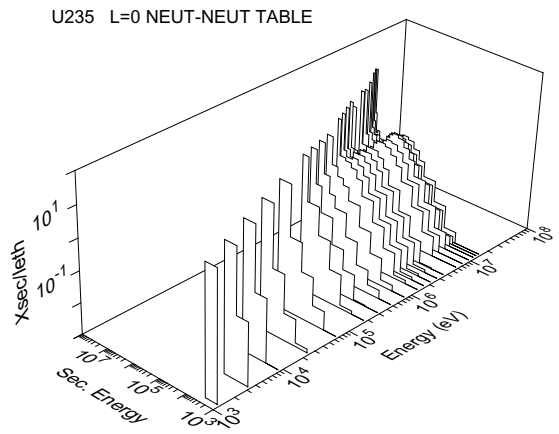

FIG. 86: This plot shows the P_0 scattering matrix for ENDF/B-VI U-235 by incident and secondary energy.

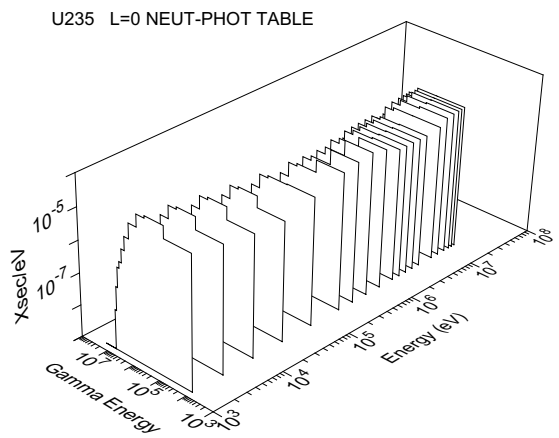

FIG. 87: The 30×12 P_0 photon production matrix for U-235 is plotted versus neutron energy and photon energy.

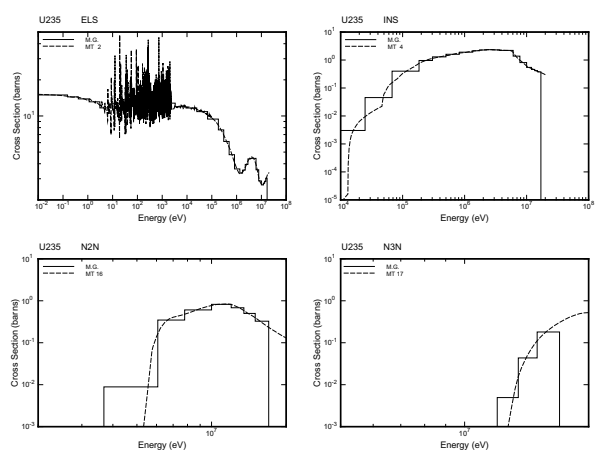

FIG. 88: An example of DTFR plotting using IFILM=2 for ENDF/B-VI U-235 for some of the standard edits with `iedit=1`.

D. Running DTFR

The user input specifications will be found in the comment cards at the beginning of the DTFR module source code, the NJOY manuals, or the NJOY web site. Some comments on the input commands are given here.

As usual, card 1 is used to assign the input and output units for the module. `nin` must be an output tape from GROUPR, and it can be in either binary or coded mode. The output file `nout` must be in coded mode. It will contain the DTF-format card images. File `npend` should be the same PENDF tape that was used in GROUPR when `nin` was made. It is only needed if plots are requested. The `nplot` file will contain the input lines that VIEWR will use to prepare the Postscript plots of the DTFR results. Card 2 starts out with the print flag `iprint`, which is usually set to 1. The parameter `ifilm` can be used to suppress plotting, or to request plots with either 1 or 4 graphs per frame. Examples of the DTFR plotting capability were given above. The value of `iprint` is used to control the output format. If it is equal to zero, a conventional DTF-type table is produced. If any edits were requested, they are given in the table using the first few positions in the table. The separate-edits option is used to produce cross sections in the "CLAW" format. In this format, the edits are extracted from the table and written out separately with identification information appended to the right side of each card image. The scattering tables have only the standard 3 edits, and they also have standard identification fields added to the right side of each card. Examples of both styles were given above.

Cards 3 through 5 are used for `iedit=0` only. The first parameter is `nlmax`. It is the number of Legendre tables desired; that is, it would be 4 for a P_3 set. The number of groups `ng` must agree with the number on the input tape from GROUPR. The value of `iptotl` is used to determine both the position of the total cross section in the table and the number of special edit positions at the front of the table (`iptotl-3`, which can be zero). Note that `ned` is not the number of edit positions; it is the number of edit specification triplets to be read. Therefore, `ned`$\geq$`iptotl-3`. Also `ipingp=iptotl+1` if there is no upscatter, and `ipingp=iptotl+nup+1` when the total number of upscatter groups is `nup`. (The parameter NUP is not actually given in the input; it is always equal to `ipingp-iptotl-1`.) The length of a full table is given by

```
  iptotl-3
+ 3
+ nup
+ ng
  itable .
```

However, smaller table lengths can be requested; truncation will be performed in a way that preserves the production cross sections. The parameter `ntherm` can be set to zero if no thermal upscatter cross sections are desired. If nonzero, it refers to the number of incident thermal groups, and it is used to define the breakpoint between the thermal and epithermal treatments. It is not necessarily equal to `nup`.

Card 3a is only given if `ntherm>0`. It gives the MT numbers for the thermal incoherent and coherent cross sections to be extracted from the GENDF tape. Examples might include 221 and zero for thermal scattering, or 229 and 230 for graphite. `NLC` can be used to truncate the anisotropy of the coherent term, if desired. For thermal cases, DTFR subtracts the static elastic scattering from both the total and absorption positions for the lowest `ntherm` groups, and then it adds in the cross sections corresponding to `mti` and `mtc` for these thermal groups. When reading through the matrices, it omits all the contributions from the static elastic matrix for the thermal groups, and adds in the `mti` and `mtc` contributions for these groups instead.

Card 4 gives Hollerith names for the edit cross sections. These `ned` names will appear on the output listing, but they are not passed on to the output file. Card 5 specifies the contents of each of the special edit fields. Note that each edit can be any linear combination of the cross sections on the input GENDF tape. This feature can be used to produce complex edits like gas production. An example follows:

```
1.   ...
2.   n.he4 kerma fiss
3.   1 107/
4.   1 91 3/
5.   2 301/
6.   3 444/
7.   ....
```

The line numbers are not part of the input. Line 1 represents all the input cards before card 4, and line 7 represents all the cards after card 5. This input is for ENDF/B-IV ^{12}C. Lines 3 and 4 construct a helium-production cross section as the sum of (n,α) and three times (n,n$'$)3α. Lines 5 and 6 assign two more edit positions for heating and damage, respectively. The MT numbers used for the `mt` field on card 5 are usually just the ENDF MT numbers for the reaction. However, special values are used to request the weighting flux, the steady-state and delayed components of the fission neutron spectrum, or the delayed fission neutron yield. Remember that the steady-state fission neutron production cross section will be found in position `iptotl-1` of the transport table.

When a multigroup edit is a combination of several cross sections as in this example, the plot of the edit includes the pointwise cross sections for all of the component reactions. Fig. 89 illustrates this for the ^{12}C helium-production reaction. Another useful trick is to build up a compound edit out of two reactions using a multiplicity of zero for the second reaction. This causes the second reaction to be plotted but not included in the edit cross section. This trick can be used to compare the energy-balance (MT=301) and kinematic (MT=443) versions of the KERMA factor. The appropriate input lines are

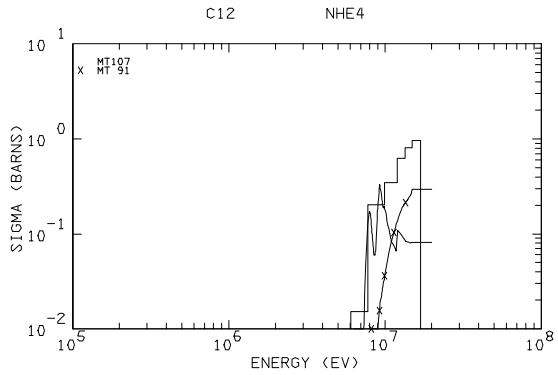

FIG. 89: An example of a DTFR edit plot showing multiple pointwise curves that are the components of a compound edit. This graph is for ENDF/B-IV C-12 helium production. The histogram gives the sum of the (n,α) cross section and three times the (n,n$'$)3α cross section.

```
...
heat/
1 301 1 1 443 0/
...
```

Note that this method is used in the predefined edits associated with `iedit=1` (see Table IX).

Card 6 is used instead if `iedit=1`. As described above, the list of special edit cross sections is fixed for the CLAW format. Therefore, it is only necessary to give `nlmax` and `ng`. The number of thermal groups is automatically set to zero.

The next card read for either choice of output format is card 6, which controls the generation of a photon production matrix. The number of photon production tables is usually zero (none), or one. Only a few materials in the ENDF/B libraries have anisotropic photon production data. Of course, `ngp` must agree with the photon group structure used in GROUPR.

The input deck ends with a material description card for each material to be processed. These materials must all be on the input GENDF tape. (Multi-MAT GENDF tapes can be prepared using MODER from single-material GROUPR output files, but since it is easy to combine materials at the DTF-format stage, DTFR is usually run for one MAT at a time.) The Hollerith material names given will appear in the comment cards before tables and in the special labels added to the right-hand edge of each card. The material numbers `mat` are the same ENDF MAT numbers used when preparing the multigroup cross sections. DTFR has only a limited capability of handling self-shielded cross sections. The user can specify that a given set of tables be prepared using one particular temperature `dtemp` and one particular background cross section, the `jsigz`th value in the set.

XVIII. CCCCR

This module is used to produce the standard CCCC interface files ISOTXS, BRKOXS, and DLAYXS from GROUPR output.

A. Introduction

The CCCC interface files (commonly pronounced "four cees") were developed by the Committee for Computer Code Coordination for the US Fast Breeder Reactor Program. When the members of this committee started work in 1970, they noted that because of the large variety of computers and computer systems, computer codes developed at one laboratory were often incompatible with computers at other laboratories. Major rewrites of codes or wasteful duplicate efforts were common. They hoped to create a system that would allow different laboratories to create codes that could be moved to other sites more easily. Moreover, they hoped that the codes developed at different laboratories could easily work together, thereby achieving larger and more capable calculational systems than any one laboratory could hope to develop by itself.

They approached this problem in two ways. First, they tried to establish general programming standards that would make computer codes more portable. And second, they tried to establish standard interface files for reactor physics codes that would make it easier for computer codes to communicate with each other. The results of this work appeared in fullest form as the CCCC-III and CCCC-IV standards [8, 77].

The MINX code [78], which was the predecessor of NJOY, was able to produce libraries that used the CCCC-III interface formats [79]. The LINX and BINX library management codes [80] and the CINX group collapse code [81] were also released during this period. Major codes that used data in CCCC-format included SPHINX [82] from Westinghouse, TDOWN [83] from General Electric, DIF3D [12] from Argonne National Laboratory, and ONEDANT [11] from Los Alamos. It was indeed found that codes could be moved more easily than before. Analysts could use ONEDANT and DIF3D on similar problems; they could even use one code to generate utility files (for example, mixture and geometry files) that would work with the other code! When NJOY was developed, it first produced version III formats and was later upgraded to the CCCC-IV standard.

With the demise of the breeder reactor program, development of the CCCC system stopped. However, many good programs are still available that make use of CCCC files and programming standards. The Los Alamos DANDE system [84] is an example of how the use of standard interface files can be used to couple several reactor physics programs together into an easy-to-use and powerful product. In areas where the CCCC standards were not very successful, such as gamma ray cross sections and cross sections for the fusion energy

range, the MATXS format is available as an alternative. This generalized material cross section library format uses CCCC-type techniques. The current advanced S_N transport code at Los Alamos, PARTISN [13] uses CCCC methods and interface files extensively. Thus, the CCCC spirit is not dead.

B. The Standard Interface Files

The CCCCR module produces data libraries that use three of the CCCC-IV standard interface files, namely:

ISOTXS for nuclide (isotope)-ordered multigroup neutron cross sections including cross section versus energy functions for the principal cross sections, group-to-group scattering matrices, and fission neutron production and spectra tables;

BRKOXS for Bondarenko-type self-shielding factors versus energy group, temperature, and background cross section for the reactions with major resonance contributions; and

DLAYXS for delayed-neutron precursor yields, emission spectra, and decay constants for the major fissionable isotopes.

The format of each of these files, the definition of the types of data included, and the uses and weaknesses of these three standard file formats are discussed in the following three sections.

As mentioned in the preceding section, the normal form of the CCCC files is binary and sequential. CCCCR writes its output in this binary mode. Of course, coded versions (BCD, ASCII, EBCDIC, etc.) are needed to move library files between different machines, and the formats used for the coded versions are given in the file descriptions below. A separate program, BINX [80], is used to convert back and forth between coded and binary modes. BINX can also be used to prepare an interpreted listing of a library. CCCCR can prepare an entire multimaterial library in one run if a multimaterial GENDF file is available. It can also be used to prepare an interface file containing only one material. These one-material files can be merged into multimaterial libraries using the LINX code [80].

C. ISOTXS

The format for the ISOTXS material (isotope)-ordered cross section file will be found in the CCCC reference, in the NJOY manuals, and on the NJOY web site.

Most of the variables in the "File Identification and File Control" record are taken from the user's input. Note that `MAXUP` is always set to zero. CCCCR does not process the NJOY thermal data at the present time. The `ICHIST` parameter will always be zero. CCCCR does not produce a file-wide fission spectrum or matrix. The old practice of using a single fission spectrum for all calculations is inaccurate and obsolete. Actually, the effective fission spectrum depends on the mixture of isotopes and the flux. Any file-wide spectrum would have to be at least problem dependent, and it should also be region dependent. The parameters `NSCMAX` and `NSBLOK` in the "File Control" record will be discussed in connection with the scattering matrix format.

In the "File Data" record, the Hollerith set identification and the isotope names are taken from the user's input. As mentioned above, the file-wide fission spectrum `CHI` never appears. The mean neutron velocities by group (`VEL`) are obtained from the inverse velocities computed by GROUPR:

$$\left\langle \frac{1}{v} \right\rangle_g = \frac{\int_g \frac{1}{v}\,\phi(E)\,dE}{\int_g \phi(E)\,dE} \,, \tag{402}$$

where g is the group index, $\phi(E)$ is the GROUPR weighting spectrum, and v is the neutron velocity, which is computed from the neutron mass and energy using $v=\sqrt{2E/m}$. The units of these quantities are s/m; they are converted to cm/s for ISOTXS by inverting and multiplying by 100. The group structure [see `EMAX(J)` and `EMIN`] is obtained directly from MF=1,MT=451 on the GENDF tape. Note that GROUPR energy groups are given in order of increasing energy and ISOTXS energy groups are given in order of decreasing energy. CCCCR handles the conversion.

The "File-Wide Chi Data" record never appears; see the discussion above for the reasons.

In the "Isotope Control and Group Independent Data" record, the first ten parameters are taken from the user's input. The gram atomic weight for the material (`AMASS`) can be computed from the ENDF AWR parameter available on the GENDF file using the gram atomic weight of the neutron as a multiplier. The energy-release parameters `EFISS` and `ECAPT` must also be computed by the user. The `ECAPT` values are normally based on the ENDF Q values given in File 3, but, in some cases, it is also necessary to add additional decay energy coming from short-lived activation steps. For example, the `ECAPT` value for ^{238}U should include the energy for the ^{239}U β-decay step, and perhaps even the energy from the ^{239}Np β decay. The values for `EFISS` should be based on the total non-neutrino energy release, which can be obtained from MF=3,MT=18 or MF=1,MT=458 on the ENDF tape. The `TEMP` parameter is normally set to 300 K. The value of `SIGPOT` can be computed from the scattering-radius parameter AP in File 2 of the ENDF tape using $\sigma_p=4\pi a^2$. The parameter `ADENS` is usually set to zero to imply infinite dilution. `KBR` can be chosen based on the normal use of the material by the community for which the library is being produced.

The `ICHI` parameter is related to the `ichix` parameter in the user's input. As discussed above, the op-

tion ICHI=0 is never used by CCCCR. Beyond that, the NJOY user has the option of producing a fission χ vector using the default GROUPR flux (which is available on the GENDF tape) or a user-supplied weighting flux SPEC. This enables the user to produce an ISOTXS library appropriate to a class of problems with a flux similar to SPEC. In general, the incident-energy dependence of the fission spectrum is weak, so the choice of this weighting spectrum is not critical. Noticeable differences might be expected between a thermal spectrum on the one hand and a fast-reactor or fusion-blanket spectrum on the other. The χ vector is defined as follows:

$$\chi_{g'} = \frac{\sum_g \sigma_{fg\to g'}\,\phi_g}{\sum_{g'}\sum_g \sigma_{fg\to g'}\,\phi_g} \ , \tag{403}$$

where σ_f is the fission group-to-group matrix from GROUPR, ϕ_g is either the model flux or SPEC, and the denominator assures that $\chi_{g'}$ will be normalized. Actually, the calculation is more complicated than that because of the necessity to include delayed-neutron production. A "steady-state" value for the fission spectrum can be obtained as follows:

$$\chi^{SS}_{g'} = \frac{\sum_g \sigma_{fg\to g'}\,\phi_g + \chi^D_{g'}\sum_g \bar{\nu}^D_g \sigma_{fg}\,\phi_g}{\sum_{g'}\sum_g \sigma_{fg\to g'}\,\phi_g + \sum_g \bar{\nu}^D_g \sigma_{fg}\,\phi_g} \ , \tag{404}$$

where $\bar{\nu}^D_g$ is the delayed-neutron yield obtained from MF=3,MT=455 on the GENDF tape, χ^D_g is the total delayed-neutron spectrum obtained by summing over the 6 time groups in MF=5,MT=455, and σ_{fg} is the fission cross section for group g obtained from MF=3,MT=18.

This is still not the end of the complications of fission. If the partial fission reactions MT=19, 20, 21, and 38 are present, the fission matrix term in the above equations is obtained by adding the contributions from all the partial reactions found. In these cases, a matrix for MT=18 will normally not be present on the GENDF tape. If it is, it will be ignored. Beginning with NJOY 91.0, a new and more efficient representation is used for the fission matrix computed in GROUPR. It is well known that the shape of the fission spectrum is independent of energy up to energies of several hundred keV. GROUPR takes advantage of this by computing this low-energy spectrum only once. It then computes a fission neutron production cross section for all the groups up to the energy at which significant energy dependence starts. At higher energies, the full group-to-group fission matrix is computed as in earlier versions of NJOY. Therefore, it is now necessary to compute the values of $\sigma_{fg\to g'}$ as used in the above equations using

$$\sigma_{fg\to g'} = \chi^{LE}_{g'}(\bar{\nu}\sigma_f)^{LE}_g + \sigma^{HE}_{fg\to g'} \ , \tag{405}$$

where LE stands for low energy, HE stands for high energy; the low-energy production cross sections written as $\nu\sigma_f$ will be found on the GENDF tape using the special flag IG2LO=0, and the low-energy χ will be found on the GENDF tape with IG=0.

In order to obtain still better accuracy, CCCCR can produce a fission χ matrix instead of the vector. Using the above notation, the full χ matrix becomes

$$\chi^{SS}_{g\to g'} = \frac{\chi^{LE}_{g'}(\bar{\nu}\sigma_f)^{LE}_g + \sigma^{HE}_{fg\to g'} + \chi^D_{g'}\bar{\nu}^D_g\sigma_{fg}}{\text{NORM}} \ , \tag{406}$$

where NORM is just the value that normalizes the χ matrix; that is, the sum of the numerator over all g'. Note that the "Isotope Chi Data" record allows for a rectangular fission matrix similar to the one produced by GROUPR. It is obtained by using the input SPEC array to define the range of groups that will be averaged into each of the final ICHIX spectra. For example, to collapse a ten-group χ matrix into a five-group matrix, SPEC might contain the ten values 1, 2, 3, 4, 5, 5, 5, 5, 5, 5. More formally,

$$\chi^{SS}_{k\to g'} = \frac{\sum_{s(g)=k}\left(\chi^{LE}_{g'}(\bar{\nu}\sigma_f)^{LE}_g + \sigma^{HE}_{fg\to g'} + \chi^D_{g'}\bar{\nu}^D_g\sigma_{fg}\right)\phi_g}{\text{NORM}} \ , \tag{407}$$

where ϕ_g is the default weighting function from the GENDF tape, $s(g)$ is the SPEC array provided by the user, and the summation is over all groups g satisfying the condition that $s(g)=k$. Future versions of CCCCR could construct the SPEC array automatically using the information in the new GENDF format.

Continuing with the description of the "Isotope Control and Group Independent Data" record, the next 9 parameters are flags that tell what reactions will be described in the "Principal Cross Sections" record. They will be described below. Similarly, the parameters IDSCT, LORDN, JBAND, and IJJ will be described later in connection with the "Scattering Sub-Block" records.

The ISOTXS format allows for a fixed set of principal cross sections that was chosen based on the needs of fission reactor calculations. This is one of its main defects; the list does not allow for other reactions that become important above 6–10 MeV, and it does not allow for other quantities of interest, such as gas production, KERMA factors, and radiation damage production cross sections. Most of the reactions are simply copied from MF=3 on the GENDF tape with the group order inverted—this is true for SNGAM, the (n,γ) cross section, which is taken from MT=102; for SFIS, the (n,f) cross section, which is taken from MT=18; for SNALF, the (n,α) cross section, which is taken from MT=107; for SNP, the (n,p) cross section, which is taken from MT=103; for SND, the (n,d) cross section, which is taken from MT=104; and for SNT, the (n,t) cross section, which is taken from MT=105. The (n,2n) cross section, SN2N, is normally taken from MT=16. However, earlier versions of ENDF represented the sequential (n,2n) reaction in ^{9}Be using MT=6, 7, 8, and 9. If present, these partial (n,2n) reactions are added

into SN2N. The flags IFIS, IALF, INP, IN2N, IND, and INT in the "Isotope Control and Group Independent Data" record are set to indicate which of these reactions have been found for this material.

In going from version III of the CCCC specifications to version IV, there was some controversy over the appropriate definition for the (n,2n) cross section and matrix. It was decided that the quantity in SN2N would be the (n,2n) reaction cross section; that is, it would define the probability that an (n,2n) reaction takes place. The (n,2n) matrix would be defined such that the sum over all secondary groups would produce the (n,2n) production cross section, which is two times larger than the reaction cross section.

The CHISO vector, which contains the fission spectrum vector (if any), was discussed above. A complete calculation of the fission source also requires the fission yield, SNUTOT, which can be used together with the fission cross section to calculated the fission neutron production cross section, $\bar{\nu}\sigma_f$. The fission yield can be calculated from the GROUPR fission matrix using

$$\bar{\nu}_g = \frac{\sum_{g'} \sigma_{fg\to g'}}{\sigma_{fg}} \ . \tag{408}$$

Adding delayed neutron contributions and accounting for the partition of the fission matrix into low-energy and high-energy parts (see the discussion of χ above) gives the equation actually used by CCCCR:

$$\bar{\nu}_g^{SS} = \frac{\sum_{g'} \sigma_{fg\to g'}^{HE} + (\bar{\nu}\sigma_f)_g^{LE} + \bar{\nu}_g^D \sigma_{fg}}{\sigma_{fg}} \ . \tag{409}$$

The total cross section produced by GROUPR contains two components: the flux-weighted or P_0 total cross section, and the current-weighted or P_1 total cross section. The P_0 part is stored into STOTPL, and the LTOT flag is set to 1. STRPL contains the transport cross section used by diffusion codes; this is,

$$\sigma_{tr,g} = \sigma_{t1,g} - \sum_{g'} \sigma_{e1,g\to g'} \ , \tag{410}$$

where $\sigma_{t1,g}$ is the P_1 total cross section and $\sigma_{e1,g\to g'}$ is the P_1 component of the elastic scattering matrix, which is obtained from MF=6,MT=2 on the GENDF tape from GROUPR. The flag LTRN is set to 1; that is, no higher-order transport corrections are provided. Direction-dependent transport cross sections are not computed by CCCCR; therefore, ISTRPD is always zero, and the STRPD vectors are missing.

As discussed above, the "Isotope Chi Data" record may be present if the user set ICHIX>1 and supplied a SPEC vector to define how the full χ matrix is to be collapsed into a rectangular χ matrix.

The treatment of scattering matrices in the ISOTXS format is complex and has lots of possible variations. Only the variations supported by CCCCR will be described here. First of all, the scattering data are divided into blocks and subblocks. A block is either one of the designated scattering reactions [that is, total, elastic, inelastic, or (n,2n)] and contains all the group-to-group elements and Legendre orders for that reaction (IFOPT=1), or it is one particular Legendre order for one of the designated reactions and contains all the group-to-group elements for that order and reaction (IFOPT=2). Its actual content is determined by IDSCT and LORDN. If IFOPT=1 has been selected, IDSCT(1)=100 and LORDN(1)=4 would designate a block for the elastic scattering matrix of order P_3 that contains all 4 Legendre orders and all group-to-group elements. If IFOPT=2 has been selected, IDSCT(1)=100 with LORDN(1)=1 would designate a block containing all group-to-group elements for the P_0 elastic matrix, IDSCT(2)=101 with LORDN(2)=1 would designate a block containing the P_1 elastic matrix, and so on. In CCCCR, LORDN is always equal to 1 for IFOPT=2.

The ISOTXS format attempts to pack scattering matrices efficiently. First, all the scattering matrices treated here are triangular because only downscatter is present. And second, because of the limited range of elastic downscatter, only a limited range of groups above the inscatter group will contribute to the scattering into a given secondary-energy group. Therefore, ISOTXS removes zero cross sections by defining bands of incident energy groups that contribute to each final energy group. The bands are defined by JBAND, the number of initial energy groups in the band, and IJJ, an index to identify the position of the ingroup element in the band. The following table illustrates banding for a hypothetical elastic scattering reaction:

Band	Element	JBAND	IJJ
1	1→1	1	1
2	2→2	2	1
	1→2		
3	3→3	2	1
	2→3		
4	4→4	2	1
	3→4		

Note that IJJ is always 1 in the absence of upscatter. This scheme is efficient for elastic scattering, but it is not efficient for threshold reactions because the ingroup element must always be included in the band. This means that lots of zeros must be given for final energy groups below the threshold group. An improved and simplified scheme is used in the MATXS format.

For IFOPT=2, the elements in the table above would be stored in the sequence shown, top to bottom. Each Legendre order would have its own block arranged in the same order. For IFOPT=1, the Legendre order data are intermixed with the group-to-group data. In each band, the elements for all initial groups for P_0 are given, then all initial groups for P_1, and so on through LORDN Legendre orders.

Scattering matrices can be very large. For example, an 80-group elastic matrix can have as many as $80 \times 79/2 = 3160$ elements per Legendre order. That would be 12640 words for a P_3 block using `IFOPT=1`, or four blocks of 3160 words each for `IFOPT=2`. The latter is practical; the former is not. The corresponding numbers for 200 groups would be 79600 and 19900. Both of these numbers are clearly impractical as record sizes. This is where subblocking comes in. If each block is divided up so that there is one subblock for each energy group, the maximum record size is reduced substantially. For `IFOPT=1`, the maximum record size is equal to the number of groups times the number of Legendre orders, or 800 for the 200-group P_3 case. For `IFOPT=2`, the maximum record size is just equal to the number of groups. Although the ISOTXS formats allows subblocks to contain several groups, CCCCR does not. The possible values of `NSBLOK` are limited to 1 and `NGROUP`. In summary, the CCCCR user has four matrix blocking options:

1. *IFOPT=1 and NSBLOK=1.* This produces a single block and a single subblock for each reaction. It is probably the best choice for small group structures (up to about 30 groups). The maximum record size is $n_\ell \times n_g(n_g-1)/2$.

2. *IFOPT=1 and NSBLOK=NGROUP.* This produces a block for each reaction, and each block contains n_g subblocks. The maximum record size is $n_\ell \times n_g$. This is a good choice for larger group structures because it keeps the record size up as compared with option 4 below.

3. *IFOPT=2 and NSBLOK=1.* This produces n_ℓ blocks and subblocks for each reaction. The maximum record size is $n_g(n_g-1)/2$. It has only a modest advantage in the maximum number of groups over option 1. Unless the application that uses the library finds it convenient to read one Legendre order at a time, the user might just as well choose option 2 if option 1 produces records that are too large.

4. *IFOPT=2 and NSBLOK=NGROUP.* This produces n_ℓ blocks for each reaction, each with n_g subblocks. The maximum record size is n_g. The number of groups would have to be on the order of 1000 before this option would be preferred to option 2.

If CCCCR does not have enough memory to process option 1 or 3, the code automatically sets `NSBLOK` to `NGROUP`, thereby activating option 2 or 4, respectively.

Note that the ISOTXS format specifies that the total scattering matrix is the sum of the elastic, inelastic, and (n,2n) matrices [see the definition of `IDSCT(N)`]. This implies that the inelastic matrix must contain the normal (n,n′) reactions from MT=51–91, and also any other neutron-producing reactions that might be present. Examples are (n,n′α), (n,n′p), and (n,3n).

D. BRKOXS

The format for the BRKOXS Bondarenko-type self-shielding factor file can be found in the CCCC reference, in the NJOY manuals, and on the NJOY web site.

The BRKOXS file is used to supply self-shielding factors for use with the Bondarenko method [41] for calculating effective macroscopic cross sections for the components of nuclear systems. As discussed in detail in the GROUPR section, this system is based on using a model flux for isotope i of the form

$$\phi^i_\ell(E,T) = \frac{C(E)}{[\sigma^i_0 + \sigma^i_t(E,T)]^{\ell+1}} \ , \qquad (411)$$

where $C(E)$ is a smooth weighting flux, $\sigma^i_t(E,T)$ is the total cross section for material i at temperature T, ℓ is the Legendre order, and σ^i_0 is a parameter that can be used to account for the presence of other materials and the possibility of escape from the absorbing region (heterogeneity). GROUPR uses this model flux to calculate effective multigroup cross sections for the resonance-region reactions (total, elastic, fission, capture) for several values of σ_0 and several values of T.

When σ_0 is large with respect to the highest peaks in σ_t, the flux is essentially proportional to $C(E)$. This is called infinite dilution, and the corresponding cross sections are appropriate for an absorber in a dilute mixture or for a very thin sample of the absorber. As σ_0 decreases, the flux $\phi(E)$ develops dips where σ_t has peaks. These dips cancel out part of the effect of the corresponding peaks in the resonance cross sections, thereby reducing, or self-shielding the reaction rate. It is convenient to represent this effect as a self-shielding factor; that is,

$$\sigma_{xg}(T,\sigma_0) = f_{xg}(T,\sigma_0) \times \sigma_{xg}(300K,\infty) \ . \qquad (412)$$

In the CCCC system, the f-factors are stored in the BRKOXS file and the infinitely dilute cross sections are stored in the ISOTXS file. These files were originally use to prepare effective cross sections with the SPHINX [82] code, which determined the appropriate T and σ_0 values for each region and group in a reactor problem using equivalence theory together with the user's specifications for composition and geometry. It then read in the cross sections and f-factors from the ISOTXS and BRKOXS files, interpolates in the T and σ_0 tables to obtain the desired f-factors, and multiplies to obtain the effective cross sections.

In the BRKOXS file specification given above, the components of the "File Identification" record are the same as for ISOTXS. The parameters in the "File Control" record are used to calculate the sizes and locations of data in the records to follow. The "File Data" record contains all the isotope names, all the σ_0 values for every isotope, all the T values for every isotope, the group structure, and several arrays used for unpacking the other data.

The Hollerith material names in `HISONM` are the same as those used in ISOTXS, and they are obtained from the user's input. The σ_0 values are packed into `X(K)` using `NTABP(I)`. Note that the base-10 logarithm of σ_0 is stored. Therefore, a typical library might contain the following:

material	`I`	`NTABP(I)`	`X(K)`
1	3		3.0 2.0 1.0
2	7		4.0 3.0 2.0 1.477 1. 0. -1.
3	5		4.0 3.0 2.0 1.0 0.0
...	...		...

Similarly, the T values are stored in `TB(K)` using `NTABT(I)`. The absolute temperatures used by GROUPR must be changed to degrees C before being stored. The energy bounds for the group structure found on the GENDF tape are stored in `EMAX(J)` and `EMIN`. As discussed in connection with the ISOTXS format, group boundaries are stored in order of decreasing energy in the BRKOXS file.

The self-shielding factor approach is designed to account for resonance self-shielding. It is not necessarily appropriate for low energies if only broad resonance features are apparent, or for high energies where only small residual fluctuations in the cross section are seen. For this reason, the BRKOXS format provides the `JBFL(I)` and `JBFH(I)` arrays in the "File Data" record. They are used to limit the range of the numbers given in the "Self-Shielding Factors" record. The reactions that are active in the resonance energy range usually include only the total, elastic, fission, and capture channels. The total cross section is usually computed for the two Legendre orders ℓ=0 and ℓ=1. This second value is often called the current-weighted total cross section, and it is needed to compute the self-shielded diffusion coefficient. GROUPR also computes a self-shielded elastic scattering matrix. It can be used to provide two quantities for the BRKOXS file. First, the diffusion coefficient requires the calculation of a transport cross section for diffusion. The relationships are as follows:

$$D_g = \frac{1}{3\sigma_{tr,g}} \ , \tag{413}$$

and

$$\sigma_{tr,g} = \sigma_{t1,g} - \sum_{g'} \sigma_{e1,g\to g'} \ . \tag{414}$$

Therefore, the current-weighted P_1 elastic cross section contributes to the transport self-shielding factor. The second use for the self-shielded elastic scattering matrix is to compute a self-shielding factor for elastic removal. For heavy isotopes, the energy lost in elastic scattering is small, and all the removal is normally to one group. The format requires that at least the five standard reactions be given in a specified order. NJOY is able to add one more as follows:

1. total (P_0 weighted),
2. capture (P_0),
3. fission (P_0),
4. transport (P_1),
5. elastic (P_0), and
6. elastic removal (P_0).

The normal pattern for the BRKOXS file expects that there will be one record of self-shielding factors for each material. Such a record could get quite large. For example, with 6 reactions, 6 temperatures, 6 σ_0 values, and 100 resonance groups, a "Self-Shielding Factors" record could have over 20000 words. This number can be made more manageable by setting the `IBLK` flag in the "File Control" record to 1. Then there will be a separate record of self-shielding factors for each reaction; this would reduce the record size for the example to a more reasonable 3600 words.

The actual self-shielding factors are computed from the cross sections given on the GENDF tape and stored into the `FFACT(N,K,J,M)` array of the "Self-Shielding Factors" record with group order converted to the standard decreasing-energy convention. Following the FORTRAN convention, `N` is the fastest varying index in this array, `K` is the next fastest varying index, and so on. The identities of these indices are

`N` σ_0,

`K` temperature T,

`J` group, and

`M` reaction.

The "Cross Sections" record contains some additional cross sections and special parameters that are often used in self-shielding codes and are not included in the ISOTXS file. The `XSPO` cross section is taken to be constant and equal to the CCCCR input parameter `XSPO`, which is computed as $4\pi a^2$. The `XSIN` cross section is obtained by summing over the final group index for every group-to-group matrix in File 6 except elastic, (n,2n), and fission. Therefore, it may contain effects of multiplicities greater than 1 if reactions like (n,3n) or (n,2nα) are active for the material. It may be slightly larger in high-energy groups than the cross section that would be obtained using the same sum over reactions in File 3. The `XSE` cross section is obtained from MF=6,MT=2 on the GENDF tape by summing over final groups. The parameters for continuous slowing down theory, `XSMU` and `XSXI`, are obtained from MT=251 and MT=252, respectively. The methods used for calculating these quantities are described in the GROUPR section of this report. Finally, the elastic downscattering cross section is obtained from the elastic matrix on the GENDF tape (MF=6,MT=2).

E. DLAYXS

The format for the DLAYXS delayed-neutron data file will be found in the CCCC reference, the NJOY manuals, and the NJOY web site.

This file is used to communicate delayed-neutron data to reactor kinetics codes. The ENDF files give a total delayed-neutron yield $\bar{\nu}_d$ in the section labeled MF=1,MT=455. GROUPR averages this yield for each neutron group g using

$$\bar{\nu}_g^D = \frac{\int_g \nu_d(E)\sigma_f(E)\phi(E)\,dE}{\int_g \sigma_f(E)\phi(E)\,dE} \,. \tag{415}$$

This same section of the ENDF tape contains the decay constants for several delayed-neutron time groups. ENDF/B evaluations normally use 6 time groups, but the European evaluations use 8. These numbers are passed on to the GROUPR routine that averages the delayed-neutron spectra, and they end up in the MF=5 record on the GENDF tape. The ENDF/B evaluations give the delayed-neutron spectra for the six time groups in MF=5,MT=455. These spectra are not separately normalized. Instead, the sum over all six time groups is normalized, and the integral of any one of the spectra gives the "delayed fraction" for that time group. GROUPR simply averages these spectra using the specified neutron group structure. The resulting group spectra (and the decay constants from File 1) are written onto the GENDF tape.

Returning to the DLAYXS format description, the parameters in the "File Identification" record are obtained from the user's input, just as for ISOTXS and BRKOXS. The parameter `NGROUP` is also a user input quantity. `NISOD` is determined after the entire GENDF tape has been searched for isotopes that are on the user's list of `NISO` materials and that have delayed-neutron data. The `NFAM` parameter needs some additional explanation. A "family" for the DLAYXS file is actually an index that selects one particular spectrum from the `CHID` array. It could correspond to an actual delayed-neutron precursor isotope left after a fission event. In such a case, there would be many "families" corresponding to the many possible fission fragment isotopes. The `SNUDEL` yields would be analogous to fission product yields. The ENDF evaluations take a more macroscopic approach. Spectra are chosen to include all the emissions from a group of delayed-neutron precursors for a particular fissioning nucleus with similar decay constants. In this representation, the DLAYXS "family" corresponds to a particular decay constant and spectrum for a particular target isotope. Therefore, the number of families for a typical ENDF/B evaluation is simply 6 times the number of isotopes, or `NFAM=6*NISOD`.

In the "File Data, Decay Constants, and Emission Spectra" record, the Hollerith isotope names `HABSID` are obtained from the names in the user's input. The `FLAM` values come directly from the decay constant values originally extracted from File 1 of the ENDF tape. The family index for isotope `ISO` and time group `I` is simply computed as `6*(ISO-1)+I`. The `CHID` array is loaded from the MF=5,MT=455 section on the GENDF tape using the family index and the group index. As usual, the order of groups has to be changed from the GROUPR convention with increasing-energy order to the CCCC convention with decreasing-energy order. The group structure itself is obtained from the GENDF header record and stored into `EMAX` and `EMIN` in the conventional order. The value of `NKFAM` is simply 6 for every isotope. The `LOCA` values are also easy to compute; they are just `ISO` minus 1.

The delayed-neutron yields versus incident neutron energy group and family index are given in the "Delayed Neutron Precursor Yield Data" record. As mentioned above, the total yield is given in MF=3 of the GENDF tape, and the delayed fractions can be computed by summing the spectrum for each time group over the energy group index. The array `SNUDEL` in this DLAYXS record contains the product of these values. Note that there is one of these yield records for each delayed-neutron isotope, and each record contains six families. The `NKFAM` vector is used to establish the correspondence between these families and the entire list of families used in the file data record. As an example, `NUMFAM(1)` would contain 1, 2, 3, 4, 5, 6; `NUMFAM(2)` would contain 7, 8, 9, 10, 11, 12, and so on.

F. Running CCCCR

The user input instructions will be found in the comment cards at the beginning of the CCCCR source code, in the NJOY manuals, and on the NJOY web site. It is always a good idea to check the comment cards in the current version of the code in case there have been any changes.

The instructions are divided into four parts. First, there is a general section that applies to all three interface files following `-CCCCR-`. Second, there is a section of special parameters for ISOTXS following `-CISOTX-`. Third, there is a section of special parameters for BRKOXS following `-CBRKXS-`. And fourth, there is a comment following `-CDLYXS-` noting that no special input is required for the DLAYXS file.

In the `-CCCCR-` section, Card 1 is used to read in the unit numbers for input and output. `NIN` must be a GENDF tape prepared using GROUPR, and it can have either binary (`NIN>0`) or BCD (`NIN<0`) mode. The units for the output ISOTXS, BRKOXS, and DLAYXS files are all binary, but either sign can be used on the unit numbers. If any unit number is given as zero, the corresponding CCCC interface file will not be generated. On Card 2, the `LPRINT=1` is used to request a full printout of all the CCCC files generated. The file version num-

ber IVERS can be used to distinguish between different libraries generated using NJOY. The user identification field HUSE can contain any desired 12-character string. An example of Card 2 might be

```
0  4  'T2 LANL NJOY'/  .
```

Card 3 contains a description of the library using up to 72 characters (12 standard CCCC words of 6 characters each). For example,

```
'LIB-IV 50-GROUP LMFBR LIBRARY FROM ENDF/B-IV (1976)'/
```

On Card 4, the values given for NGROUP must agree with the number of groups on the input GENDF tape or a fatal error message will be issued. NGGRUP is not used. NISO is the total number of materials or isotopes to be searched for on the GENDF tape. The value of MAXORD should be less than or equal to the maximum Legendre order used in the GROUPR run. The use of the matrix blocking parameter IFOPT was discussed in detail in connection with the description of the ISOTXS format above. A value of 1 has been normally used for Los Alamos libraries.

A line using the Card 5 format is given for each of the NISO isotopes or materials to be processed. The Hollerith isotope label and the Hollerith absolute isotope label have normally been set to the conventional isotope name at Los Alamos; for example, U235 or FENAT. The library name in HABSID can vary quite a lot now that many other libraries are available in ENDF format. The values of HMAT and IMAT will normally be derived from the MAT number characteristic of all libraries in ENDF format. Unfortunately, XSPO, the average potential scattering cross section, must be entered by hand. It can be obtained from MF=2,MT=151 on the ENDF file for the material by determining the scattering radius a from the AP field and computing $\sigma_p=4\pi a^2$. The following fragment of an ENDF/B evaluation shows the vicinity of AP:

```
  ...
 9.223500+4 2.330250+2          0  ... 9228 2151  573
 9.223500+4 1.000000+0          0  ... 9228 2151  574
 1.000000-5 1.100000+2          1  ... 9228 2151  575
 3.500000+0 1.001760+0          0  ... 9228 2151  576
 2.330250+2 0.000000+0          0  ... 9228 2151  577
-9.997600+1 3.000000+0 3.771300-3 3... 9228 2151  578
  ....
```

The scattering radius is the second number on the fourth card in the section MF=2,MT=151. Using it as a gives $4\pi(1.00176)^2 = 12.611$ barns for the value of XSPO. An example of Card 5 follows:

```
U235 U235 ENDF6 '9228' 9228 12.611
```

Note that the Hollerith string "9228" had to be delimited by quotes because it does not begin with a letter. The delimiters are optional for the other Hollerith variables.

The next block of input cards is specific to ISOTXS and only appears if ISOTXS processing was requested with a nonzero value for NISOT. The first parameter on Card 1 is NSBLOK, which was discussed in connection with matrix blocking and subblocking. NSBLOK and IFOPT work together to control how a large scattering matrix is broken up into smaller records on the ISOTXS interface file. The important factor is the maximum size of the binary records. They should be small enough to fit into a reasonable amount of memory in any application codes that use ISOTXS files, but they should be large enough to keep the number of I/O operations to a minimum. The maximum record size for each option is repeated below for the convenience of the reader:

1. *IFOPT=1 and NSBLOK=1*: $n_\ell \times n_g(n_g-1)/2$,
2. *IFOPT=1 and NSBLOK=NGROUP*: $n_\ell \times n_g$,
3. *IFOPT=2 and NSBLOK=1*: $n_g(n_g-1)/2.$,
4. *IFOPT=2 and NSBLOK=NGROUP*: n_g,

where n_ℓ is the number of Legendre orders and n_g is the number of groups. If the user specifies NSBLOK=1 and the resulting output record is too large for the available memory, NSBLOK will be changed to NGROUP automatically.

Card 1 of the ISOTXS section continues with MAXUP. This parameter is always zero for CCCCR; thermal upscatter matrices are not processed. The normal value of MAXDN is NGROUP, but it can be made smaller to reduce the size of the matrices. The cross section for any removed downscatter groups will be lumped into the last group in order to preserve the production cross section. The ICHIX is used to control the generation of fission χ vectors and matrices. The representations allowed were discussed above in connection with the description of the ISOTXS format (see Sect. XVIII C). The most commonly used option has been ICHIX=-1 because most application codes cannot handle fission χ matrices. The GROUPR flux is normally chosen to be characteristic of the class of problems a given library is intended to treat; therefore, it is rarely necessary to supply an input weighting spectrum (see ICHIX=+1 and SPEC). The following scenario illustrates a case where this option might be useful. Assume that an 80-group library is made using the GROUPR fast reactor weight function (IWT=8), which contains a shape in the fission range typical of both fast reactors and fusion blankets plus a fusion peak at 14 MeV. This GENDF library could be used to generate two different ISOTXS libraries, one using the default flux and useful for fusion problems, and one using a spectrum SPEC from which the fusion peak has been removed. The latter would be better for fast reactor analysis because the χ vectors would not contain the component of high-energy fission from the 14 MeV range. Card 2 is used to input the user's choice for SPEC.

CCCCR can also produce fission χ matrices for codes that can use them. These matrices can be rectangular to take advantage of the fact that the $\chi_{g\to g'}$ function is basically independent of g at low energies (large values of g). Taking the GROUPR 30-group structure as an example, if the energy at which significant incident-energy

dependence begins is taken to be about 100 keV, then groups 16 through 30 will have identical χ vectors. The value of ICHIX should be set to 16, and the SPEC vector of Card 3 should be set to

```
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 15R16/  .
```

The resulting χ matrix will be rectangular with 16×30 elements.

Card 4 completes the input specific to ISOTXS. The value of KBR can be set to reflect the normal use of this material in the applications that this library is intended to treat. The AMASS parameter has units of gram atomic weight. It can be calculated from the normal ENDF AWR parameter (the atomic weight ratio to the neutron) by multiplying by the gram atomic weight of the neutron. TEMP is normally 300 K for NJOY libraries. The same value can be used for SIGPOT and XSPO (see above). The ADENS parameter has no meaning for CCCCR; it can be set to zero to imply infinite dilution.

The choice of values for EFISS and ECAPT is more complicated. As discussed in Sect. XVIII C, EFISS is basically the total non-neutrino energy released by a fission reaction. It is available in eV as the pseudo Q value in MF=3,MT=18 (the energy release from fission is given in more detail in MF=1,MT=458). The following fragment of the ENDF/B-VI evaluation for ^{235}U shows how to find the Q value:

```
 ...
9.223500+4 2.330250+2          0  ... 9228 3 18 4414
1.937200+8 1.937200+8          0  ... 9228 3 18 4415
       333          2             ... 9228 3 18 4416
1.000000-5 0.000000+0 7.712960+1 0... 9228 3 18 4417
2.250000+3 5.362770+0 2.300000+3 5... 9228 3 18 4418
 ....
```

The Q value is the second number on the second card of the section MF=3,MT=18. Converting to CCCC units gives

193.72×10^{6} eV $\times$ 1.602×10^{-19} watt-s/eV = $.31034\times10^{-10}$ watt-s/fission

The value for ECAPT is determined from the Q value for MF=3,MT=102, the radiative capture reaction. However, if the isotope remaining after capture has a relatively short half-life, the energy of the decays leading to the final stable daughter should be added onto the capture Q value. (The meaning of "stable" may vary from application to application). As an example, consider aluminum. The ^{28}Al capture product decays with a half-life of 2.24 minutes producing 9.31 MeV of β^- energy and a 1.779 MeV photon. Therefore, the actual value of ECAPT should be calculated as follows:

$$\begin{array}{rl} 7.724 \text{ MeV} & \text{MT=102 prompt Q value} \\ 9.310 \text{ MeV} & \beta^- \text{ energy} \\ 1.779 \text{ MeV} & \text{delayed-}\gamma \text{ energy} \\ \hline 18.813 \text{ MeV} & \\ \times 1.602 \times 10^{-13} & \\ \hline .3014 \times 10^{-11} & \text{in watt-s/capture} \end{array}$$

The next section of the input file is specific to BRKOXS. Card 1 enables the user to just accept all or part of the temperatures and sigma-zero values on the input GENDF tape. If the value of NTI is negative, the first abs(NTI) T values for each material will be used. If fewer values are available, only those will be used. If NTI is positive, input Card 2 will be read for a list of T values, and only data with temperatures on that list will be extracted from the GENDF tape. The parameter NZI and the list of σ_0 values on Card 3 work in the same way.

No additional input is required for DLAYXS files.

XIX. MATXSR

The MATXS material cross section format is a generalized CCCC-type interface format for neutron, photon, and charged-particle data, including cross sections, group-to-group matrices, temperature variations, self-shielding, and time-dependence. The CCCC standards are discussed in more detail in the CCCCR section of this manual and in the CCCC-III and CCCC-IV reports [8, 77]. MATXS libraries can be used with the TRANSX code [9, 43] to produce effective cross sections for a wide variety of application codes.

A. Background

Even the very best nuclear cross section processing code would be useless if it were unable to deliver its products to users. This is the role of the interface file. There have been interface files since the beginning of calculational neutronics; examples include the DTF format (see the DTFR section of this manual) that was devised for the early discrete-ordinates transport code DTF-IV [75], and the CCCC ISOTXS format [8] (see the CCCCR section of this report). Both of these interface formats are still in use today, but both of them have problems and show their age. Some of these problems result from the increase in the capabilities of computer systems (capabilities that allow us to consider much more complex problems), some arise from the many new kinds of nuclear systems that are being studied today, and some come from 20/20 hindsight, which makes it easy to see the design flaws in earlier formats.

Based on the problems seen with existing interface files, an ideal interface file should be

extendable, so that new cross section types, new incident or secondary particle types, or new energy ranges are easy to add without changing the basic format;

comprehensive, in order to be able to handle as many of the kinds of data produced by the processing code as possible (results should not be lost just because there are no places for them);

generalizable, to allow common methods to be used for similar kinds of data (for example, *nn* matrices and $\gamma\gamma$ matrices) in order to transfer the experience gained in one field to another, and in order to simplify coding by allowing components to be reused;

self-contained, because it should not be necessary to provide additional information that is not in the file in order to use or interpret the file;

compact, because nuclear data often have many zeros or very small numbers in tables (for example, threshold reactions, scattering matrices), and these zeros must be removed effectively for economic storage and fast transfer of libraries; and

efficient, thus implying that binary mode should be used, that the records have a well-defined maximum size, and that there is a minimum number of records to reduce the number of I/O operations.

Comparing the DTF format to these principles gives the following results: it is fairly *extendable* because it has no fixed particles, energy limits, or reaction types; it is not very *comprehensive* because it can only transmit the total scattering matrix; it is fairly *generalizable* because of the lack of fixed types; it is not at all *self-contained* in that it requires outside definitions like table length, position of the total, group structure, and identity of edit cross sections; it is not very *compact* because most zeros must be given explicitly in the tables; and it is not very *efficient* because it uses coded card-image records.

Similarly, studying the ISOTXS format gives the following results: it is not *extendable* because it works for neutrons only and allows only very limited types of reactions to be included; it is not *comprehensive* because it works for neutrons only and allows only very limited types of reactions to be included; it is not *generalizable* because of its specialization to fast-reactor problems (as proof, note that the CCCC files for γ cross sections use completely different formats); it is reasonably *self-contained* because all the parameters are internal, the group structure is given, and all names needed for labeling an interpreted listing are well determined; it is fairly *compact* because many zeros are removed (but too many still remain); it is fairly *efficient* because it uses binary mode, but record sizes are poorly predictable and very nonuniform, thereby needlessly increasing the size of application codes and the number of I/O operations.

B. The MATXS Format

Following these general principles, the MATXS material cross section file was designed to extend and generalize the existing interface formats while still using the CCCC approach for efficiency and familiarity (see the CCCCR section of this report for more details). The first design principle was that all information would be identified using lists of Hollerith names. As an example, if the list of reactions included in the file contains entries such as `nf`, `ng`, and `n2n`, it is trivial to add additional reactions such as `kerma` or `dpa`. This approach is much more extendable than the fixed set of reaction flags used in ISOTXS. The second design principle was that the file would be designed to hold sets of vectors and rectangular matrices and that the same format would be used regardless of the contents of the vectors and matrices. As a consequence, once a code can handle $n\rightarrow n$, it can also handle $\gamma\rightarrow\gamma$; once a code can handle $n\rightarrow\gamma$, it can also handle $\gamma\rightarrow n$, $n\rightarrow\beta$, or even $d\rightarrow p$. This approach is an example of generalization. Each material is now divided into data types identified by input and output particle. As an example, $n\rightarrow\gamma$ is a data type characterized by input particle equals neutron, and output particle equals photon. The matrices for this data type contain cross sections for producing photons in photon group γ due to reactions of neutrons in group n. The vectors, if any, would contain photon production cross sections versus neutron group. The use of completely general data types helps make the format comprehensive.

TABLE XI: Standard particle names.

Name	Particle
`n`	neutron
`g`	gamma
`p`	proton
`d`	deuteron
`t`	triton
`h`	^{3}He nucleus
`a`	alpha
`b`	beta
`r`	residual or recoil (heavier than α)

The names for materials are written in the forms `u235`, `fe56`, `tinat`, or `h2a`. Note that "`nat`" is used explicitly for elements; names like `be` or `c` should be avoided. Suffixes "`a`", "`b`", or "`c` are used to label different versions of a material in a library. In order to keep names to six characters, isomers should be identified by incrementing the "thousands" digit in the atomic number field; for example, `nb193a` would be the second version of the first isomer of ^{93}Nb. The standard names for MATXS particles are given in Table XI.

The standard names for the data types (`htype`) are mostly based on these particle names; the use of the terms `scat`, `dk`, `therm` are exceptions. Table XII illustrates the scheme used.

Reactions names are constructed in MATXSR from the ENDF MT number, the LR flag (if present), and the incident particle name. Examples of the standard names are given in Tables XIII–XXI. Note that the first `n` is omitted from the last three reactions in Table XIII. It is implicit in the data-type name. This convention saves space in

TABLE XII: Standard data-type names.

Name	Data Type
`nscat`	neutron scattering
`ng`	neutron-induced gamma production
`np`	neutron-induced proton production
`nr`	neutron-to-recoil matrix
`gscat`	gamma scattering
`pscat`	proton scattering
`pn`	proton-induced neutron production
...	...
`ntherm`	thermal scattering data
`dkn`	delayed-neutron data
`dkhg`	decay heat and gamma data
`dkb`	decay beta data

TABLE XIII: Simple neutron-emitting reactions.

Name	MT	Description
`nelas`	2	neutron elastic scattering
`nnonel`	3	neutron nonelastic (MT=1–MT=2)
`ninel`	4	neutron inelastic sum (MT=51–91)
`n2n`	16	(n,2n)
`n3n`	17	(n,3n)
`nna`	22	(n,n′α)
`nnp`	28	(n,n′p)
`n01`	51	(n,n$_1$)
`n02`	52	(n,n$_2$)
`ncn`	91	(n,n′) to continuum

the name for possible breakup products (see Table XIV). The first `n` is also implicit in the reactions of Table XIV. No multiplicity is used in the breakup product strings in order to avoid possible confusions with the discrete-level number; just count the like letters to get the multiplicity. The names used for the most common neutron absorption reactions are given in Table XV, and the names used for the fission reactions are given in Table XVI. MATXS libraries typically give the total fission cross section and all the partial cross sections (when available) in the vector blocks, but they do not give the total fission matrix when the partial matrices are available.

Table XVII gives some additional reaction names,

TABLE XIV: Breakup reactions (LR flags).

Name	MT	LR	Description
`n07a`	57	22	(n,n$_7$)α
`n51p`	65	28	(n,n$_{15}$)p
`n02aa`	52	29	(n,n$_2$)2α
`ncnaaa`	91	23	(n,n′)3α
`n06na`	56	24	(n,n$_5$)nα
`n01ee`	51	40	(n,n$_1$)ee

TABLE XV: Neutron-absorption reactions.

Name	MT	Description
`nabs`	101	total absorption
`ng`	102	radiative capture
`np`	103	(n,p)
`na`	107	(n,α)

TABLE XVI: Fission reactions.

Name	MT	Description
`nftot`	18	total fission
`nf`	19	(n,f) first-chance fission
`nnf`	20	(n,n′f) second-chance fission
`n2nf`	21	(n,n2f) third-chance fission
`n3nf`	38	(n,n3f) fourth-chance fission
`nudel`	455	delayed-neutron yield (MF=3)
`chid`	455	delayed-neutron spectrum (MF=5)

some of which are special NJOY names. As discussed in the GROUPR section of this report, the total cross section can be averaged with the ℓth order of the flux to obtain the multigroup total cross section components $\sigma_{t\ell,g}$. These total cross section components and the corresponding Legendre fluxes are given names like the first four shown in Table XVII. Related names with first letters `g`, `p`, `d`, *etc.*, will also be found in MATXS libraries. The average inverse velocities are defined to preserve the time term of the time-dependent Boltzmann equation

$$\left\langle \frac{1}{v} \right\rangle = \frac{\int_g (1/v)\phi(E)\,dE}{\int_g \phi(E)\,dE} \ . \tag{416}$$

The meaning of the terms energy-balance heat production and kinematic KERMA factor are discussed in more detail in the HEATR section of this manual. Briefly, the energy-balance heating (MT=301) is computed by subtracting the energy carried away by neutrons and photons from the energy available for a reaction. The result should be the energy deposited by charged particles and the recoil nucleus, that is, the local heating. Unfortunately, problems with the energy-balance consistency of evaluations, the difficulty of determining the available energy for elements, and the inaccuracy in the difference between relatively large numbers sometimes cause this value to have unphysical values (for example, negative heating). These values do have the property of always conserving energy for large systems. The kinematic value (MT=443) is computed from reaction kinematics alone. It is very accurate at low energies, but when three or more particles are involved in the reaction, it begins to fail. The results in `kerma` are always larger than the correct heating value. Comparing the two estimates for the local

heating can reveal problems in the evaluations [20, 86]. The MATXS user is free to choose whichever number is more appropriate for the problem. The reaction `dame` is also generated using data from HEATR. As discussed in the HEATR section of this report, this "damage-energy production" cross section can be used to obtain the DPA (displacements per atom) parameter used in radiation damage studies.

The gas-production reaction names given in Table XVIII can also appear with other particle names before the decimal point. The names for reactions induced by incident charged particles follow the neutron names in most cases, except that the first letter is changed to indicate the incident particle type. Discrete-level scattering reactions are exceptions; `n01` is used for both (n,n_1) and (p,n_1). Also note the `n00` cannot be used for incident neutrons; the name `nelas` is used instead. Similarly, `p00` is not used for incident protons.

As discussed in more detail below, scattering from thermal moderators is treated like *materials* in ENDF/B libraries, but it is treated like *reactions* on the GENDF files. The free-gas scattering reaction can appear in any material, but the other thermal MT numbers can only appear in the material corresponding to the dominant scattering isotope. For example, `h2o` only appears in H1. There are two versions of `zrhyd`; one appears in H1 and the other in Zr. The coherent and incoherent terms in the thermal cross section are kept separate for the convenience of applications; all the coherent names end with $. Note that MATXS files contain two different representations for the scattering cross sections at low energies:

static, where the cross section and group-to-group matrix are obtained from `nscat`, which is derived from MT=2 on the ENDF evaluation (This is scattering for "static" nuclei; energy loss from recoil is included.); and

thermal, where the cross section and group-to-group matrix are obtained from one of the thermal reactions in the `ntherm` data type. (The scattering nuclei are in motion with a distribution described by the Maxwell-Boltzmann law; both energy loss and energy gain events are possible.)

The TRANSX code gives the user the choice of static or thermal scattering, and it also allows the user to choose which binding state is desired for a particular moderator material.

The ENDF representation of photoatomic reactions is described in the GAMINR section of this report. The `gheat` reaction is constructed in GAMINR to represent the local heating due to atomic recoil and the photoelectric production of electrons. Fluorescence photons from photoelectric interactions are assumed to deposit their energy locally.

GROUPR and MATXSR are capable of supporting a new experimental capability for generating nuclide production cross sections. This capability is most useful for

TABLE XVII: Special NJOY names.

Name	MT	Description
`ntot0`	1	P_0 total cross section
`ntot1`	1	P_1 total cross section
`nwt0`	1	P_0 weight function (flux)
`nwt1`	1	P_1 weight function (flux)
`mubar`	251	scattering $\bar{\mu}$
`xi`	252	scattering ξ
`invel`	259	inverse velocity (sec/m)
`heat`	301	energy-balance heat production
`kerma`	443	kinematic KERMA factor
`dame`	444	damage-energy production

TABLE XVIII: Gas-production reactions.

Name	MT	Description
`n.neut`	201	total neutron production
`n.gam`	202	total γ production
`n.h1`	203	hydrogen production
`n.h3`	205	tritium production
`n.HE4`	207	helium production

TABLE XIX: Incident-proton reactions.

Name	MT	Description
`pelas`	2	proton elastic scattering
`p01`	601	discrete-level (p,p_1) scattering
`n00`	50	discrete-level (p,n_0)
`n01`	51	discrete-level (p,n_1)
`p2n`	16	(p,2n)
`pg`	102	(p,γ)
`pt`	104	(p,t)

TABLE XX: Thermal cross sections.

Name	MT	Description
`free`	221	free-gas scattering
`h2o`	222	H in H_2O
`poly`	223	H in polyethylene (CH_2) incoherent
`poly$`	224	H in polyethylene (CH_2) coherent
`zrhyd`	225	H in ZrH incoherent
`zrhyd$`	226	H in ZrH coherent
`benz`	227	Benzene incoherent
`d2o`	228	D in D_2O
`graph`	229	C in graphite incoherent
`graph$`	230	C in graphite coherent
`be`	231	Be metal incoherent
`be$`	232	Be metal coherent
`beo`	233	BeO incoherent
`beo$`	234	BeO coherent
`zrhyd`	235	Zr in ZrH incoherent
`zrhyd$`	236	Zr in ZrH coherent

TABLE XXI: Photoatomic cross sections.

Name	MT	Description
gtot0	501	P_0 total
gwt0	501	P_0 weight function (flux)
gcoh	502	coherent scattering
ginch	504	incoherent scattering
gpair	516	pair production ($\gamma, 2\gamma$)
gabs	522	photoelectric absorption
gheat	525	heating

radionuclides and isomers, but it is general enough to handle all the possible heavy products of a nuclear reaction. The input GENDF file may contain several different sections that produce a given nuclide. MATXSR adds them up into a single named reaction. The naming convention used for capture reactions is `cZZAAA`, where Z and A are the charge and mass numbers for the nuclide. Isomers are handled by incrementing the first position of the "AAA" field. Products of other reactions are named using the pattern `rZZAAA`, with the same convention used for isomers. The reason that capture products are distinguished from those from other reactions is that the former may have to be self shielded.

The CCCC standards have always used 6-character Hollerith strings for names. These kinds of names are represented as "`REAL*8`" double precision variables on 32-bit machines (IBM, VAX, Sun, etc.) and as single-precision variables on 60- to 64-bit machines (CDC, Cray). However, a double-precision variable on a short-word machine can hold 8 characters. So can single-precision variables on CDC and Cray machines. There do not seem to be any computer systems currently in use that require 6-character words. Therefore, the latest versions of the MATXS format and the MATXSR module have been written to handle 8-character names.

The formal format specification for the MATXS material cross section file can be found in the NJOY manuals or on the NJOY web site.

The MATXS format is intended to communicate multigroup cross sections and matrices for all reaction types, incident particles, and outgoing particles from a nuclear data processing code to applications. It also includes temperature and self-shielding effects, delayed-neutron data, and a limited format for decay heat and delayed photon or particle emission. As shown in the "File Structure" presentation above, the main loop is over material. Materials are subdivided into submaterials, which usually correspond to different data types, temperatures, and background cross section (σ_0) values. Each submaterial can contain a series of "Vector Blocks" giving cross section versus energy for one of the allowed group structures and incident particles, and it can contain a series of matrix blocks and subblocks giving the cross sections for group-to-group transfers.

The "File Identification" record is the same for all CCCC files. It gives the Hollerith name for the file (which is always `matxs`), a version number `ivers`, which can be used to distinguish between different libraries in this format, and a Hollerith identification string `huse`, which can be used for entries like "`T2 LANL NJOY`".

The "File Control" record contains parameters that are needed to compute the lengths of the following records. The meaning of the various names is well explained in the format specification, and the values of the parameters are obtained from the user's input. The purpose of `maxw` is to tell application codes how much memory they will need to read through the records on these MATXS files. It is used for both vector blocks and matrix subblocks when deciding how to break them up. The MATXSR value is 5000 words. The code tries to make as many records as possible that have nearly this size in order to minimize the number of I/O operations. The parameter `length` is used to help find the end of the file when appending a new material to an existing file. Its units are left unspecified in the format. It is usually the length in records, in which case record skipping can be used to find the end. Or it could be the length in words on computer systems that allow direct word-addressed I/O operations.

The "Set Hollerith Identification" record comes next. It contains an arbitrary amount of Hollerith text to describe the contents of the library. The description comes from the user's input.

The "File Data" record contains a number of important arrays that define the structure of data types and the location of materials. The parameter `hprt` contains the standard names for the `npart` particles. The standard names for particles were discussed above. The standard names for the data types and materials (`htype` and `hmatn`) were also discussed above. The next three parameters are used to complete the specification of the data types included in this MATXS library. `ngrp` just gives the number of groups used for each particle type; for example, the traditional Los Alamos 30×12 library (with 30 neutron groups and 12 photon groups) would have `ngrp(1)=30` and `ngrp(2)=12`. Using the same example, the `nscat` data type would have `jinp(1)=1` and `joutp(1)=1`; the `ng` data type would have `jinp(2)=1` and `joutp(2)=2`; and the `gscat` data type would have `jinp(3)=2` and `joutp(3)=2`. The information for these 6 arrays is given in the user's input.

The final two parameters in the "File Data" record are `nsubm` and `locm`. The value of `nsubm` depends on the number of data types, the number of temperatures, and the number of background cross sections found on the input GENDF tapes. The value for `locm(i)` is usually the record index for material `i`. A code can then jump directly to a desired material using record skipping (forward or backward). However, the units for `locm` have been left unspecified to allow direct random access for systems that use word-addressable random-access I/O operations (such as CTSS on Cray computers).

The "Group Structure" records give the energy bounds for `npart` group structures. Following the normal convention for application codes, the energy bounds are given in the order of decreasing energy. The numbers are ob-

tained from MF=1,MT=451 on the GENDF tape. The GROUPR module currently uses one group structure for all particles (n, p, α, etc.), and another for photons (γ).

Inside the material loop, there is a "Material Control" record for each material. The choice of names for `hmat` was discussed above. The `amass` parameter is the same as the ENDF AWR parameter; that is, it is the ratio of the target mass to the neutron mass. Temperatures `temp` are given in degrees Kelvin, and background cross sections for self-shielding codes (`sigz`, or σ_0) are given in barns. The parameter `itype` tells which data-type each submaterial belongs to using the data type codes defined by `htype`. Although it is not specified in the format description, the order that MATXSR loops through submaterials is as follows: the outer loop is over data type, the next loop is over temperature, and the innermost loop is over background cross section. The number of cross section vector reactions and matrix reactions for each submaterial are given in `n1d` and `n2d`. Finally, `locs(i)` normally gives the record index for submaterial `in`. A code can search the arrays `temp`, `sigz`, and `itype` for a desired submaterial, and then jump right to the desired submaterial using record skipping. Alternatively, a version that uses word addresses in `locs` could use random-access methods to jump to the desired submaterial.

Each submaterial that contains vectors (`n1d>0`) starts with a "Vector Control" record. This record gives a list of reaction names in `hvps`. MATXSR constructs these reaction names automatically based on the MT number, ENDF "LR flag" (if any), and the incident particle type. Examples of these names were given above. The parameters `nfg` and `nlg` are used to remove unnecessary leading or trailing zeros in reaction cross section vectors. The zeroes are usually due to thresholds. For example, an (n,2n) reaction might only have nonzero cross sections for groups 1 through 5 out of an 80-group structure. Storing only the 5-element band will save 75 words on the MATXS file. Even with the zeros removed, the number of words of vector cross section data for a submaterial can be quite large. Therefore, the MATXS format provides a way to break the vector data into a number of vector blocks. The idea is to sum up the bandwidths for each reaction (that is, `nlg(i)-nfg(i)+1`) in order to find the largest number of reactions that will fit within a block of length less than or equal to `maxw`. The data for this block are written to the output file, and then the next group of reactions is found. This continues until all the vector data have been written out as a series of "Vector Block" records with lengths less than `maxw`. This method minimizes the number of records on the file while allowing the application codes that read MATXS libraries to allocate space for reading the records economically.

Most submaterials will contain `n2b` "Scattering Matrix Control" records. The convention for the reaction names used in `hmtx` were discussed above. `lord` gives the *number* of Legendre orders present for this reaction; that is, `lord=4` for a P_3 matrix. Matrices are compacted for efficient storage and data transfer using two techniques. First, unnecessary leading and trailing zeros for group-to-group transfers into a particular final group are removed by banding. `jband(i)` gives the number of incident groups in the band for final group `i`, and `ijj(i)` gives the group index for the lowest-energy group (highest group index) in the band for group `i`. For example, consider an isotropic (n,2n) reaction with a threshold in group 3 of a 30-group structure. The values for `jband` and `ijj` for group 20 might well be 3 and 3, respectively. That is, groups 1 through 3 scatter into group 20. The order of storage for this matrix would be as follows:

Band	Element	`jband`	`ijj`
1	1→1	1	1
2	2→2	2	2
	1→2		
3	3→3	3	3
	2→3		
	1→3		
4	3→4	3	3
	2→4		
	1→4		
...	...		

Note that this scheme is more efficient than the similar one used for the CCCC ISOTXS file, because that method required that the ingroup element had to be included in the band. For anisotropic matrices, the `lord` Legendre components are stored with the source-group elements. Thus, there are `lord(i)*jband(i)` elements in each band, and they are stored in the following order (assuming a P_1 matrix):

Band	Element	Order	`jband`	`ijj`
1	1→1	0	1	1
	1→1	1		
2	2→2	0	2	2
	1→2	0		
	2→2	1		
	1→2	1		
3	3→3	0	3	3
	2→3	0		
	1→3	0		
	3→3	1		
...	...			

This kind of matrix block can be subdivided into subblocks using a method similar to the one described above for vector cross sections. The code starts summing the product of `jband` and `lord` for the final energy groups until the data for the next band will cause the sum to exceed `maxw` (normally 5000 words in MATXSR). These data are then written out as a matrix subblock. The code then repeats the process for the rest of the group ranges. The result is a minimal number of "Scattering Sub-Blocks," none of which has a size larger than the file limit.

The second method used to compact scattering matrices is based on the observation that the shape of the outgoing neutron or photon spectrum from fission and radiative capture reactions tends to be independent of energy at low neutron energies. GROUPR determines the group where significant energy dependence begins. Below this point, it computes a single spectrum to describe the outgoing neutron or photon distribution and a production cross section to go with it. At high energies, it produces a group-to-group matrix. This method can lead to appreciable reductions in storage requirements. For example, consider the 187-group structure, which has many low-energy groups. The fission spectrum for ^{235}U doesn't begin to show significant energy dependence until an energy of about 9 keV. This means that there are 118 constant groups. A 187×187 matrix is thereby reduced to a 187×69 matrix, and 187-group vector, and a 118-group vector—a 62 % reduction in storage requirements. Even larger reductions in size are obtained in more favorable cases. The parameter `jconst` gives the number of low-energy groups having a constant spectrum. If `jconst>0`, a single "Constant Sub-Block" record will be given after the regular "Scattering Sub-Block" records. This record will contain the spectrum `spec` and the production cross section `prod` needed to reconstruct the low-energy part of the full matrix. In mathematical form,

$$\sigma_{x,g\rightarrow g'} = \chi^{LE}_{g'}\,\sigma^{LE}_{Px,g} + \sigma^{HE}_{x,g\rightarrow g'}\ , \qquad (417)$$

where LE stands for low energy, HE stands for high energy, χ^{LE} is the constant spectrum, σ^{LE}_{Px} is the production cross section for reaction x, and σ^{HE} is the normal group-to-group matrix.

C. Historical Notes

The original version of the MATXS specification was constructed in September of 1977. The "Matrix Control" record in this original version contained the names and banding parameters for every group of every reaction. As a result, the record could become very large for libraries with many groups. The MATXS format was modified in September of 1980 to have a different "Matrix Control" record for each reaction. For several years following this date, NJOY contained both MATXSR and NMATXS modules, and two different versions of TRANSX were in use. All traces of the original version of the format have now disappeared.

In June 1988, changes were introduced to MATXSR to handle data types with either incoming or outgoing charged particles. Actually, the format wasn't changed. The particle, data type, and reaction name conventions described in Tables XI–XXI were chosen, and these new names required some corresponding changes in the code. The TRANSX code also had to be modified to recognize the new names and to work with multiparticle coupled sets.

The concept of using constant subblocks to reduce the size of GENDF and MATXS files is actually quite old. Various versions of updates to install the scheme were written over about a 5-year period. It was finally decided to permanently install the scheme in June of 1990. The changes required to GROUPR were the easier part of the job; corresponding changes were required in DTFR, CCCCR, MATXSR, POWR, and WIMSR. In addition, TRANSX had to be changed to accept the new constant subblocks. Unfortunately, this change was large enough to impact all MATXS users.

If many users were to be impacted, other nagging problems with the code could also be corrected at this time. With the original format, it was always very difficult to add, replace, or extract a material. The data type loop was outside the material loop, and the pieces for a given material were scattered throughout the file. Therefore, the ordering of the file was changed to put the data type loop inside the material loop along with the other submaterials. With this change, it was easy to rewrite the BBC code, which is the library maintenance code from the TRANSX package, to be able to insert a new material at any point in the library. It was also easy to give BBC the capability of extracting a short library containing only selected materials from a large MATXS library. Using the short library can significantly speed up TRANSX runs to prepare data for reactor design problems. The only disadvantage of the new ordering is that identical photoatomic data blocks have to be given for each isotope of an element. However, these data are not too bulky, and the additional overhead is manageable. This arrangement might make it easier to add photonuclear data in the future.

Finally, the current scheme of dividing the vector and matrix data into subblocks was added. The goal was to keep the record size below some fairly large maximum value (5000 words is currently being used). This kind of limit makes it easier to design application codes that make efficient use of memory. In addition, using as few records as possible reduces the number of I/O operations needed for a trip through the library, thereby improving execution time. The new subblocking scheme is also very simple.

D. MATXS Libraries

The normal process for preparing a MATXS library using NJOY starts with a series of runs to prepare PENDF files for each of the materials of interest. (For incident neutrons, these PENDF runs normally involve running the modules RECONR, BROADR, UNRESR, HEATR, and THERMR. For incident photons and charged particles, only RECONR is needed.) The next step is to run GROUPR for each material and incident particle. Each GROUPR run can produce data for all outgoing particles and for photons. As an example, consider the problem of producing data for a coupled neutron-photon-

proton library. GROUPR would be run using the ENDF-6 incident-neutron data and requesting MFD values of 3 (cross sections), 6 (neutron matrices, nn), 16 (photon production matrices, $n\gamma$), and 21 (proton production matrices, np). GAMINR would be run to produce the photoatomic cross sections and $\gamma\gamma$ matrices (no photonuclear data are currently available, so γn and γp matrices cannot be generated). GROUPR would be run again for the ENDF-6 proton sublibrary, requesting MFD values of 3 (cross sections), 6 (neutron production, pn), 16 (photon production, $p\gamma$), and 21 (proton scattering and production, pp).

Once all the GENDF fragments from all these GROUPR and GAMINR runs are available, they can be merged into multimaterial GENDF tapes using MODER. For the example above, three multimaterial GENDF tapes should be prepared: one for incident neutrons, one for incident photons, and one for incident protons. The input description in the following section shows how the unit numbers for these three GENDF files are given to MATXSR.

When the MATXSR run is made, the code scans through all the input GENDF tapes searching for the specified materials. For each material, it extracts all the data types requested in the MATXSR input and appearing on the input tapes. In addition, it extracts every temperature, σ_0 value, reaction, and Legendre order found on the input tapes. The following paragraphs describe various special features of the formatting process.

a. Normal Neutron and Photon Data. Normal, infinitely dilute cross sections and group-to-group matrices for neutron reactions, photon production, and photonuclear reactions are read from the input tapes and stored in the MATXS file using the formats described above. Almost all partial reactions are kept. Reaction names are constructed automatically; a number of examples were given above in Tables XIII–XXI. The total cross section and weighting flux from GROUPR are given in the section MF=3,MT=1 with both P_0 and P_1 components. All four vectors are written to the MATXS file. A similar treatment is used for MT=501 in the photoatomic case, except that only P_0 terms are saved.

The treatment of fission is also special. ENDF libraries sometimes use only MT=18, the total fission reaction, for both cross section and emission spectrum data; sometimes the partial fission cross sections (n,f), (n,n′f), (n,2nf), and (n,3nf) are also given in File 3 (MT=19, 20, 21, and 38); and sometimes the partial reactions are also used to describe the neutron emission matrix. If the last case is found, the MT=18 matrix may not be complete above the threshold for second-chance fission, and it should be ignored. Normally, it would not be processed in GROUPR, but just in case, MATXSR will ignore it. Delayed-neutron spectra processed in GROUPR are written into the section labeled MF=5,MT=455 on the GENDF tape, and this section includes data for all time groups. MATXSR adds up these separate time-group spectra to produce a single delayed-neutron spectrum for the MATXS vector data blocks. Application codes can use the fission data in the MATXS library to construct steady-state fission vectors as follows:

$$\bar{\nu}_g^{SS} = \frac{\sum_{g'} \sigma_{fg\to g'}^{HE} + \sigma_{Pfg}^{LE} + \bar{\nu}_g^D \sigma_{fg}}{\sigma_{fg}} , \qquad (418)$$

where σ^{HE} is the matrix part of the prompt fission reaction, σ_{Pfg}^{LE} is the low-energy production cross section for fission neutrons, $\bar{\nu}^D$ is the total delayed-neutron yield, and σ_{fg} is the fission cross section. If partial fission matrices are available, the first two terms in the denominator would also have to be summed over the reactions present. Continuing,

$$\chi_{g'}^{SS} = \frac{\sum_g \sigma_{fg\to g'}^{HE}\phi_g + \chi_{g'}^{LE}\sum_g \sigma_{Pfg}^{LE}\phi_g + \chi_{g'}^{D}\sum_g \bar{\nu}_g^D \sigma_{fg}\phi_g}{\text{NORM}} , \qquad (419)$$

where χ^{LE} is the constant-spectrum part of the fission reaction, χ^D is the total delayed-neutron spectrum, and NORM is the quantity that normalizes χ^{SS}, namely, the sum of the numerator over all g'. The weighting flux ϕ_g would normally be problem- and region-dependent in the application code.

b. Self-Shielding Data. If higher temperatures and σ_0 values are found on the input GENDF file, MATXSR automatically prepares submaterials for them. This self-shielding information can be used by application codes to prepare effective cross sections for mixtures of materials in various geometrical arrangements using equivalence theory. This approach is often called the Bondarenko method [41].

As discussed in more detail in the GROUPR section of this report, this system is based on using a model flux for isotope i of the form

$$\phi_\ell^i(E,T) = \frac{C(E)}{[\sigma_0^i + \sigma_t^i(E,T)]^{\ell+1}} , \qquad (420)$$

where $C(E)$ is a smooth weighting flux, $\sigma_t^i(E,T)$ is the total cross section for material i at temperature T, ℓ is the Legendre order, and σ_0^i is a parameter that can be used to account for the presence of other materials and the possibility of escape from the absorbing region (heterogeneity). GROUPR uses this model flux to calculate effective multigroup cross sections for the resonance-region reactions (total, elastic, fission, capture) for several values of σ_0 and several values of T.

When σ_0 is large with respect to the highest peaks in σ_t, the flux is essentially proportional to $C(E)$. This is called infinite dilution, and the corresponding cross sections are appropriate for an absorber in a dilute mixture or for a very thin sample of the absorber. As σ_0 decreases, the flux $\phi(E)$ develops dips where σ_t has peaks. These dips cancel out part of the effect of the corresponding

peaks in the resonance cross sections, thereby reducing, or self-shielding the reaction rate. In the MATXS format, these effects are represented by differences instead of the f-factors used in earlier formats (see the description of the BRKOXS file in the CCCCR section of this report). That is, the submaterial corresponding to temperature T and background cross section σ_0 contains the differences between the cross sections found on the GENDF tape for those parameters and the infinitely dilute values found in the first submaterial (normally, T=300 K and σ_0=1×10^{10}barns).

This approach has two advantages. First, data for groups with no self-shielding are automatically removed from MATXS vectors and matrices by the banding process without having to violate the principle of *generalization* and define a special format for neglecting values of "1.0" instead of values of "0.0." Second, effective cross sections can be accumulated by simply adding the self-shielding effects multiplied by appropriate interpolation weights to an accumulating sum that starts out equal to the infinitely dilute cross section. With f-factors, it is necessary to save the infinitely dilute cross sections during each step so that they can be multiplied by the f-factors. Thus, using differences is more economical in coding and in storage space requirements. The TRANSX code makes good use of this feature.

c. Thermal Data. ENDF-6 thermal data comes from a thermal sublibrary. In this sublibrary, the various material configurations are handled as separate materials. However, after the ENDF data have been processed by the THERMR module, the different thermal materials are handled as reactions. Table XX gives the correspondence between the special NJOY thermal MT numbers and the actual thermal materials.

Each of these thermal reactions corresponds to a particular dominant material. For example, MT=222 gives the thermal cross section for hydrogen bound in water. It will only appear in the material H1 in a MATXS library. Similarly, MT=223–227 will only appear in H1, MT=228 will only appear in H2, MT=229–230 will only appear in CNAT, MT=231–234 will only appear in BE9, and MT=235–236 will only appear in ZRNAT. Free-gas scattering (MT=221) will appear in all materials.

Many of these materials describe the scattering from one atom of a compound bound in that compound; for example, H in H_2O, D in D_2O, or Zr in ZrH. The application code using this data is expected to add on the effects of scattering from the other atoms of the compound. For water, the scattering from free oxygen is added to the "H in H_2O" scattering. For zirconium, the scattering for "H in ZrH" is added to the scattering from "Zr in ZrH". There are two exceptions in the existing ENDF/B evaluations. The benzene data set contains the entire scattering from the C_6H_6 molecule normalized to the hydrogen cross section. Therefore, if an application code specifies H1 with the benzene scattering option and the correct density for H1 in the system, the result will contain all of the benzene scattering effect. No additional scattering is to be added for the carbon atoms. Formerly, BeO contained all the scattering from the compound normalized to the beryllium cross section. More recently, BeO is been split up into Be in BeO and O in BeO. Check the THERMR section of this report for more information.

Four of these materials have names ending with $. These reactions represent coherent elastic scattering from crystalline powdered materials (C, Be, Be in BeO) or incoherent elastic scattering from solids containing hydrogen (polyethylene). These reactions contain ingroup elements only in the scattering matrices; that is, they cause angular redistribution without energy loss in scattering. Each $ reaction should be added to the corresponding inelastic reaction by the application code. This part of the scattering can lead to difficulties with transport corrections in discrete-ordinates transport codes.

The final unique aspects of the thermal data are that the matrices show upscatter, and they are only defined below some maximum energy. The energy range aspect is easily handled by the banding method used to reduce the size of the vector blocks. The upscatter aspect is handled by `jband` and `ijj`. The order of storage for a simple thermal case with only 2 upscatter groups would be as follows:

Band	Element	`jband`	`ijj`
...	...		
27	27→27	2	27
	26→27		
28	29→28	3	29
	28→28		
	27→28		
29	30→29	3	30
	29→29		
	28→29		
30	30→30	2	30
	29→30		

d. Charged-Particle Data. The treatment of data for incident charged particles is very similar to that used for neutron data, except for charged-particle elastic scattering. The elastic channel has contributions from Coulomb scattering that become infinite as the scattering angle goes to zero. In nature, this singularity is removed by electronic screening, but some other approach is needed for a data set that concentrates on isolated nuclear reactions. The approach selected is used in some existing applications. The elastic scattering distribution is broken up into two parts: (1) a normal angular distribution for angles from some low cutoff, say 20°, back to 180°, and (2) a straight-ahead continuous slowing-down contribution to represent the effects of angles below the cutoff. The continuous slowing-down part is closely related to the normal "stopping power" for charged particles. The large-angle part can be converted into a normal scattering matrix (see the discussion of charged-particle elastic scattering in the GROUPR section of this manual for more details).

Discrete-ordinates transport codes can often be modified to handle charged-particle data in this form, but they require an effective total cross section. The ENDF-6 format for charged-particle data does not define a total cross section because of the singularity in the elastic contribution. However, for application purposes, a reasonable definition of an effective total cross section is that it is the sum of all the partial reaction cross sections including the cross section obtained for the truncated elastic scattering reaction. This is the method that MATXSR uses to compute the quantities `ptot0`, `dtot0`, `ttot0`, etc.

e. Delayed-Neutron Data. Delayed-neutron yields, spectra, and decay constants by time group are required by reactor kinetics codes. The previous CCCC format for these data was DLAYXS (see the CCCCR section of this manual). Because of the generalization inherent in the MATXS format, these data can be added without any changes in the structure of the file.

f. Decay-Photon and Decay-Heat Data. Many nuclear reactions leave radioactive products. These products may be simple daughter isotopes that decay in a few steps to a stable final state with the emission of a few photons and electrons (or heat), or they may be a complex array of fission product isotopes that emit a complex spectrum of photons and electrons (or heat) showing many time constants. Procedures to store these data in MATXS libraries are not yet defined.

E. Running MATXSR

The user input specifications will be found in the comment cards at the beginning of the MATXSR module, in the NJOY manuals, and on the NJOY web page. It is always a good idea to check the comment cards in the current version of the code for possible changes.

The following example demonstrates the production of neutron data, photoatomic data, and thermal data for ^{238}U from ENDF/B-VII:

```
 1. matxsr
 2. 23 26 25/
 3. 13 't2lanl njoy'/
 4. 2 4 1 1/
 5 'U238 ENDF/B-VII NJOY10 187x24'/
 6. n g/
 7. 187 24/
 8. nscat ng gscat ntherm/
 9. 1 1 2 1/
10. 1 2 2 1/
11. u238 9237/
```

The line numbers are for reference and are not part of the input. The file `tape23` contains the neutron and photon production data from a GROUPR run. The file `tape26` contains the photoatomic data from a GAMINR run. Finally, `tape25` contains the MATXS-format output. Line 3 sets the CCCC version number and the text ID string for the file. Line 4 specifies that 2 particle types are defined—they are indicated as neutrons and photons in line 6. The number of groups for the two particle types are given in line 7. There will be 4 data types as defined in Line 8. Lines 9 and 10 define the data types in terms of the two particles. The first type uses particle 1 for incident and outgoing particles. The second uses particle 1 for the incident channel and particle 2 for the outgoing channel. And so on. Finally, line 11 gives the name for the material and its MAT number.

The next few paragraphs give additional details on the various input parameters.

Card 1 is used to specify the units for the input GENDF tapes and the output MATXS library. As usual, the sign of a GENDF unit number is used to determine its mode; negative numbers mean binary, and positive numbers mean coded (that is, ASCII, EBCDIC, etc.). The output file is always binary, and its sign is ignored. The most common MATXSR runs are for neutron and photon data only. In these cases, card 1 can be truncated after the `nmatx`. If incident charged-particle GENDF files are available, any of the units `ngen3` through `ngen7` can be assigned.

Card 2 is used to control the MATXSR print option `iprint` and to provide the information for the MATXS "User Identification" record. An example for Card 2 follows:

```
 0  17  'T2 LANL NJOY'/  .
```

Card 3 is used to input the counts for the "File Control" record and to tell the user input routine how many quantities to read in subsequent input cards. Card 4 is repeated `nholl` times to give the Hollerith description for the library. Each line must be delimited by quote characters and terminated by /. For example,

```
' MATXS17                                27 FEB 91 '/
' 69-GROUP THERMAL LIBRARY FROM ENDF/B-VI          '/
'   THIS LIBRARY INCLUDES NEUTRON, PHOTON, AND     '/
'   THERMAL SCATTERING DATA FOR 133 MATERIALS.     '/
```

The lines are written out on the MATXS file as given, except that the maximum length for each line is 72 characters.

Card 5 is used to read in the Hollerith names for the `npart` particles for this library. The standard names for the particles were given above in Table XI. The number of groups desired for each particle are given on card 6. The data-type names (see Table XII) are given on Card 7. Cards 8 and 9 are used to specify the input and output particles for each data type. The user should take care that the number of groups given for each particle is consistent with the input data, and that the particle assignments to data type names (`jinp` and `joutp`) are consistent with the names. The code does not check.

Card 10 is repeated `nmat` times to specify the materials to be written out on the library. The rules for constructing the material names `hmat` were given in Sect. XIX B. The `matno` parameter is the ENDF MAT number for this material as used on the input GENDF files. For the

ENDF-6 format, the MAT number is the same for all sublibraries (that is, for incident neutrons, photons, protons, etc.), and only one value is needed to specify the desired material. However, photoatomic data are atomic in character, and the MAT numbers always refer to the element. For example, MAT=2600 for the photoatomic data of iron. MATXSR reads a second MAT number field, `matgg`, for the photoatomic data. It's default value is given by

```
100*(MATNO/100)  ,
```

and it is, therefore, not usually needed. A special branch is used for Fm-255, whose elemental MAT number is 9920. The normal form for card 10 would be as follows:

```
FE56  2631/  .
```

However, the photoatomic libraries for earlier versions of the ENDF/B files used MAT numbers like 26 for elements. The `matgg` parameter can be used to process data from these older libraries. Note that delimiting quote characters are not required for Hollerith names that are single words and start with a letter.

MATXSR always processes all submaterials found on all the nonzero GENDF units `ngen1` through `ngen7`. It processes every temperature and σ_0 value found. It processes (almost) every reaction cross section and matrix found, and it processes every Legendre order given. The only way to control the contents of the MATXS library is through the input to GROUPR or GAMINR and by the materials included when building the input GENDF files. A convenient way to handle this task is to assemble the results of a number of single-material GROUPR runs into composite GENDF tapes using MODER.

XX. WIMSR

This module is used to prepare libraries for the reactor-physics code WIMS [87]. WIMS stands for "Winfrith improved multigroup scheme;" it has been developed through its various versions at the UK laboratory AERE/Winfrith. WIMS-9A is the current Winfrith version, and it is distributed commercially. WIMS-D is an older version that is freely available through various distribution centers; therefore, it is very popular all around the world. WIMS-E was the commercialized version that succeeded WIMS-D, and its data files are also supported by WIMSR.

WIMS uses collision-probability methods for computing fluxes in reactor pin cells and more complicated geometrical arrangements. Therefore, it requires transport, fission, and capture cross sections, a transfer matrix for epithermal neutrons, fission-source information ($\overline{\nu}$ and χ), and a bound-atom scattering matrix for thermal neutrons. Self-shielded cross sections are obtained using equivalence theory from tabulated resonance integrals with intermediate resonance corrections. The resonance integrals can be obtained from the self-shielded cross sections produced by GROUPR, and the intermediate-resonance λ values by group can be computed using the NJOY flux calculator. WIMS libraries normally use a standard 69-group structure with 14 fast groups, 13 resonance groups, and 42 thermal groups.

A. Resonance Integrals

WIMS computes the self-shielded cross sections for a wide range of mixtures and fuel geometries using equivalence theory. The GROUPR section of this report describes the narrow-resonance (NR) version of equivalence theory; that is, all systems with the same value for the "sigma-zero,"

$$\sigma_{0i} = \frac{1}{n_i}\left\{\sum_{j\neq i} n_j\sigma_{tj} + \sigma_e\right\} , \tag{421}$$

in a group have the same self-shielded cross section in that group. Here, n_j is the number density for material j with cross section σ_{tj}, and σ_e is the escape cross section (which takes care of the geometry of the fuel).

However, in the near epithermal range (*e.g.,* 4–100 eV), some resonances are too wide for the NR approximation to apply well. For these resonances, the effect of a moderator material is reduced, because collisions with the moderator do not always result in enough energy loss to remove the neutron from the resonance. For this reason, WIMS uses an intermediate-resonance (IR) extension to equivalence theory in which the background cross section is taken to be the following:

$$\sigma_{Pi} = \frac{1}{n_i}\left\{\sum_{j} n_j\lambda_j\sigma_{pj} + \sigma_e\right\} , \tag{422}$$

where the λ factors are numbers between zero and one. Note that σ_p, the potential scattering cross section, is used here, and that the sum is now over all materials. The basic concept is the same: all systems with the same value of the IR "sigma-P" for a group will have the same self-shielded cross sections for that group.

WIMS takes the additional step of expressing the self-shielding data in terms of "resonance integrals," instead of using the self-shielded cross sections produced by GROUPR. That is,

$$\sigma_x(\sigma_0) = \frac{\sigma_P I_x(\sigma_P)}{\sigma_P - I_a(\sigma_P)} , \tag{423}$$

and

$$I_x(\sigma_P) = \frac{\sigma_P\sigma_x(\sigma_0)}{\sigma_P + \sigma_a(\sigma_0)} , \tag{424}$$

where x denotes the reaction, either "a" for absorption or "nf" for nu*fission, and I_x is the corresponding resonance integral.

In order to clarify the meaning of this pair of equations, consider a homogeneous mixture of ^{238}U and hydrogen with concentrations such that there are 50 barns of hydrogen scattering per atom of uranium. The GROUPR flux calculator can be used to solve for the flux in this mixture, and GROUPR can then calculate the corresponding absorption cross section for ^{238}U. Assuming that $\lambda = 0.1$ and $\sigma_p = 10$ for the uranium, the numbers being appropriate for WIMS group 25, we get $\sigma_P = 51$. This value of σ_p goes into the `sigz` array in the resonance-integral block on the WIMS library, and the corresponding I_a goes into the `resa` array.

At some later time, a WIMS user runs a problem for a homogeneous mixture of ^{238}U and hydrogen that matches these specifications. WIMS will compute a value of σ_P of 51.0, interpolate in the table of resonance integrals, and compute a new absorption cross section that is exactly equal to the accurate computed result from the original GROUPR flux-calculator run.

This argument can be extended to more complex systems. For example, the assembly calculated using the flux calculator could represent an enriched uranium-oxide fuel pin of a size typical of a user's reactor system with a water moderator. The computed absorption cross section is converted to a resonance integral and stored with the computed value of σ_P. In any later calculation that happens to mimic the same composition and geometry, WIMS will return the accurate calculated absorption cross section. Equivalence theory, with all its approximations, is only used to interpolate and extrapolate around these calculated values. This is a powerful approach, because it allows a user to optimize the library in order to obtain very accurate results for a limited range of systems without having to modify the methods used in the lattice-physics code. Unfortunately, the present version of WIMSR does not allow you to enter σ_P directly; it computes it from input data that only consider one material at a time. A future version may include the more general capabilities described in this paragraph.

Let us call the homogeneous uranium-hydrogen case discussed above "case 0." Now, consider a homogeneous mixture of ^{238}U, oxygen, and hydrogen. Ratio the number densities to the uranium density such that there is 1 barn/atom of oxygen scattering and 50 barn/atom of hydrogen scattering. Carry out an accurate flux calculation for the mixture, and call the result "case 1." Also do an accurate flux calculation with only hydrogen, but at a density corresponding to 51 barns/atom. Call this result "case 2." The IR lambda value for oxygen is then given by

$$\lambda = \frac{\sigma_a(1) - \sigma_a(0)}{\sigma_a(2) - \sigma_a(0)} \ . \tag{425}$$

Note that λ will be 1 if the oxygen and hydrogen have exactly the same effect on the absorption cross section. In practice, $\lambda = .91$ for WIMS group 27 (which contains the large 6.7 eV resonance of ^{238}U), and $\lambda = 1$ for all the other resonance groups. That is, all the resonances above the 6.7 eV resonance are effectively narrow with respect to oxygen scattering.

Similarly, do a flux-calculator solution for a homogeneous mixture of ^{238}U combined with 1 barn/atom of ^{235}U and 50 barn/atom of hydrogen. Call the result "case 3." Now, the lambda value for ^{235}U is given by

$$\lambda = \frac{\sigma_a(3) - \sigma_a(0)}{\sigma_a(2) - \sigma_a(0)} \ . \tag{426}$$

The actual value obtained for WIMS group 27 is .035. Group 26 gives 0.50, and group 25 gives 0.09. An examination of the flux-calculator equations in the GROUPR section of this manual shows that the effect of the "admixed" moderator term depends only on its atomic mass (through the α value); therefore, the IR λ values will be the same for all uranium isotopes (and the same values should work for all the actinides). This conclusion neglects the small effects of absorption in the admixed isotope on the intraresonance flux for one resonance.

This process can be continued for additional admixed materials from each important range of atomic mass. The result is the table of λ_{gi} values needed as WIMSR input.

What are the implications of this discussion? Foremost is the observation that the lambda values for the isotopes are a function of the composition of the mixture that was used for the base calculation. To make the effect of this clear, let us consider two different types of cells:

1. a homogeneous mixture of U238 and hydrogen, and
2. a homogeneous mixture of U235 and hydrogen.

A look at the pointwise cross sections in group 27 shows very different pictures for the two uranium isotopes. The U-238 cross section has one large, fairly wide resonance at 6.7 eV, and the U-235 cross section has several narrower resonances scattered across the group. If the lambda values are computed for these two different situations, the results in Table XXII are obtained.

It is clear that the energy dependence of the two lambda sets is quite different. This is because of the difference in the resonance structure between U-238 and U-235. Clearly, the one resonance in group 27 in U-238 is effectively wider than the group of resonances in group 27 for U-235. Group 26 has essentially no resonance character for ^{238}U, which reverses the sense of the difference. In groups 24 and 25, the ^{235}U resonances become more narrow, while the ^{238}U resonances stay fairly wide. Finally, in group 23, the ^{238}U resonances begin to get narrower.

These results imply that completely different sets of lambda values should be used for different fuel/moderator systems, such as U-238/water, U-235/water, or U-238/graphite. In practice, this is rarely done.

The remaining question is, "How should the self-shielded cross sections for the minor isotopes be calculated?" Formally, the best approach using NJOY would be to first do an accurate flux calculation for pure ^{238}U

TABLE XXII: IR λ values for several resonance groups and two different reactor systems.

WIMS group	λ(U) U238@50b	λ(O) U238@50b	λ(U) U235@200b	λ(O) U235@200b
27	.035	.91	0.20	1.00
26	.50	1.00	.38	1.00
25	.092	1.00	.44	1.00
24	.090	1.00	.55	1.00
23	.29	1.00	.46	1.00

mixed with hydrogen (or to be really accurate, a typical reactor cell containing pure ^{238}U oxide), and to save the resulting flux on a scratch file. This flux would then be used as input for the ^{235}U calculation. (See the GROUPR section for details.) This approach takes care of all the complexities of resonance-resonance interference, the drop in the average flux across the group caused by accumulated ^{238}U absorptions, and so on. In practice, this is rarely done. Since the self-shielding effects in the minor actinides are much smaller than those in the main absorber, it is usually sufficient to do a simple NR calculation for the minor actinides and to convert them into WIMS resonance integrals with the normal lambda values for heavy isotopes.

B. Cross Sections

The first part of the WIMS cross section data contains σ_p for the resonance groups (15–27 in the normal 69-group structure), the scattering power per unit lethargy for the resonance groups, the transport cross section for the fast groups (1–27 normally), the absorption cross section for the fast groups, an obsolete quantity for the resonance groups (set to zero), and the intermediate-resonance λ values for the resonance groups. For the WIMS-E version of the format, the (n,2n) cross section is added between the slowing-down power and the transport cross section.

The σ_p value is assumed to be constant (see `sigp` in the user input instructions). It must be obtained by finding the scattering length a in the ENDF file and computing $\sigma_p = 4\pi a^2$. The scattering power per unit lethargy is $\xi\sigma_s/\tau$, where ξ is the log energy loss parameter given as MT=252 on the GENDF file, σ_s is the elastic scattering cross section (MT=2), and τ is the lethargy width for the group, which can be calculated from the group structure given in MF=1/MT=451 on the GENDF file. The (n,2n) cross section is obtained from MF=3/MT=16 on the GENDF file. The absorption cross section is computed by adding up the fission cross section (MT=18) and all the cross sections given with MT=102–150. The (n,2n) cross section (MT=16) is then subtracted from the sum. Finally, the λ values are obtained from the user's input. See Sect. XX A for more details on these intermediate-resonance corrections.

The next part of the WIMS data file contains the fission neutron production cross section $\overline{\nu}\sigma_f$ and the fission cross section σ_f for the fast groups (1–27 normally). The cross section is always obtained from MT=18 on the GENDF file, but there are several complications involved in getting $\overline{\nu}$.

A shortcut for obtaining the fission data is to run MFD=3/MTD=452 and MFD=5/MTD=452 in GROUPR. This approach ignores the energy dependence of fission neutron emission at high energies and the effects of delayed neutrons on the fission spectrum. If these options are used in GROUPR, it is important not to use the other options described below at the same time. When WIMSR finds a section on the GENDF file with MF=3/MT=452, it can read in $\overline{\nu}\sigma_f$ directly.

A better approach to fission in GROUPR is to prepare a full fission matrix for MT18, or to prepare matrices for all the partial fission reactions, MT=19, 20, 21, and 38. The latter is the recommended approach for evaluations with both MT=18 and MT=19 given in File 5. See the GROUPR section (Sect. IX) for more details. WIMSR reads in the data given in MF=6/MT=18, or in MF=6/MT=19,20,..., and sums over all secondary-energy groups to obtain the prompt part of $\overline{\nu}\sigma_f$. It adds in the delayed part of $\overline{\nu}\sigma_f$ from MF=3/MT=455. If the input GENDF file contains both MT=18 and partial fission matrices, a diagnostic message will be printed, and the partial-fission representation will be used.

The next section of the WIMS data file contains the nonthermal P_0 scattering matrix for incident-energy groups in the fast range (1–27 normally). This matrix is loaded by summing over all of the reactions found on the GENDF tape except the thermal reactions `mti` and `mtc`. If requested, this matrix is transport corrected by subtracting the sum over secondary-energy groups of the P_1 matrix for each primary group. When the individual reactions are read, they are loaded into "full" matrix (typically 69×69). At the same time, a record is kept of the lowest and highest secondary groups found for each primary group. These limits are then used to pack the scattering matrix into a more compact form.

The scattering matrix is not actively self-shielded in WIMS, but WIMSR allows the user to request that the elastic component be evaluated at some reference σ_0 value different from infinity. This option can be useful for the major fertile component of reactor fuel, that is,

for ^{238}U in pins of a uranium system, or for ^{232}Th for fuel in a Thorium/U233 system.

Because the thermal scattering matrix depends on temperature, the next component of the WIMSR data contains the `ntemp` versions of the basic cross sections and the P_0 scattering matrix for the thermal groups (28–69 normally). The cross sections included are transport, absorption, nu*fission, and fission. The transport positions contain the sum of the thermal inelastic cross section obtained by summing up the P_0 matrix (MF=6/MT=MTI), the thermal elastic cross section from the diagonal elements of the P_0 matrix (MF=6/MT=MTC), if present, and the absorption cross section. If separate P_1 matrices are not given for this material, the P_1 cross section obtained by summing the P_1 matrices over secondary groups for each primary group is subtracted. The matrix data are read from sections on the GENDF file with MF=6/MT=MTI and MF=6/MT=MTC (if present). As for the temperature-independent matrices, they can be transport-corrected by subtracting the sum over secondary groups of the P_1 matrix for each primary group from the self-scatter position. Also, minimum and maximum limits on the secondary group are determined for each primary group, and the matrix is compacted for efficiency.

The next part of the WIMS data file contains the resonance data, which were discussed in subsection A.

In some cases, these resonance data are followed by a fission spectrum block. The complications of obtaining the fission spectrum are the same as those described above for obtaining the fission neutron production cross section, $\overline{\nu}\sigma_f$. If the short-cut option was used in GROUPR, the fission spectrum can be read directly from MF=5/MT=452 on the GENDF file. The sum over groups is also accumulated in `cnorm` in order to allow the final spectrum to be normalized accurately. The shortcut approach neglects the effects of delayed neutron emission on the fission spectrum.

If fission matrices are available (either MT=18, or MT=19+20+21...), the prompt part of the fission spectrum is obtained by summing $\sigma_{gg'}\phi_g$ over all primary groups, g. These numbers are also summed into `cnorm` for use later in normalizing χ. The delayed part is obtained as

$$\left\{\sum_g \nu_d \sigma_{fg} \phi_g\right\} \chi_{dg'} \ , \tag{427}$$

which also contributes to the normalization. Note that the energy dependence of the fission matrix is factored into the final χ in proportion to the weighting flux used in GROUPR to prepare the WIMSR input file. For thermal reactor problems, it is easy to provide a good estimate for this weighting flux.

The final block on the WIMS data tape is optional. If present, it contains P_1 scattering matrices for each temperature. These matrices are defined over the entire group range (normally 1–69), and they contain both the temperature-independent and temperature-dependent reactions in each matrix. The methods used to build up these matrices are parallel to those discussed above for the P_0 matrices. Note that if P_1 matrices are given, the transport corrections are <u>not</u> included in the transport cross sections or P_0 matrices.

C. Burn Data

WIMS uses a simplified burn model for tracking the production and depletion of actinides and fission products, and the chains used are hard-wired into the code. WIMSR provides a method to enter new fission-yield data into the WIMS library format, but it has not been used or tested very much so far.

D. Running WIMSR

A usual, The input specifications will be found in the comment cards at the beginning of the WIMSR source, in the NJOY manuals, or on the NJOY web site. It is always a good idea to check the comment cards in the current version to see if there have been any changes.

The first card specifies the input and output unit numbers, as is normal for NJOY modules. `ngendf` comes from a previous GROUPR run, and it can be in either binary or ASCII mode. `nout` is always in ASCII mode.

The options card allows the user to select how much detail will be printed on the output listing (`iprint`), whether the output is intended for WIMS-D or WIMS-E (`iverw`), and how many groups are desired. At the time this module was developed, the WIMS-D and WIMS-E formats were supported. The only difference between WIMS-D and WIMS-E output was that some additional reaction cross sections are included for the latter. If the user selects some group structure different from the standard 69-group structure, an additional input card is required to give the number of groups (`ngnd`), the number of fast groups (`nfg`, 14 for 69 groups), the number of resonance groups (`nrg`, 13 for 69 groups), and the reference group used for normalizing the flux (`igref`, normally the low-energy group of the fast groups).

Card three is always required. It gives the ENDF MAT number for the material to be processed. If this MAT doesn't appear on the GENDF tape, a fatal error message will be issued. `nfid` will be the identification number for this material used on the output WIMS library, and `rdfid` will be the identification number for the resonance data. Formally, WIMS libraries allow for data sets with more than one version of the resonance-integral tabulation. The last parameter on this card is `iburn` to flag whether burn data are included in the input stream.

The next card starts out with `ntemp` and `nsigz`, which define the size of the resonance-integral tables. They are normally both set to zero; the code then uses all of

TABLE XXIII: Conventional Values for the Thermal MT Numbers (`mti` and `mte`) Used in WIMSR, GROUPR, and THERMR.

Thermal Material	`MTI` Value	`MTC` Value
H in H_2O	222	
D in D_2O	228	
Be metal	231	232
Graphite	229	230
Benzine	227	
Zr in ZrH	235	236
H in ZrH	225	226
Be(BeO)	233	234
O(BeO)	237	238
O(UO2)	239	240
H in Polyethylene	223	224
Al metal	243	244
Fe metal	245	246

the values computed by GROUPR. The reference sigma-zero value, `sgref`, is used for the elastic cross section and matrix, because these quantities are not normally self-shielded by WIMS. Normally, `1e10` is appropriate, but for the major fissionable material in the reactor (*i.e.,* U-238 or Th-232), it may be better to use a realistic number like `sgref=50`. WIMSR assumes that the potential scattering cross section for the material is constant, but this constant value is not available from the GROUPR output. The WIMSR user will have to look in the ENDF-format evaluation for the scattering length a, compute $\sigma_p = 4\pi a^2$, and enter the value as `sigp`. The fragment in Fig. 90 shows where to find the scattering length (`9.56630- 1` in this case).

The parameters `mti` and `mtc` select the thermal inelastic and elastic data from the sections that might be available on the GENDF tape. Most materials have only free-gas scattering available, and the appropriate values would be `mti=221` and `mtc=0`. The conventional values to use for reactor moderator materials are given in Table XXIII.

WIMSR allows P_1 scattering to be treated in two ways. If `ip1opt=1`, the P_1 matrix for the material is written to the WIMS output file explicitly. This option is normally used only for major moderator materials, such as the components of water. The other option, `ip1opt=1`, instructs the code to use the P_1 data to transport-correct the P_0 elastic scattering matrix; that is, the ingroup elements of the P_0 matrix are reduced by the sum over all outgoing groups of the P_1 matrix for that ingoing group.

The `inorf` parameter can be set to 1 to eliminate the resonance-integral table for nu*fission from the WIMS output. Some of the higher actinides are treated this way for some WIMS libraries. The `isof` flag is set to 1 to tell WIMSR to produce a fission spectrum. This is usually done for main fissile materials, such as ^{235}U and ^{239}Pu. The `ifprod` flag is used to control whether resonance tables are included for fission products.

WIMSR has some capability to format burn data for incorporation into a WIMS library (see cards 5 and 6). This part of the code has not been used or tested very much.

The final card gives the intermediate-resonance λ values for each of the resonance groups. Methods for obtaining these quantities are outlined in subsection A.

E. WIMS Data File Format

The following section describes the WIMS data output provided by WIMSR. It consists of a number of logical blocks of information written out in coded form. The output is intended to be used by a library maintenance code to prepare a binary library for use by the WIMS code.

Library Header `(2I5)`

- `NFID` material identifier
- `NPOS` position to insert material on a large multimaterial library (given for WIMS-E only)

Burnup Data `(3(1PE15.8,I6))`

- `(YIELD(I),IFISP(I),I=1,JCC/2` fission yields and fission product flag

Material Identifier Data `(I6,1PE15.8,5I6)`

- `IDENT` material identifier
- `AWR` atomic weight ratio to neutron
- `IZNUM` atomic charge number Z for this material
- `IFIS` fission and resonance flag: 0=non-fissile with no resonance tabulation, 1=non-fissile with resonance absorption only, 2=fissile with resonance absorption only (*e.g.,* Pu-240), 3=fissile with resonance absorption and fission, 4=fissile with no resonance tabulations.
- `NTEMP` number of temperatures
- `NRESTB` number of resonance tables included (0 or 1)
- `ISOF` fission spectrum flag (0=no, 1=yes)

Temperature-Independent Vectors `(1P5E15.8)`

- `(SIGP(I),I=1,N2)` σ_p for resonance groups
- `(XX(I),I=1,N2)` $\xi\sigma_s/\tau$ for resonance groups
- `(XTR(I),I=1,N1+N2)` transport cross section for fast and resonance groups
- `(ABS(I),I=N1+N2)` absorption cross section for fast and resonance groups

```
...
9.22350+ 4 2.33025+ 2          0          0          1          01395 2151
9.22350+ 4 1.00000+ 0          0          1          2          01395 2151
1.00000+ 0 8.20000+ 1          1          1          0          01395 2151
3.50000+ 0 9.56630- 1          0          0          1          01395 2151
2.33025+ 2 0.00000+ 0          0          0        780        1301395 2151
...
```

FIG. 90: How to find the scattering length. These are the first few lines of File 2, and the scattering length appears in the second field on the fourth line as 0.95663.

`(DUM,I=1,N2)` unused dummy for resonance groups

`(ALAM(I),I=1,N2)` IR λ values for resonance groups

Temperature-Indep Fission (`IFIS>1` only) `(1P5E15.8)`

`(NSIGF(I),I=1,N1)` $\overline{\nu}\sigma_f$ for fast and resonance groups

`(SIGF(I),I=1,N1)` σ_f for fast and resonance groups

P_0 Matrix Length `(I15)`

`NDAT` length of P_0 scattering block to follow

Temperature-Independent P_0 Matrix `(1PE15.8)`

`(XS(I),I=1,NDAT)` packed scattering data: `IS` for group 1, `NS` for group 1, `NS` scattering elements for group 1, `IS` for group 2, `NS` for group 2, *etc.*, through all of the fast and resonance groups (normally through group 27). `IS` is the position of self-scatter in the band of scattering elements (always 1 here), and `xxx` is the number of elements in the band.

Temperature Values `(1PE15.8)`

`(TEMP(I),I=1,NTEMP)` temperatures in Kelvin

Repeat the following 4 blocks for each of the `NTEMP` temperatures.

Temperature-Dep Transport and Absorption `(1PE15.8)`

`(XTR(I),I=1,N3)` transport cross section for thermal groups

`(ABS(I),I=1,N3)` absorption cross section for thermal groups

Temperature-Dependent Fission Vectors `(1PE15.8)`

`(NSIGF(I),I=1,N3)` fission neutron production cross section $\overline{\nu}\sigma_f$ for thermal groups

`(SIGF(I),I=1,N3)` fission cross section σ_f for thermal groups

Temperature-Dependent P_0 Matrix Length `(I15)`

`NDAT` length of P_0 scattering block

Temperature-Dependent P_0 Matrix `(1PE15.8)`

`XS(I),I=1,NDAT)` packed scattering data: `IS` for group N, `NS` for group N, `NS` scattering elements for group N, `IS` for group N+1, `NS` for group N+1, *etc.*, through the last group (normally group 69). `IS` is the position of self-scatter in the band of scattering elements, and `NS` is the number of elements in the band.

Record Mark `(' 999999999999')`, WIMS-D only

Resonance Control Data

`RID` resonance set identifier

`NTEMP` number of temperatures for this resonance set

`NSIGZ` number of background cross section values

Absorption Resonance Data

`(TEMP(I),I=1,NTEMP)` temperatures

`(SIGZ(I),I=1,NSIGZ)` background cross section values

`(RESA(I),I=1,NTEMP*NSIGZ` absorption resonance integrals

Nu*fission Resonance Data

`(TEMP(I),I=1,NTEMP)` temperatures

`(SIGZ(I),I=1,NSIGZ)` background cross section values

`(RESNF(I),I=1,NTEMP*NSIGZ` nu*fission resonance integrals

Scattering Resonance Data, WIMS-E only

`(TEMP(I),I=1,NTEMP)` temperatures

`(SIGZ(I),I=1,NSIGZ)` background cross section values

`(RESS(I),I=1,NTEMP*NSIGZ` scattering resonance integrals

Record Mark `(' 999999999999')`, WIMS-D only

Fission Spectrum

`(FSPECT(I),I=1,NGND)` fission spectrum χ

Repeat the following two blocks for each of `NTEMP` *temperatures.*

P_1 Matrix Length `(I15)`

`NDAT` length of P_1 scattering block

P_1 Matrix `(1PE15.8)`

`XS(I),I=1,NDAT)` packed scattering data: `IS` for group 1, `NS` for group 1, `NS` scattering elements for group 1, `IS` for group 2, `NS` for group 2, *etc.*, through the last group (normally group 69). `IS` is the position of self-scatter in the band of scattering elements, and `NS` is the number of elements in the band.

F. WIMSR Auxiliary Codes

The WIMSR output as described above is not directly usable by WIMS. Two private library-maintenance codes have been used at Los Alamos. FIXER is used to modify (fix up) an existing WIMS-D library, or to create a new one, using WIMSR output. It processes burnup data, main data, resonance data, and P_1 matrices. In its fix-up mode, it can replace, delete, or add a material. WRITER is a code to read a WIMS-D library in coded form, convert it to binary form, and list it in a user readable form. Similar codes have been used by other laboratories that use WIMS.

XXI. HISTORY AND ACKNOWLEDGMENTS

NJOY was started as a successor to MINX [78] late in 1973 (it was called MINX-II originally). The current name was chosen in late 1974 to be evocative of "MINX plus." The first goals were to add a photon production capability, to add a photon interaction capability, to provide an easy link to the Los Alamos 30-group libraries of the day using DTF [75] format, and to merge in the existing capabilities to produce libraries for the MCNP Monte Carlo code. Most of the work was done by MacFarlane; Rosemary Boicourt joined the project in 1975. First, the RESEND and SIGMA1 modules of MINX were converted to use union grids, and a new method of resonance reconstruction was developed. These steps led to RECONR and BROADR. UNRESR, which was based on methods from ETOX, was moved over from MINX with only a few changes. Next, a completely new multigroup averaging module, GROUPR, was developed around the unifying concept of the "feed function," and it handled neutron- and photon-production cross sections in a unified manner. The CM Gaussian integration for discrete two-body scattering was developed. DTFR was installed as the first NJOY output module. The first versions of the NJOY utility codes were introduced; the new concepts of "structured programming" inspired some of the features of the new NJOY code.

Major influences during this period included Don Harris, Raphe LaBauve, Bob Seamon, and Pat Soran at Los Alamos, and Chuck Weisbin at the Oak Ridge National Laboratory (ORNL). Odelli Ozer at Brookhaven (and later EPRI) helped with RESEND, and Red Cullen at Livermore and John Hancock at Los Alamos helped with the Doppler broadening module. In those days, the development of NJOY was supported by the US Fast Breeder Reactor and Weapons programs.

Code development continued during 1975. The ERRORR module was added for calculating covariances from ENDF/B files. The ACER module was created, borrowing heavily from LaBauve's ETOPL and Chuck Forrest's MCPOINT code. Rich Barrett joined the project, and he did much of the work in creating a new CCCC module for NJOY that had several advances over the MINX version and that met the CCCC-III standards. By the end of the year, HEATR had also been added to the code (with ideas from Doug Muir).

During 1976, free-form input and dynamic data storage were added to NJOY. GAMINR was written to complete the original NJOY goal of processing photon interaction cross sections, and the MATXSR module was designed and written, primarily by Rich Barrett. This completed the capability to construct fully coupled cross sections for neutron-photon-heating problems. A major new effort was writing the THERMR module to improve upon the thermal moderator scattering cross section then produced using FLANGE-II and HEXSCAT, and starting the POWR module for the EPRI-CELL and EPRI-CPM code being developed by the US electric utility companies. This work was funded by the Electric Power Research Institute (EPRI) and coordinated by Odelli Ozer.

The first release of NJOY to what was then called the Radiation Shielding Information Center (RSIC) at Oak Ridge and to the National Energy Software Center (NESC) at Argonne was made in the summer of 1977. This version was tested and converted for IBM machines by R. Q. Wright (ORNL). Also, TRANSX was developed during 1977, the MATXS1 30x12 library was produced based on ENDF/B-IV, the flux calculator was added to NJOY to support the EPRI library work, and the first version of the EPRI-CELL library was created and used.

A second release of NJOY was made in 1978. In addition, further improvements were made for preparing EPRI cross sections, the MATXS/TRANSX system was improved, and a thermal capability was added to

the MCNP Monte Carlo code using cross sections from THERMR as processed by ACER.

In 1979, the radiation damage calculation was added to HEATR, and the GROUPR flux calculator was further improved. In 1980, a plotting option was added to ERRORR. During this period, NJOY had become more stable. Changes consisted of small improvements or bug fixes instead of major new capabilities. Starting in this period, NJOY received some support from the US Magnetic Fusion Energy Program, mostly for covariance work and TRANSX-related library support.

In 1981 and 1982, improvements included the momentum-conservation method for radiative capture in HEATR. Analytic $\psi\chi$ broadening was added to RECONR for some cases, and the integral criteria for resonance reconstruction with significant-figure control were installed in RECONR. Several new capabilities were added to ERRORR, and the COVR module was added to NJOY to handle both ERRORR plotting and library output (this work was done by Doug Muir). In addition, CCCCR was updated to the CCCC-IV standards. A new major release called NJOY (10/81) was made to the code centers, and the first two volumes of a new NJOY report were written and published. European users began to make important contributions about this time. Enrico Sartori of the NEA Data Bank (then at Saclay in France), Margarete Mattes of the University of Stuttgart in Germany, and Sandro Pelloni of the Paul Scherer Institute in Switzerland deserve mention.

Another major release, NJOY 6/83, was made in 1983. By this time, NJOY was in use in at least 20 laboratories in the US and around the world. Small improvements continued, such as the kinematic KERMA calculation in HEATR. The temperature dependence of the BROADR energy grid was introduced early in 1984 based on an observation by Ganesan (India). Volume IV of the NJOY report was published in 1985, and Volume III appeared in 1987.

The next big set of improvements to NJOY was associated with the introduction of the ENDF-6 format. This required significant changes in RECONR to support new resonance formats like Reich-Moore and the Hybrid-R-Function (implemented with help from Charlie Dunford of Brookhaven), in HEATR to implement direct calculations of KERMA and damage from charged-particle and recoil distributions in File 6, in THERMR to support new formats for File 7, and in GROUPR to support the group-to-group transfer matrices using energy-angle data from File 6. The PLOTR module was also developed during the period. The result was the release of NJOY89 in time for processing the new ENDF/B-VI library and the European JEF-2 library (which was also in ENDF-6 format).

During 1989 and 1990, initial processing of ENDF/B-VI and JEF-2 exposed a number of problems that had to be fixed. In addition, the ACER module was rewritten to clean it up, to add capabilities to produce ACE dosimetry and photoatomic libraries, and to provide for the convenient generation of files in several different formats for users away from Los Alamos. A MIXR module was added to NJOY, mostly to allow elemental cross sections to be reconstructed from ENDF/B-VI isotopes for plotting purposes. A new technique was introduced in GROUPR and all the output modules that provided for more efficient processing of fission and photon production matrices with lots of low-energy groups. Major revisions were made to the MATXS format to allow for charged-particle cross sections, to pack matrices with lots of low-energy groups more efficiently, and to make inserting and extracting materials easier. Initial capabilities for processing the ENDF/B-VI photoatomic data files for ACER were added. The WIMSR module, which had been under development for a number of years in cooperation with WIMS users in Canada and Mexico, was introduced into NJOY. Finally, the PURR module for generating unresolved-region probability tables for use with MCNP, which also had been under development for several years, was formally added to the code. The result of this block of work was NJOY91 and a complete new user's manual (which is still the primary reference for NJOY as of 2010).

A number of modest changes were made through the rest of 1991, including T-dependent weighting functions and additional group structures. There were additional changes in 1992, including changes to improve portability revealed by running a Fortran analysis code available at the time. In 1993, additional options were added in PLOTR for preparing 2-D and 3-D plots for GENDF data. Work was done on the handling of ENDF-6 energy-angle distributions for ACER. The WIMSR module was improved based on suggestions from Fortunado Aguilar in Mexico. PLOTR was modified to allow the presentation of $S(\alpha, \beta)$ data. A fix to the Kalbach option for energy-angle distributions was made based on work by Bob Seamon (LANL). The energy-dependence of fission Q values was added to HEATR. Also, work was done in HEATR to properly handle the damage energy cutoff at low energies. A number of additional changes were made in 1994. All the work from 1991 through 1994 was gathered together, the new LEAPR module for producing thermal scattering laws and a significant update to PURR were added, and a new major release called NJOY94 was released.

NJOY development continued during 1995 with various bug fixes. Capabilities for processing gas production and radionuclide production were enhanced, and the GASPR module was added to generate pointwise gas production cross sections by summing up the various contributions on the ENDF evaluations. A capability to include radiation damage cross sections on ACE file for MCNP was added. Moving on into 1996, some enhancements were made to the ACER plots, and other improvements to the appearance of all NJOY graphics output were made. A change was made in the erfc function to improve the consistency of 1/v cross sections in the thermal range (observed by Cecil Lubitz of KAPL).

Later in 1996, we began to make changes to support new evaluations that extended the maximum energy to 150 MeV instead of the normal 20 MeV. The highly forward peaked distributions found at high energies took some work to handle well. New methods for handling recoil distributions in File 6 were developed working with Mark Chadwick (LANL). Updates were made to ACER to handle anisotropic photon production for MCNP. In 1997, color was added to PLOTR and VIEWR with support for graphs from DTFR, COVR, HEATR, and ACER. The shaded colors used for covariance plots made a nice improvement. In mid 1997, a number of improvements to WIMSR based on the work of Andrej Trkov (IJS Slovenia). A more accurate erfc function was imported from the SLATEC library to improve doppler broadening. A capability was added to BROADR to generate standard thermal integral quantities at room temperature. Changes were made in GROUPR to handle charged-particle production given for discrete levels. Work was done to activate the capability of using PURR to generate unresolved-range probability tables and pass them to MCNP. Additional changes were made to help with portability between 32-bit and 64-bit machines and with stack-based memory allocation, which requires careful attention to SAVE statements. Changes were made to remove Hollerith literals. The pioneering free-format input developed for early NJOY versions was retired in favor of the Fortran standard READ* list-directed input method. Other changes were made for generating energy grids in RECONR and BROADR to allow 9-digit energies when 7-digit values were insufficient. The result of all this work was a new release called NJOY97. It was supported by an expanded suite of test problems and a new revision control system (UPD).

The changes to NJOY97 started out in 1998 with some additional code compatibility fixes for single and double precision. A capability was added to PURR and ACER to produce conditional probabilities for heating in the MCNP unresolved-range probability tables. The tolerances for reconstruction in RECONR and BROADR were modified to use a tighter tolerance in the thermal range, which help preserve the 0.0253-eV cross section and other thermal parameters better (noted by Cecil Lubitz of KAPL). In addition, BROADR was modified to assume a 1/v extension below the lowest point in order to better match current versions of SIGMA1 (Red Cullen, LLNL). Some additional changes were made for 150-MeV evaluations, such as the `LTT=3` option. A change was made to NJOY input to allow comment cards in the decks (they start with two dashes and a space). Piet DeLeege (Delft), Sandro Pelloni (PSI), and Andrej Trkov (IJS/Slovenia) were helpful in finding problems during this period. The performance of RECONR at low energies for R-function evaluations was improved. A number of additional code compatibility improvements were made after scanning the program with a Fortran checking code after Giancarlo Panini (Italy) questioned small differences seen using different compilers. In 1999, we improved the treatment for the damage threshold in HEATR and provided default displacement energies. We improved the values used for the neutron mass, the constant pi, and other basic parameters. We are trying to make sure that NJOY uses a consistent set of fundamental physics constants throughout the code. The channel-spin patches for Reich-Moore resonances were provided by Nancy Larson (ORNL). Combining these things with a number of less striking patches resulted in NJOY99. This version of NJOY (with its many updates) is still active in 2010.

We began making patches to NJOY99 in the spring of 2000, starting with fixing a problem with the cold hydrogen and deuterium calculations in LEAPR. This was followed by a number of other small patches. A capability was added for processing anisotropic charged particle emission in ACER. In 2001, the series of MT numbers from 875–891 was installed to represent levels in the (n,2n) reaction (needed for a new Be-9 evaluation). We added a capability to include delayed neutron data in the ACE files to feed a new capability in MCNP. A photonuclear capability was added to MATSR—this enables the TRANSX code to generate fully coupled sets for nγ transport. Some coding was added to generate fluorescence data for MCNP using the existing format with new numbers coming from the ENDF/B-VI atomic data. This work does not completely support all the atomic data now available in ENDF/B-VI. The ACER consistency checks were upgraded to include delayed neutrons, and plots for nubar and the delayed neutron spectra were added. In addition, delayed neutron processing was generalized to allow for 8 time bins as used by the JEFF evaluations. PLOTR was modified to allow ratio and difference plots using the right-hand scale. The default tolerances used in RECONR and BROADR were tightened up a bit. Some changes to the heating for photoatomic data were provided by Morgan White (LANL). A change was made to HEATR to provide the photon contribution to heating using a special MT number (442). When passed to MCNP, it allows the code to get good answers for heating even when photons are not being transported. Late in 2002, LEAPR was updated to include coherent elastic scattering for FCC and BCC crystalline lattices. In 2004, some extensions to the energy grid used for incoherent inelastic scattering were made. A few additional smaller patches were also made during this period. In 2005, changes proposed for THERMR and LEAPR by Margarete Mattes (IKE/Stuttgart) to support the new IAEA-sponsored evaluations for thermal scattering in water, heavy water, and ZrH were installed. Some additional group structures used in Europe were added. Through this period, we were always increasing the storage space allowed as we adapted to newer and larger evaluations coming out for ENDF/B-VII. During the summer, a number of changes to covariance processing were provided by Andrej Trkov (working at IAEA). Early in 2006, a new sampling scheme for thermal scattering was developed for MCNP that uses

continuous distributions for secondary energy instead of the previous discrete values. This removed unsightly artifacts in computed fluxes at low energies, and it alleviated some problems that the cold-neutron-source people were having. More code improvements were made based on detailed compiler checking. Some errors in the treatment of energy-dependent fission Q were fixed based on a review of the work of Dave Madland (LANL). An approximate treatment for the relativistic gamma in the ENDF/B-VII evaluation for n+^{1}H was added in HEATR and GROUPR based on theoretical work from Gerry Hale (LANL). We added a new plot to the ACER set that shows the recoil part of the heating. This is a sensitive test of energy-balance. In the summer of 2007, we added some smoothing options to make the low-energy shape of neutron distributions look more like the theoretical shape, namely sqrt(E'). Similar changes were provided for delayed neutron spectra. Additional smoothing was provided for some of the fission spectra at energies above 10 MeV using an exponential shape. A change was made in GROUPR to override Cartesian interpolation in favor of unit-base interpolation for scattering distributions. This give smoother scattering source functions and is consistent with what MCNP does. In 2007, a big change was made to covariance processing by replacing the original NJOY ERRORR module with ERRORJ as contributed to the NJOY project by Japan (Go Chiba). This new module added covariance capabilities for the more modern resolved-resonance representations, angular distributions, and secondary-energy distributions. A series of additional changes to ERRORJ were made over the next couple of years. Go Chiba, Andrej Trkov (IAEA and IJS Slovenia), and Ramon Arcilla (BNL) were involved in this. A capability to hand energy-dependent scattering radius data in the unresolved range was added. In 2008, ERRORJ work continued. some work was done in PLOTR to implement graphs of GROUPR emission spectra. In 2009, changes were made for unresolved resonance cross sections to force log-log interpolation to better represent $1/v$ cross sections. The parameter that looked for steps in the unresolved-range energy grids that were too large was changed from its former value of 3 to 1.26. Some of the steps in ENDF/B-VII are unreasonably large when representing $1/v$ cross sections. The default energy grid used in the unresolved range was upgraded to one using about 13 points per decade. Additional pages were added to the ACER plots to display the unresolved-range self-shielded cross sections. This work was influenced by Red Cullen (LLNL). In a related effort, some changes were made in the binning logic for PURR to improve unresolved-range results. In 2010, changes were made to support the processing of the IRDF international dosimetry file. Some work was done on the photonuclear options to support the TENDL-2009 library. More ERRORJ changes from Trkov were implemented, and a capability to process the uncertainty in the scattering radius was added.

It is fairly clear from this account that NJOY has been blessed with many contributions from the world-wide community of users. The contributions of the International Atomic Energy Agency (IAEA) from its code comparison project, its hosting of one of the authors (REM) during the early work on the NJOY plotting package, and its development of the Fusion Evaluated Data Library (FENDL) should be mentioned. Nuclear data evaluations aren't fully useful if they cannot be processed, and this kind of testing can provide changes to both the codes and the evaluations to complete the design system. The many users of NJOY in the EU countries with their active nuclear industry also provided many useful items of feedback, and the NEA Data Bank has done us a great service by hosting the NJOY User Group, an NJOY email forum, and a number of opportunities for face-to-face meeting.

This summary of the development of NJOY is necessarily brief. For more details, see the NJOY web pages, which include complete histories for NJOY91, NJOY94, NJOY97, and NJOY99.

For a number of years, the development of a Fortran-90/95 version of NJOY has been carried on in parallel with the work on NJOY99. Using F95 alleviates many of the problems with numerical precision, COMMON blocks, data typing, dynamic data storage, and code structure that we have seen in the past. This code version is now called NJOY10, and it is strongly based on F95 module and dynamic data storage concepts. In most regards, NJOY10 is functionally consistent with NJOY99. However, there are some new features that are not supported in the older code. We imported methods from SAMMY (Nancy Larson, ORNL) for handling the Reich-Moore-Limited (RML) resonance representation. In addition to providing RML cross sections for RECONR, these methods allow us to compute angular distributions from resonance parameters in RECONR and sensitivities of cross sections to resonance parameters in ERRORR. In THERMR, we provided an option to generate scattering data using (E, μ, E') ordering in addition to the normal (E, E', μ) ordering with output in the File 6/Law 7 format. These kinds of data can be more directly compared to experiment, and this ordering is used in some Monte Carlo codes.

[1] M. B. Chadwick *et al*, "ENDF/B-VII.0: Next Generation Evaluated Nuclear Data Library for Nuclear Science and Technology," *Nuclear Data Sheets* **107**, 2931-3060 (2006).

[2] M. Herman and A. Trkov, Eds., "ENDF-6 Formats Manual, Data Formats and Procedures for the Evaluated Nuclear Data File ENDF/B-VI and ENDF/B-VII," Brookhaven National Laboratory report BNL-90365-2009 (ENDF-102) (June 2009).

[3] R. E. MacFarlane, D. W. Muir, and R. M. Boicourt, "The NJOY Nuclear Data Processing System, Vol. I: User's

Manual," Los Alamos National Laboratory report LA-9303-M (ENDF 324) (May 1982).
[4] R. E. MacFarlane and D. W. Muir, "The NJOY Nuclear Data Processing System, Version 91," Los Alamos National Laboratory report LA-12740-M (1994).
[5] R. E. MacFarlane, "NJOY10 Manual," to be published.
[6] See the web URL http://t2.lanl.gov/codes/njoy99 or http://t2.lanl.gov/codes/NJOY10.
[7] X-5 Monte Carlo Team, "MCNP—A General Monte Carlo N-Particle Transport Code, Version 5," Los Alamos National Laboratory report LA-UR-03-1987 (April 2003).
[8] R. Douglas O'Dell, "Standard Interface Files and Procedures for Reactor Physics Codes, Version IV," Los Alamos Scientific Laboratory report LA-6941-MS (September 1977).
[9] R. E. MacFarlane, "TRANSX 2: A Code for Interfacing MATXS Cross-Section Libraries to Nuclear Transport Codes," Los Alamos National Laboratory report LA-12312-MS (July 1992).
[10] W. W. Engle, Jr., "A User's Manual for ANISN: A One-Dimensional Discrete Ordinates Transport Code with Anisotropic Scattering," Oak Ridge Gaseous Diffusion Plant Computing Technology Center report K-1693 (1967).
[11] R. D. O'Dell, F. W. Brinkley, Jr., D. R. Marr, and R. E. Alcouffe, "Revised User's Manual for ONEDANT: A Code Package for One-Dimensional, Diffusion-Accelerated, Neutral-Particle Transport," Los Alamos National Laboratory report LA-9184-M, Rev. (December 1989).
[12] K. L. Derstine, "DIF3D: A Code to Solve One-, Two-, and Three-Dimensional Finite-Difference Diffusion Theory Problems," Argonne National Laboratory report ANL-82-64 (April 1984).
[13] R. E. Alcouffe, R. S. Baker, J. A. Dahl, S. A. Turner, and R. C. Ward, "PARTISN: A Time-Dependent, Parallel Neutral Particle Transport Code System," Los Alamos National Laboratory report LA-UR-05-3925 (Revised May 2005).
[14] O. Ozer, "RESEND: A Program to Preprocess ENDF/B Materials With Resonance Files into Pointwise Form," Brookhaven National Laboratory report BNL-17134 (1972).
[15] H. Henryson II, B. J. Toppel, and C. G. Stenberg, "MC^2-2: A Code to Calculate Fast Neutron Spectra and Multigroup Cross Sections," Argonne National Laboratory report ANL-8144 (ENDF-239) (1976).
[16] N. M. Larson, "Updated Users' Guide for SAMMY: Multilevel R-Matrix Fits to Neutron Data Using Bayes' Equations," Oak Ridge National Laboratory report ORNL/TM-9179/R8 (October 2008).
[17] D. E. Cullen, "Program SIGMA1 (Version 77-1): Doppler Broadened Evaluated Cross Sections in the Evaluated Nuclear Data File/Version B (ENDF/B) Format," Lawrence Livermore National Laboratory report UCRL-50400, Vol. 17, Part B (1977).
[18] M. A. Abdou, C. W. Maynard, and R. Q. Wright, "MACK: A Computer Program to Calculate Neutron Energy Release Parameters (Fluence-to-Kerma Factors) and Multigroup Reaction Cross Sections from Nuclear Data in ENDF Format," Oak Ridge National Laboratory report ORNL-TM-3994 (July 1973).
[19] D. W. Muir, "Gamma Rays, Q-Values, and Kerma Factors," Los Alamos Scientific Laboratory report LA-6258-MS (March 1976).
[20] R. E. MacFarlane, "Energy Balance of ENDF/B-V," Trans. Am. Nucl. Soc. **33**, 681 (1979). See also R. E. MacFarlane, "Energy Balance of ENDF/B-V.2," Minutes of the Cross Section Evaluation Working Group (available from the National Nuclear Data Center, Brookhaven National Laboratory, Upton, NY) (May 1984).
[21] T. A. Gabriel, J. D. Amburgy, and N. M. Greene, "Radiation-Damage Calculations: Primary Knock-On Atom Spectra, Displacement Rates, and Gas Production Rates," *Nucl. Sci. Eng.* **61**, 21 (1976).
[22] D. G. Doran, "Neutron Displacement Cross Sections for Stainless Steel and Tantalum Based on a Linhard Model," *Nucl. Sci. Eng.* **49**, 130 (1972).
[23] L. R. Greenwood and R. K. Smither, "Displacement Damage Calculations with ENDF/B-V," in Proceedings of the Advisory Group Meeting on Nuclear Data for Radiation Damage Assessment and Reactor Safety Aspects, October 12-16, 1981, IAEA, Vienna, Austria (October 1981).
[24] M. T. Robinson, in *Nuclear Fusion Reactors* (British Nuclear Energy Society, London, 1970).
[25] J. Lindhard, V. Nielsen, M. Scharff, and P. V. Thomsen, *Kgl. Dansk, Vidensk. Selsk, Mat-Fys. Medd.* **33** (1963).
[26] R. E. MacFarlane, D. W. Muir, and F. W. Mann, "Radiation Damage Calculations with NJOY," *J. Nucl. Materials* **122** and **123**, 1041 (1984).
[27] R. E. MacFarlane and D. G. Foster, Jr., "Advanced Nuclear Data for Radiation Damage Calculations," *J. Nucl. Materials* **122** and **123**, 1047 (1984).
[28] R. E. MacFarlane, "Energy Balance Tests for ENDF/B-VII," http://t2.lanl.gov/data/ebalVII/summary.html.
[29] J. U. Koppel and D. H. Houston, "Reference Manual for ENDF Thermal Neutron Scattering Data," General Atomic report GA-8774 revised and reissued as ENDF-269 by the National Nuclear Data Center, Brookhaven National Laboratory (1978).
[30] Y. D. Naliboff and J. U. Koppel, "HEXSCAT: Coherent Scattering of Neutrons by Hexagonal Lattices," General Atomic report GA-6026 (1964).
[31] H. C. Honeck and D. R. Finch, "FLANGE–II (Version 71-1), A Code to Process Thermal Neutron Data for an ENDF/B Tape," Savannah River Laboratory report DP-1278 (ENDF-152) (1971).
[32] See "view PDF plots" for the desired material indexed in http://t2.lanl.gov/data/thermal7.html.
[33] L. B. Levitt, "The Probability Table Method for Treating Unresolved Resonances in Monte Carlo Criticality Calculations," *Trans. Am. Nucl. Soc.* **14**, 648 (1971).
[34] R. E. Prael and L. J. Milton, "A User's Manual for the Monte Carlo Code VIM," Argonne National Laboratory report FRA-TM-84 (1976). Also, R. Blomquist, "VIM continuous energy Monte Carlo transport code," in Proc. Inter. Conf. on Mathematics, Computations, Reactor Physics and Environmental Analysis, Portland, OR, April 30 – May 4, 1995.
[35] C. Kalbach and F. M. Mann, "Phenomenology of Continuum Angular Distributions. I. Systematics and Parametrization," Phys. Rev. C. **23**, 112 (1981).
[36] P. G. Young and E. D. Arthur, "GNASH: A Preequilibrium Statistical Nuclear Model Code for Calculation of Cross Sections and Emission Spectra," Los Alamos Scientific Laboratory report LA-6947 (1977).

[37] C. Kalbach, "Systematics of Continuum Angular Distributions: Extensions to Higher Energies," *Phys. Rev. C.* **37**, 2350 (1988).
[38] See the "view PDF plots" for various ENDF/B-VII materials indexed in http://t2.lanl.gov/data/neutron7.html.
[39] E. D. Cashwell, J. R. Neergaard, C. J. Everett, R. G. Schrandt, W. M. Taylor, and G. D. Turner, "Monte Carlo Photon Codes MCG and MCP," Los Alamos Scientific Laboratory report LA-5157-MS (March 1973).
[40] C. J. Everett and E. D. Cashwell, "MCP Code Fluorescence-Routine Revision," Los Alamos Scientific Laboratory report LA-5240-MS (May 1973).
[41] I. I. Bondarenko, Ed., *Group Constants for Nuclear Reactor Calculations* (Consultants Bureau, New York, 1964).
[42] G. I. Bell and S. Glasstone, *Nuclear Reactor Theory* (Van Nostrand Reinhold, New York, 1970).
[43] R. E. MacFarlane, "TRANSX-CTR: A Code for Interfacing MATXS Cross-Section Libraries to Nuclear Transport Codes for Fusion Systems Analysis," Los Alamos National Laboratory report LA-9863-MS (February 1984).
[44] R. E. MacFarlane, "ENDF/B-IV and -V Cross Sections for Thermal Power Reactor Analysis," in Proc. Intl. Conf. of Nuclear Cross Sections for Technology, Knoxville, TN (October 22-26 1979), National Bureau of Standards Publication 594 (September 1980).
[45] M. Abramowitz and I. Stegun, *Handbook of Mathematical Functions* (Dover Publications, New York, 1965).
[46] J. H. Hubbell, Wm. J. Viegle, E. A. Briggs, R. T. Brown, D. T. Cromer, and R. J. Howerton, "Atomic Form Factors, Incoherent Scattering Functions, and Photon Scattering Cross Sections," *J. Phys. Chem. Ref. Data* **4**, 471 (1975).
[47] S. A. W. Gerstl, D. J. Dudziak, and D. W. Muir, "Cross Section Sensitivity and Uncertainty Analysis with Application to a Fusion Reactor," *Nucl. Sci. Eng.* **62**, 137 (January 1977).
[48] W. A. Reupke, D. W. Muir, and J. N. Davidson, "Consistency of Neutron Cross-Section Data, S_n Calculations, and Measured Tritium Production for a 14-MeV Neutron-Driven Sphere of Natural-Lithium Deuteride," *Nucl. Sci. Eng.* **82**, 416 (December 1982).
[49] D. W. Muir, "Evaluation of Correlated Data Using Partitioned Least Squares: A Minimum-Variance Derivation," *Nucl. Sci. and Eng.* **101**, 88-93 (January 1989).
[50] G. Chiba, "ERRORJ, A Code to Process Neutron-Nuclide Reaction Covariance, Version 2.3," JAEA-Data/Code 2007-007, March 2007.
[51] C. R. Weisbin, E. M. Oblow, J. Ching, J. E. White, R. Q. Wright, and J. Drischler, "Cross Section and Method Uncertainties: The Application of Sensitivity Analysis to Study Their Relationship in Radiation Transport Benchmark Problems," Oak Ridge National Laboratory report ORNL-TM-4847 (ENDF-218) (August 1975). (See especially Chapter IV.)
[52] J. D. Smith III, "Processing ENDF/B-V Uncertainty Data into Multigroup Covariance Matrices," Oak Ridge National Laboratory report ORNL/TM-7221 (ENDF-295) (June 1980).
[53] P. G. Young, "Evaluation of n + ^{7}Li Reactions using Variance-Covariance Techniques," *Trans. Am. Nucl. Soc.* **39**, 272 (1981).
[54] P. G. Young, J. W. Davidson, and D. W. Muir, "Evaluation of the ^{7}Li(n,n't)^{4}He Cross Section for ENDF/B-VI and Application to Uncertainty Analysis," *Fusion Tech.* **15**, 440-448 (1989).
[55] D. W. Muir and R. J. LaBauve, "COVFILS: A 30-Group Covariance Library Based on ENDF/B-V," Los Alamos National Laboratory report LA-8733-MS (ENDF-306) (March 1981).
[56] S. A. W. Gerstl, "SENSIT: A Cross-Section and Design Sensitivity and Uncertainty Analysis Code," Los Alamos Scientific Laboratory report LA-8498-MS (August 1980).
[57] M. J. Embrechts, "SENSIT-2D: A Two-Dimensional Cross-Section Sensitivity and Uncertainty Analysis Code," Los Alamos National Laboratory report LA-9515-MS (October 1982).
[58] D. W. Muir, "COVFILS-2: Neutron Data and Covariances for Sensitivity and Uncertainty Analysis," *Fusion Tech.* **10** (3), Part 2B, 1461 (November 1986).
[59] D. W. Muir, R. E. MacFarlane, and R. M. Boicourt, "Multigroup Processing of ENDF/B Dosimetry Covariances," Proc. 4th ASTM-EURATOM Symp. on Reactor Dosimetry, Gaithersburg, Maryland, March 22-26, 1982, NUREG/CP-0029 (CONF-820321), p. 655 (1982).
[60] J. U. Koppel, J. R. Triplett, and Y. D. Naliboff, "GASKET: A Unified Code for Thermal Neutron Scattering," General Atomics report GA-7417 (Rev.) (March 1967).
[61] R. C. F. McLatchie, 1962, unpublished.
[62] A. T. Butland, "LEAP and ADDELT, A Users Guide to Two Complementary Codes on the ICL-470 for Calculating The Scattering Law From a Phonon Frequency Function," Atomic Energy Establishment Winfrith report AEEW-M-1200, 1973.
[63] D. J. Picton, personal communication to G. J. Russell, 1987.
[64] R. E. MacFarlane, "New Thermal Neutron Scattering Files for ENDF/B-VI Release 2," Los Alamos National Laboratory report LA-12639-MS (1994).
[65] R. E. MacFarlane, "Cold Moderator Scattering Kernels," in *Advanced Neutron Sources 1988, Proceedings of the 10th Meeting of the International Collaboration on Advanced Neutron Sources (ICANS-X) held at Los Alamos, 3–7 October 1988*, Institute of Physics Conferences Series Number 97 (Institute of Physics, Bristol and New York, 1988), p.157.
[66] R. E. MacFarlane, "Cold Moderator Scattering Kernels," presented at the International Workshop on Cold Neutron Sources, March 5–8, 1990, Los Alamos, New Mexico, Los Alamos National Laboratory report LA-12146-C (August 1991).
[67] D. E. Parks, M S. Nelkin, J. R. Beyster, and N. F. Wikner, *Slow Neutron Scattering and Thermalization*, W. A. Benjamin, Inc. (New York, 1970).
[68] R. W. G. Wycoff, *Crystal Structures*, John Wiley & Sons, New York, London, 1963.
[69] J. A. Young and J. U. Koppel, Phys. Rev. **135**, A603(1964).
[70] P. A. Egelstaff, B. C. Haywood, and F. J. Webb, Proc. Phys. Soc., **90**, 681(1967).
[71] W. Schott, Z. Physik **231**, 243(1970).
[72] M. Utsuro, Z. Physik B **27**, 111(1977).
[73] J. Keinert and J. Sax, "Investigation of Neutron Scattering Dynamics in Liquid Hydrogen and Deuterium for Cold Neutron Sources," Kerntechnik **51**, 19 (1987).
[74] Adobe Systems Incorporated, *Postscript Language Reference Manual*, Addison-Wesley Publishing Company, 1985.

[75] K. D. Lathrop, "DTF-IV, A FORTRAN Program for Solving the Multigroup Transport Equation with Anisotropic Scattering," Los Alamos Scientific Laboratory report LA-3373 (November 1965).

[76] R. J. Barrett and R. E. MacFarlane, "Coupled Neutron and Photon Cross Sections for Transport Calculations," Los Alamos Scientific Laboratory report LA-7808-MS (April 1979).

[77] B. M. Carmichael, "Standard Interface Files and Procedures for Reactor Physics Codes, Version III," Los Alamos Scientific Laboratory report LA-5486-MS (February 1974).

[78] D. R. Harris, R. J. LaBauve, R. E. MacFarlane, P. D. Soran, C. R. Weisbin, and J. E. White, "MINX, A Modular Code System for Processing Multigroup Cross Sections from Nuclear Data in ENDF/B Format," Los Alamos National Laboratory report LA-UR-1766 (1973).

[79] R. B. Kidman and R. E. MacFarlane, "LIB-IV: A Library of Group Constants for Nuclear Reactor Calculations," Los Alamos Scientific Laboratory report LA-6260-MS (March 1976).

[80] R. E. MacFarlane and R. B. Kidman, "LINX and BINX: CCCC Utility Codes for the MINX Multigroup Processing Code," Los Alamos Scientific Laboratory report LA-6219-MS (February 1976).

[81] "CINX: Collapsed Interpretation of Nuclear X Sections," Los Alamos Scientific Laboratory report LA-6287-MS (April 1976).

[82] W. J. Davis, M. B. Yarbrough, and A. B. Bortz, "SPHINX: A One Dimensional Diffusion and Transport Nuclear Cross Section Processing Code," Westinghouse Advanced Reactors Division report WARD-XS-3045-17 (August 1977).

[83] C. L. Cowan, B. A. Hutchins, and J. E. Turner, "TDOWN – A Code to Generate Composition and Spatially Dependent Cross Sections," General Electric report GEAP-13740 (August 1971).

[84] R. J. LaBauve, T. R. England, D. C. George, R. E. MacFarlane, and W. B. Wilson, "DANDE—A Linked Code System for Core Neutronics/Depletion Analysis," Los Alamos National Laboratory report LA-10412-MS (June 1985).

[85] R. E. MacFarlane and D. C. George, "UPD: A Portable Version-Control Program," Los Alamos National Laboratory report LA-12057-MS (April 1991).

[86] Y. Farawila, Y. Gohar, and C. Maynard, "KAOS/LIB-V: A Library of Nuclear Response Functions Generated by KAOS-V Code From ENDF/B-V and Other Data Files," Argonne National Laboratory report ANL/FPP/TM-241 (April 1989).

[87] J. R. Askew, F. J. Fayers, and P. B. Kemshell, "A General Description of the Lattice Code WIMS," *J. Brit. Nucl. Energy Soc.* **5**, p. 564 (1966).