3 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 pointwise-ENDF (PENDF) file with all cross sections on a unionized energy grid suitable for linear interpolation to within a specified tolerance. Redundant reactions (for example, total inelastic, charged-particle reactions) 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. RECONR has the following features:

- Efficient use of dynamic storage allocation and a special stack structure allow very large problems to be run.
- The unionized grid improves the accuracy, usefulness, and ENDF compatibility of the output. All summation cross sections are preserved on the union grid. Up to nine significant figures are allowed.
- A correct directory of the output tape is provided.
- Approximate $\psi\chi$ Doppler broadening may be used in some cases to speed up reconstruction.
- A resonance-integral criterion is added to the normal linearization criterion in order to reduce the number of points added to the tabulation to represent "unimportant" resonances.
- All ENDF-6 resonance formats currently active are handled, including the calculation of angular distributions from resonance parameters in some cases.

This chapter describes the RECONR module in NJOY2016.0.

3.1 ENDF/B Cross Section Representations

A typical cross section derived from an ENDF/B evaluation is shown in Fig. 1. The low-energy cross sections are "smooth". They are described in File 3 (see Section 2.4 for a review of ENDF/B nomenclature) 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 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 supported (Single-Level Breit-Wigner (SLBW), Multilevel Breit-Wigner (MLBW), Adler-Adler, Hybrid R-Function (HRF), Reich-Moore (RM), Reich-Moore-Limited (RML), energy-independent unresolved, and energy-dependent unresolved). The Adler-Adler and Hybrid formats are not being used in modern evaluations. For an increasing number of modern evaluations, the low energy "smooth" region is omitted, and the resolved resonance region is extended to the low energy limit.

RECONR takes these separate representations and produces a simple cross section versus energy representation like the one shown in Fig. 1.

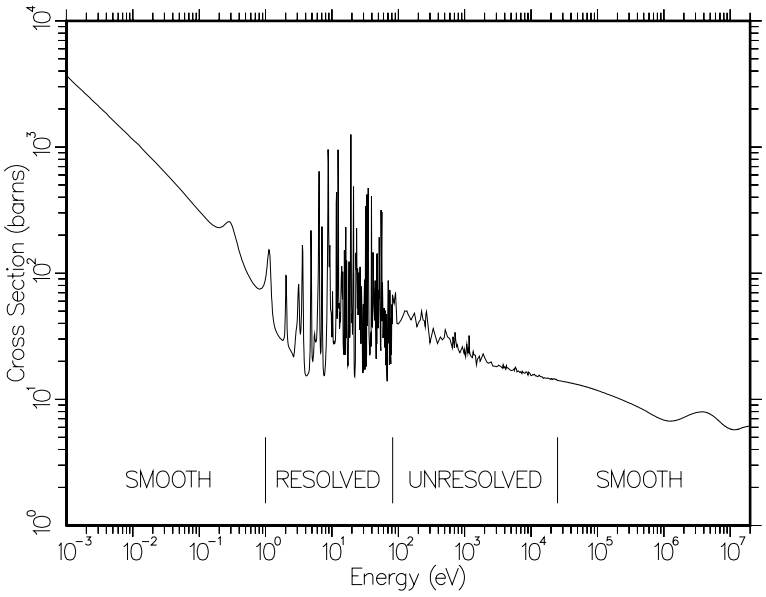

Figure 1: A typical cross section reconstructed from an ENDF/B evaluation using RECONR. The smooth, resolved, and unresolved energy regions use different representations of the cross sections. This is the total cross section for ^{235}U from ENDF/B-V.

3.2 Unionization and Linearization Strategy

Several of the cross sections found in ENDF/B evaluations are summation cross sections (for example, total, inelastic, sometimes (n,2n) or fission, and sometimes charged-particle 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.

The use of linear-linear interpolation (*i.e.,* 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), and, 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. As described in more detail below, RECONR makes one pass through the ENDF/B material to select the energy grid, and then a second pass to compute cross sections on this grid. Each cross section on the PENDF file (except for the summation cross sections) is exactly equal to its ENDF/B value. The summation cross sections are then obtained by adding up the partial cross sections at each grid point.

While RECONR is going through the reactions given in the ENDF/B 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

$$\text{threshold} \geq \frac{A+1}{A}\,Q \tag{4}$$

is not true, the threshold energy is moved up to satisfy the condition. This is usually a small change, often only in the least significant digit, and is a consequence of comparing two REAL numbers of finite precision.

If desired, the unionized grid developed from the ENDF/B 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 are not sufficiently accurate. Even some modern evaluations use inadequate energy grids for the unresolved range. 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. "Large" is currently defined by `wide` to be a factor of 1.26.

3.3 Linearization and Reconstruction Methods

Linearization (`lunion`) and resonance reconstruction (`resxs`) both function by inserting new energy grid points between the points of an original grid using an "inverted stack". The general concepts involved are illustrated with a simple example shown in Fig. 2.

The stack is first primed with two starting values. For linearization, they will be two adjacent points on the original union grid. For reconstruction, they will usually be the peaks and half-height energies of resonances. The stack is said to be inverted because the lower energy is at the "top" (`I`=2).

This interval or panel is now divided into two parts, and the cross section computed at the intermediate point is compared to the result of linear interpolation between the adjacent points. If the two values do not agree within various criteria, the top of the stack is moved up one notch (`I`=3), and the new value is inserted (`I`=2). The code then repeats the checking process for the new (smaller) interval at the top of the stack. The top of the stack rises until convergence is achieved for the top interval. The top energy and cross section are then saved on a scratch file, the stack index is decremented, and the checks are repeated. This

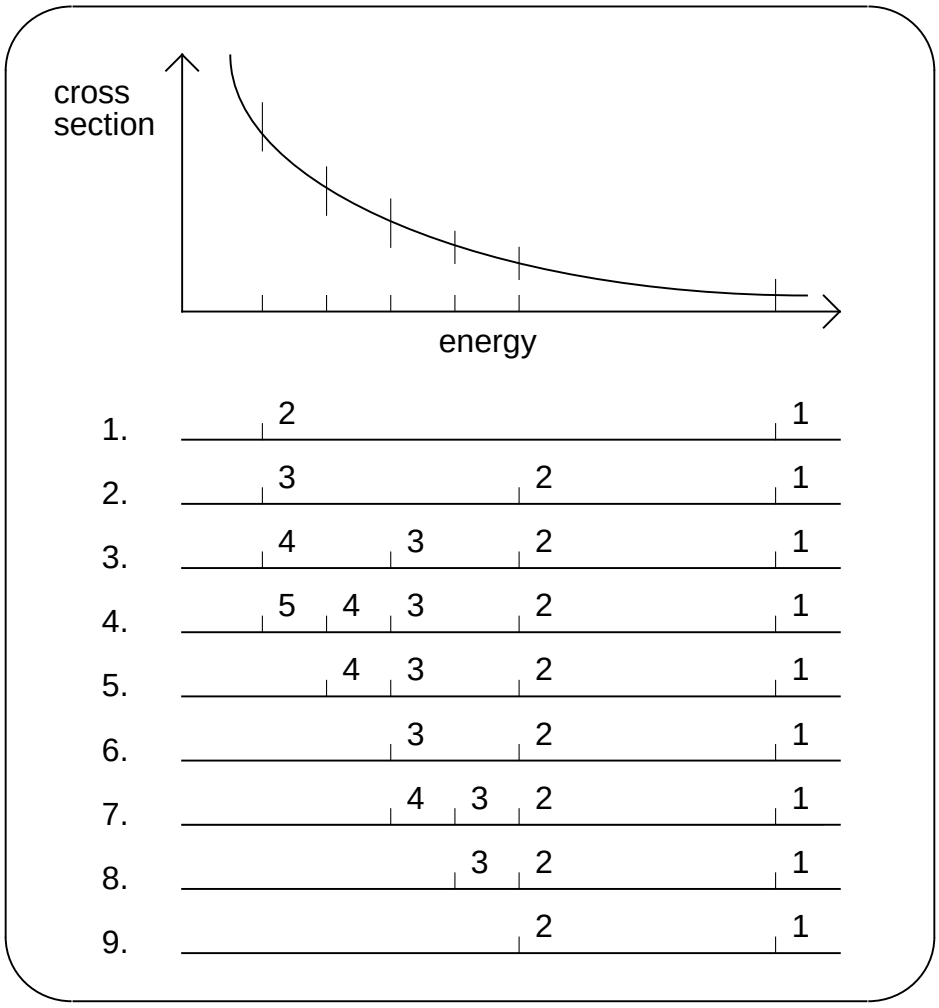

Figure 2: Inverted -stack method used in RECONR and several other places in NJOY. Line 1 shows the two initial points (the lower energy is higher in the stack). In line 2, a new point has been calculated at the midpoint, but the result was not converged, and the new point has been inserted in the stack. In line 3, the midpoint of the top panel has been checked again, found to be not converged, and inserted into the stack. The same thing happens in line 4. In line 5, the top panel is found to be converged, and the top point (5) has been written out. The same thing happens in line 6. In line 7, the top panel is tested and found to be not converged. The midpoint is added to the stack. Finally, in line 8, the top panel is found to be converged, and the top point is written out. This leaves two points in the stack (see line 9). Note that the energy points come off the stack in the desired order of increasing energy, and that only one point has to be moved up in the stack as each new result is inserted.

process is continued with the top of the stack rising and falling in response to the complexity of the cross section until the entire panel ΔE has been converged (I=1). The stack is then reprimed with the bounds of the next panel. The process continues until the entire energy range for linearization or reconstruction has been processed.

This stack logic enables a panel to be subdivided into parts as small as $\Delta E/2^n$ where n is the stack size, and several different cross sections (elastic, capture, fission) can easily be stored in arrays of this size.

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 use ENDF 11-character fields) is 7 significant figures (that is, $\pm 1.234567 \pm n$) rather than the usual 5 or 6 (see Section 2.4). For NJOY2016, the Fortran-90 "kind" parameter is used to assure sufficient precision for this. 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 but very small loss in accuracy results.

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 1% 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. 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 from ENDF/B-VII.0 ^{235}U follows:

```
             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    =  242600
```

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. The error from significant figure reduction provided by earlier versions of NJOY is no longer needed.

For energies in the thermal range (energies less than `trange`=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.

3.4 Resonance Representations

RECONR uses the resonance formulas as implemented in the original RESEND code[26] with several changes: a more efficient calculation of MLBW cross sections developed by C. R. Lubitz of the Knolls Atomic Power Laboratory (KAPL) and coded by P. Rose of the National Nuclear Data Center (NNDC) at the Brookhaven National Laboratory (BNL), the addition of competitive widths introduced for ENDF/B-V, a $\psi\chi$ Doppler-broadening calculation for SLBW and Adler-Adler resonance shapes, and a capability to process either the multi-level multi-channel R-matrix Reich-Moore parameters or the multi-level single-channel Hybrid R-Function parameters based on the work of M. Bhat and C. Dunford of the NNDC, an implementation of the GH method for MLBW resonances, which allows psi-chi broadening, and a capability to process the new RML parameters, including resolved resonance energy region angular distributions.

Expanded discussions of the following formulas can be found in the ENDF-6 format manual[9].

Single-Level Breit-Wigner Representation (SLBW) The subroutine that computes Single-Level Breit-Wigner cross sections (`cssLbw`) uses

$$\begin{aligned}
\sigma_n &= \sigma_p \\
&\quad + \sum_\ell \sum_r \sigma_{mr} \Big\{ \Big[\cos 2\phi_\ell - (1 - \frac{\Gamma_{nr}}{\Gamma_r}) \Big] \, \psi(\theta, x) \\
&\quad + \sin 2\phi_\ell \, \chi(\theta, x) \Big\} \ , \qquad (5) \\
\sigma_f &= \sum_\ell \sum_r \sigma_{mr} \frac{\Gamma_{fr}}{\Gamma_r} \, \psi(\theta, x) \ , \qquad (6) \\
\sigma_\gamma &= \sum_\ell \sum_r \sigma_{mr} \frac{\Gamma_{\gamma r}}{\Gamma_r} \, \psi(\theta, x) \ , \text{ and} \qquad (7) \\
\sigma_p &= \sum_\ell \frac{4\pi}{k^2} (2\ell + 1) \sin^2 \theta_\ell \ , \qquad (8)
\end{aligned}$$

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)

$$\sigma_{mr} = \frac{4\pi}{k^2}\, g_J\, \frac{\Gamma_{nr}}{\Gamma_r} \ , \tag{9}$$

where g_J is the spin statistical factor

$$g_J = \frac{2J+1}{4I+2} \ , \tag{10}$$

and 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:

$$k = (2.196771\times10^{-3})\frac{A}{A+1}\sqrt{E} \ . \tag{11}$$

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

$$a = 0.123\, A^{1/3} + 0.08 \ . \tag{12}$$

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

$$\Gamma_{nr}(E) = \frac{P_\ell(E)\,\Gamma_{nr}}{P_\ell(|E_r|)} \ , \tag{13}$$

where

$$P_0 = \rho \ , \tag{14}$$

$$P_1 = \frac{\rho^3}{1+\rho^2} \ , \tag{15}$$

$$P_2 = \frac{\rho^5}{9+3\rho^2+\rho^4} \ , \tag{16}$$

$$P_3 = \frac{\rho^7}{225+45\rho^2+6\rho^4+\rho^6} \ , \text{ and} \tag{17}$$

$$P_4 = \frac{\rho^9}{11025+1575\rho^2+135\rho^4+10\rho^6+\rho^8} \ , \tag{18}$$

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

$$\phi_0 = \hat{\rho} \ , \tag{19}$$

$$\phi_1 = \hat{\rho} - \tan^{-1} \hat{\rho} \ , \tag{20}$$

$$\phi_2 = \hat{\rho} - \tan^{-1} \frac{3\hat{\rho}}{3-\hat{\rho}^2} \ , \tag{21}$$

$$\phi_3 = \hat{\rho} - \tan^{-1} \frac{15\hat{\rho}-\hat{\rho}^3}{15-6\hat{\rho}^2} \ , \text{ and} \tag{22}$$

$$\phi_4 = \hat{\rho} - \tan^{-1} \frac{105\hat{\rho}-10\hat{\rho}^3}{105-45\hat{\rho}^2+\hat{\rho}^4} \ , \tag{23}$$

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,

$$\psi = \frac{1}{1+x^2} \ , \tag{24}$$

$$\chi = \frac{x}{1+x^2} \ , \tag{25}$$

$$x = \frac{2(E-E_r')}{\Gamma_r} \ , \tag{26}$$

and

$$E_r' = E_r + \frac{S_\ell(|E_r|) - S_\ell(E)}{2(P_\ell(|E_r|)} \, \Gamma_{nr}(|E_r|) \ , \tag{27}$$

in terms of the shift factors

$$S_0 = 0 \ , \tag{28}$$

$$S_1 = -\frac{1}{1+\rho^2} \ , \tag{29}$$

$$S_2 = -\frac{18+3\rho^2}{9+3\rho^2+\rho^4} \ , \tag{30}$$

$$S_3 = -\frac{675+90\rho^2+6\rho^4}{225+45\rho^2+6\rho^4+\rho^6} \ , \text{ and} \tag{31}$$

$$S_4 = -\frac{44100+4725\rho^2+270\rho^4+10\rho^6}{11025+1575\rho^2+135\rho^4+10\rho^6+\rho^8} \ . \tag{32}$$

To go to higher temperatures, define

$$\theta = \frac{\Gamma_r}{\sqrt{\frac{4kTE}{A}}} \ , \tag{33}$$

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

$$\psi = \frac{\sqrt{\pi}}{2}\,\theta\,\mathrm{Re}W\left(\frac{\theta x}{2},\frac{\theta}{2}\right)\ , \tag{34}$$

and

$$\chi = \frac{\sqrt{\pi}}{2}\,\theta\,\mathrm{Im}W\left(\frac{\theta x}{2},\frac{\theta}{2}\right)\ , \tag{35}$$

in terms of the complex probability function (see `quickw`, `wtab`, and `w`, which came from the MC^2 code[37])

$$W(x,y) = \mathrm{e}^{-z^2}\,\mathrm{erfc}(-iz) = \frac{i}{\pi}\int_{-\infty}^{\infty}\frac{\mathrm{e}^{-t^2}}{z-t}\,dt\ , \tag{36}$$

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; however, the $\psi\chi$ method is less expensive than BROADR. Previous versions of RECONR included $\psi\chi$ broadening for the SLBW and Adler-Adler representations only. This version also allows the method to be used for MLBW cases. The SLBW approach can produce negative elastic cross sections. If found, they are set to a small positive value, and a count is accumulated for a diagnostic in the listing file.

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

$$\sigma_n(E) = \frac{\pi}{k^2}\sum_{\ell}\sum_{s=|I-\frac{1}{2}|}^{I+\frac{1}{2}}\sum_{J=|l-s|}^{l+s} g_J\,|1-U_{nn}^{\ell sJ}(E)|^2\ , \tag{37}$$

with

$$U_{nn}^{\ell J}(E) = \mathrm{e}^{2i\phi_\ell} - \sum_r \frac{i\Gamma_{nr}}{E_r' - E - i\Gamma_r/2}\ , \tag{38}$$

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

$$\sigma_n(E) = \frac{\pi}{k^2}\sum_{\ell}\sum_{s=|I-\frac{1}{2}|}^{I+\frac{1}{2}}\sum_{J=|l-s|}^{l+s} g_J\left\{\left(1-\cos 2\phi_\ell - \sum_r \frac{\Gamma_{nr}}{\Gamma_r}\frac{2}{1+x_r^2}\right)^2 + \left(\sin 2\phi_\ell + \sum_r \frac{\Gamma_{nr}}{\Gamma_r}\frac{2x_r}{1+x_r^2}\right)^2\right\}, \quad (39)$$

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 Γ_J=$\Gamma_{s_1 J}$+$\Gamma_{s_2 J}$. 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:

$$\sigma_n(E) = \frac{\pi}{k^2}\sum_{\ell}\left[\sum_J g_J\left\{\left(1-\cos 2\phi_\ell - \sum_r \frac{\Gamma_{nr}}{\Gamma_r}\frac{2}{1+x_r^2}\right)^2 + \left(\sin 2\phi_\ell + \sum_r \frac{\Gamma_{nr}}{\Gamma_r}\frac{2x_r}{1+x_r^2}\right)^2\right\} + 2D_\ell(1-\cos 2\phi_\ell)\right], \quad (40)$$

where the summation over J now runs from

$$||I-\ell|-\frac{1}{2}| \rightarrow I+\ell+\frac{1}{2}, \quad (41)$$

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

$$D_\ell = \sum_{s=|I-\frac{1}{2}|}^{I+\frac{1}{2}}\sum_{J=|l-s|}^{l+s} g_J - \sum_{J=||I-\ell|-\frac{1}{2}|}^{I+\ell+\frac{1}{2}} g_J \quad (42)$$

$$= (2\ell+1) - \sum_{J=||I-\ell|-\frac{1}{2}|}^{I+\ell+\frac{1}{2}} g_J . \quad (43)$$

A case where this correction would appear is the ℓ=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 representation available that does support $\psi\chi$ broadening:

$$\begin{aligned}\sigma_n &= \sigma_p \\ &+\sum_\ell \sum_r \sigma_{mr}\Big\{\Big[\cos 2\phi_\ell - (1-\frac{\Gamma_{nr}}{\Gamma_r}) + \frac{G_{r\ell}}{\Gamma_{nr}}\Big]\,\psi(\theta,x) \\ &+(\sin 2\phi_\ell + \frac{H_{r\ell}}{\Gamma_{nr}})\,\chi(\theta,x)\Big\}\,, \end{aligned} \tag{44}$$

where

$$G_{r\ell} = \frac{1}{2}\sum_{\substack{r'\neq r\\ J_{r'}\neq J_r}} \Gamma_{nr}\Gamma_{nr'}\frac{\Gamma_r+\Gamma_{r'}}{(E_r-E_{r'})^2+(\Gamma_r+\Gamma_{r'})^2/4}\,, \tag{45}$$

and

$$H_{r\ell} = \sum_{\substack{r'\neq r\\ J_{r'}\neq J_r}} \Gamma_{nr}\Gamma_{nr'}\frac{E_r-Er'}{(E_r-E_{r'})^2+(\Gamma_r+\Gamma_{r'})^2/4}\,. \tag{46}$$

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 MLBW method is implemented in `csmlbw2`.

Adler-Adler Representation (Adler-Adler) The multilevel Adler-Adler representation (`csaa`) is defined for ℓ=0 only. It is useful for fissionable materials. The total cross sections are given by

$$\begin{aligned}\sigma_t(E) &= \frac{4\pi}{k^2}\sin^2\phi_0 \\ &+ \frac{\pi\sqrt{E}}{k^2}\Big\{\sum_r \frac{1}{\nu_r}\Big[(G_r\cos 2\phi_0 + H_r\sin 2\phi_0)\,\psi(\theta,x) \\ &+ (H_r\cos 2\phi_0 - G_r\sin 2\phi_0)\,\chi(\theta,x)\Big] \\ &+ A_1 + \frac{A_2}{E} + \frac{A_3}{E^2} + \frac{A_4}{E^3} + B_1E + B_2E^2\Big\}\,, \end{aligned} \tag{47}$$

where

$$x = \frac{\mu_r - E}{\nu_r} \ , \tag{48}$$

and where ν_r is the resonance half-width (corresponds to $\Gamma/2$ in the Breit-Wigner notation), μ_r is the resonance energy, G_r is the symmetric total parameter, H_r is the asymmetric total parameter, and the A_i and B_i are coefficients of the total background correction.

The fission and capture cross section both use the form

$$\begin{aligned} \sigma_x(E) &= \frac{\pi\sqrt{E}}{k^2}\Bigg\{\sum_r \frac{1}{\nu_r}\left[G_r\psi(\theta,x) + H_r\chi(\theta,x)\right] \\ &+ A_1 + \frac{A_2}{E} + \frac{A_3}{E^2} + \frac{A_4}{E^4} + B_1E + B_2E^2\Bigg\} \ , \end{aligned} \tag{49}$$

where the values of G, H, A_i, and B_i appropriate for the desired reaction are used.

Doppler-broadening can be applied as for the SLBW case, except note that Γ_r in Eq. 27 must be replaced with $2\nu_r$. Doppler-broadened Adler-Adler cross sections are more accurate than SLBW cross sections because the background is smoother. However, cross sections below about $16kT/A$ will still be inaccurate. The Adler-Adler method is not used in modern evaluations.

Reich-Moore Representation (RM) The Reich-Moore representation as implemented in subroutine `csrmat` is a multi-level formulation with two fission channels; hence, it is useful for both structural and fissionable materials. The cross sections are given by

$$\sigma_t = \frac{2\pi}{k^2}\sum_\ell\sum_J g_J\Big\{\left(1 - \mathrm{Re}\,U^{\ell J}_{nm}\right) + 2d_{\ell J}\left[1 - \cos(2\phi_\ell)\right]\Big\} \ , \tag{50}$$

$$\sigma_n = \frac{\pi}{k^2}\sum_\ell\sum_J g_J\Big\{|1 - U^{\ell J}_{nn}|^2 + 2d_{\ell J}\left[1 - \cos(2\phi_\ell)\right]\Big\} \ , \tag{51}$$

$$\sigma_f = \frac{4\pi}{k^2}\sum_\ell\sum_J g_J\sum_c |\mathcal{I}^{\ell J}_{nc}|^2 \ , \text{ and} \tag{52}$$

$$\sigma_\gamma = \sigma_t - \sigma_n - \sigma_f \ , \tag{53}$$

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

$$U_{nn}^{\ell J} = \mathrm{e}^{2i\phi_\ell}\left[2\mathcal{I}_{nn} - 1\right] . \tag{54}$$

The elements of the R-matrix are given by

$$R_{nc}^{\ell J} = \delta_{nc} - \frac{i}{2}\sum_r \frac{\Gamma_{nr}^{1/2}\Gamma_{cr}^{1/2}}{E_r - E - \frac{i}{2}\Gamma_{\gamma r}} . \tag{55}$$

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

$$||I - \ell| - \frac{1}{2}| \rightarrow I + \ell + \frac{1}{2} . \tag{56}$$

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. (43). 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 RM 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 current version of RECONR has such a capability, and it can be used with RM evaluations. Because of channel-spin issues, it works best with RML evaluations. See below for a discusion of angular distributions.

Hybrid R-Function Representation (HRF) The Hybrid R-Function representation treats elastic scattering as a multi-level cross section using formulas similar to those given above for the Reich-Moore format in the case where fission is absent. The other reactions are treated with formulas similar to those of the SLBW method. The main use for this format is to provide a better representation of competitive reactions than is provided by any of the other formats described above. This treatment can include a background R-function, tabulated charged-particle penetrabilities, and optical model phase shifts. Following the

RM notation, the elastic cross section is given by

$$\sigma_n = \frac{\pi}{k^2} \sum_{\ell} \sum_{s=|I-\frac{1}{2}|}^{I+\frac{1}{2}} \sum_{J=|l-s|}^{l+s} g_J |1 - U_{nn}^{\ell s J}|^2 \ , \tag{57}$$

where the U function is given by the scalar version of Eq. (54):

$$U_{nn}^{\ell s J} = \mathrm{e}^{2i\phi_\ell} \left[\frac{2}{R_{nn}^{\ell s J}} - 1 \right] . \tag{58}$$

The R-function itself is given by

$$R_{nn}^{\ell s J} = 1 - \frac{i}{2} \sum_{r} \frac{\Gamma_{nr}}{E_r - E - \frac{i}{2}\Gamma_{\gamma r}} - i\, P_{\ell s J} R_{\ell s J}^0 \ , \tag{59}$$

where $R_{\ell s J}^0$ is a (complex) background R function and $P_{\ell s J}$ is a penetrability factor. The background R function can either be read in or set to zero. The penetrability and shift factors are computed from the scattering radius or channel radius as for SLBW. The phase shifts $\phi_{\ell s J}$ can be computed from the scattering radius as before, or the (complex) phase shifts can be read in from an optical model calculation.

Note that resonance parameters are given explicitly for all three quantum numbers ℓ, s, and J. No correction to the potential scattering cross section from repeated J values is needed.

Elastic angular distributions can also be computed from HRF parameters if the LAD parameter is set; however, RECONR does not support that.

Reich-Moore-Limited Representaton (RML) The Reich-Moore-Limited representation is a more general multilevel 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. The RML processing in NJOY is based on the SAMMY code[36]. The calculation in RECONR makes use of several subroutines exported by the `samm` module; namely, `s2sammy`, `ppsammy`, `rdsammy`, `cssammy`, and `desammy`.

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^π 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^π 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

$$R_{cc'} = \sum_\lambda \frac{\gamma_{\lambda c}\gamma_{\lambda c'}}{E_\lambda - E - i\Gamma_{\lambda\gamma}/2} + R_c^b \delta_{cc'} \,, \tag{60}$$

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:

$$X_{cc'} = P_c^{1/2} \, L_c^{-1} \sum_{c''} Y_{cc''}^{-1} R_{c''c'} \, P_{c'}^{1/2} \,, \tag{61}$$

where

$$Y_{cc''} = L_c^{-1}\delta_{cc''} - R_{cc''} \,, \tag{62}$$

and

$$L_c = S_c - B_c + iP_c \,. \tag{63}$$

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

$$\sigma_{elastic} = \frac{4\pi}{k_\alpha^2} \sum_{J^\pi} \Big[\sin^2\phi_c (1 - 2X_{cc}^i) - X_{cc}^r \sin(2\phi_c) + \sum_{c'} |X_{cc'}|^2 \Big] \,,$$

where $X^r_{cc'}$ is the real part of $X_{cc'}$, $X^i_{cc'}$ is the imaginary part, ϕ_c is the phase shift, the sum over J^π 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

$$\sigma_{capture} = \frac{4\pi}{k^2_\alpha} \sum_{J^\pi} \sum_c g_{J\alpha} \sum_c \left[X^i_{cc} - \sum_{c'} |X_{cc'}|^2 \right], \tag{64}$$

where the sum over J^π 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

$$\sigma_{reaction} = \frac{4\pi}{k^2_\alpha} \sum_{J^\pi} g_{J\alpha} \sum_c \left[X^i_{cc} - \sum_{c'} |X_{cc'}|^2 \right], \tag{65}$$

where the sum over c is limited to channels in the spin group J^π 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 $\alpha \rightarrow \alpha'$. 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 as follows:

$$\rho = (2.196771 \times 10^{-3}) \frac{A}{A+1} \sqrt{|E + \frac{A+1}{A} Q|} \; . \tag{66}$$

These factors are a little more complicated for Coulomb channels. See the SAMMY reference for more details.

The RML representation is new to the ENDF format, and it isn't represented by any cases in ENDF/B-VII.0. There are experimental evaluations for ^{19}F and ^{35}Cl from ORNL available. However, the RML approach provides a very faithful representation of resonance physics, and it should see increasing use in the future.

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:

$$\frac{d\sigma_{\alpha\alpha'}}{d\Omega_{\rm CM}} = \sum_L B_{L\alpha\alpha'}(E) P_L(\cos\beta)\,, \tag{67}$$

where the subscript $\alpha\alpha'$ indicates the cross section as defined by the two 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 over the elements of the scattering matrix U, where

$$U_{cc'} = \Omega_c W_{cc'} \Omega_{c'}\,, \tag{68}$$

where

$$\Omega_c = \mathrm{e}^{i(w_c - \phi_c)}\,, \tag{69}$$

and

$$W = I + 2iX\,, \tag{70}$$

with I being the identity matrix and X having been already given in Eq. 61. The coefficients B become

$$\begin{aligned} B_{L\alpha\alpha'}(E) &= \frac{1}{4k_\alpha^2} \sum_A \sum_B \sum_C \sum_D \sum_E \sum_F \frac{1}{(2i+1)(2I+1)} \\ &\times G_{c_1c_1';c_2c_2';L}\, \mathrm{Re}[(\delta_{c1c_1'} - U_{c_1c_1'})(\delta_{c_2c_2'} - U^*_{c_2c_2'})]\,. \end{aligned} \tag{71}$$

The spins I and i are for the target and projectile for particle pair α. The complex expressions for the geometric coefficient G are given in the SAMMY documentation. The six summations are as follows:

- A sum over spin groups defined by J_1^π
- B sum over spin groups defined by J_2^π
- C sum over entrance channels c_1 belonging to group J_1^π with particle pair α
- D sum over exit channels c_1' belonging to group J_1^π with particle pair α'
- E sum over entrance channels c_2 belonging to group J_2^π with particle pair α
- F sum over exit channels c_2' belonging to group J_2^π with particle pair α'

Fig. 3 shows the first few Legendre coefficients for the elastic scattering cross sections as computed by NJOY from the experimental evaluation for ^{19}F.

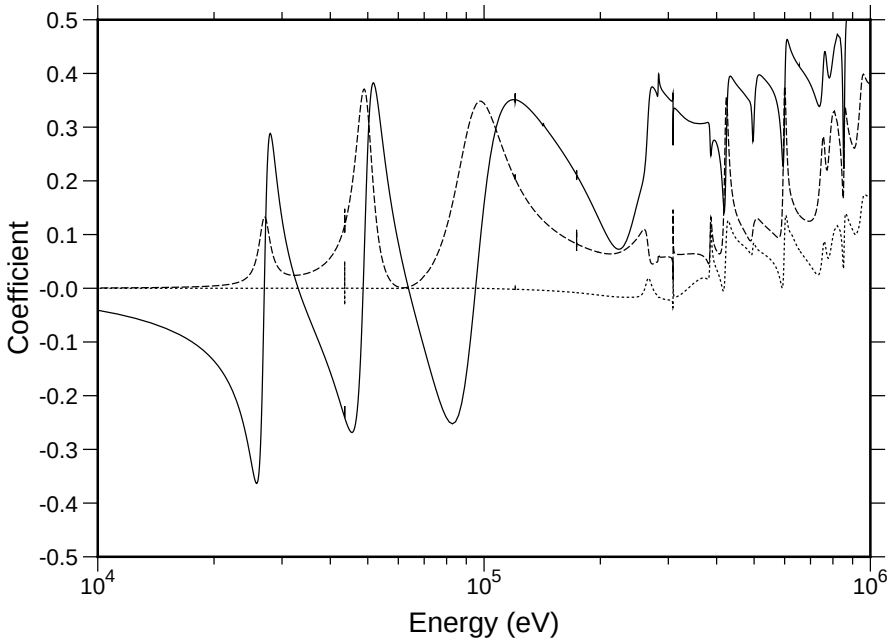

Figure 3: Legendre coefficients of the angular distribution for elastic scattering in ^{19}F using the RML resonance representation (P_1 solid, P_2 dashed, P_3 dotted).

Although first introduced in NJOY2012, computing resonance angular distributions is a new, little used feature, and so 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 NJOY2016 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 an RM evaluation. Change `Want_SAMRML_RM` to true.

Infinitely-Dilute Unresolved Range Parameters 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 SLBW approximation with interference.

$$\sigma_n(E) = \sigma_p + \frac{2\pi^2}{k^2}\sum_{\ell,J}\frac{g_J}{\overline{D}}\left[\overline{\Gamma}_n^2 R_n - 2\overline{\Gamma}_n \sin^2\phi_\ell\right] , \tag{72}$$

$$\sigma_x(E) = \frac{2\pi^2}{k^2}\sum_{\ell,J}\frac{g_J}{\overline{D}}\,\overline{\Gamma}_n\overline{\Gamma}_x R_x \text{ , and} \tag{73}$$

$$\sigma_p = \frac{4\pi}{k^2}\sum_{\ell}(2\ell+1)\sin^2\phi_\ell , \tag{74}$$

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,

$$\overline{\Gamma}_n\overline{\Gamma}_f R_i = \left\langle\frac{\Gamma_n\Gamma_f}{\Gamma}\right\rangle$$

$$= \int dx_n P_\mu(x_n)\int dx_f P_\nu(x_f)\int dx_c P_\lambda(x_c) \tag{75}$$

$$\times \frac{\Gamma_n(x_n)\,\Gamma_f(x_f)}{\Gamma_n(x_n)+\Gamma_f(x_f)+\Gamma_\gamma+\Gamma_c(x_c)} , \tag{76}$$

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[38] giving

$$R_f = \sum_i W_i^\mu \sum_j W_j^\nu \sum_k W_k^\lambda \frac{Q_i^\mu\, Q_j^\nu}{\overline{\Gamma}_n Q_i^\mu + \overline{\Gamma}_f Q_j^\nu + \Gamma_\gamma + \overline{\Gamma}_c Q_k^\lambda} . \tag{77}$$

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 Γ_n^0 (AMUN) is given in the file, and

$$\overline{\Gamma}_n = \Gamma_n^0\sqrt{E}\,V_\ell(E) , \tag{78}$$

where the penetrabilities for the unresolved region are defined as

$$V_0 = 1 , \tag{79}$$

$$V_1 = \frac{\rho^2}{1+\rho^2} , \text{ and} \tag{80}$$

$$V_2 = \frac{\rho^4}{\rho+3\rho^2+\rho^4} . \tag{81}$$

Other parameters are defined as for SLBW.

Unresolved resonance 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`.

3.5 Code Description

RECONR is implemented as a public subroutine `reconr` exported by the Fortran-90 module `reconm` defined by `reconr.f90`.

The first step is to read cards 1, 2, and 3 of the user's input. The TAPEID record of the input file (`nendf`) is read and printed, then the new TAPEID record is written to the output file (`npend`). RECONR is now ready to enter the loop over the desired materials.

For each material, space is allocated for the energy nodes (`enode`), and `ruin` is called to read cards 4 through 7 of the user's input. If the reconstruction temperature (`tempr`) is greater than zero, a table of ψ and χ functions is generated (the W table is used; see `wtab` and `quickw`). The `findf` utility routine from module `endf` is then used to find the first card of File 1 (MF=1, MT=451) for the desired material.

File 1 on the input ENDF file is examined to obtain certain constants and flags and to analyze the directory (`anlyzd`). Subroutine `anlyzd` determines which reactions should be considered "redundant"; that is, the reactions that are sums of other reactions and will be included on the output PENDF file. The total cross section (MT=1 for neutrons, MT=501 for photons) will always be included; the nonelastic cross section (MT=3) will be included if it is needed for photon production (that is, MF=12, MT=3 is found); the inelastic cross section (MT=4) will be included if sections with MT in the range 51 – 91 occur in the file, and the total fission reaction (MT=18) will be called redundant if the partial fission representation (MT=19, 20, 21, 38) is found. MT103 (n,p) can be a summation reaction if its partials MT600, MT601, ..., are present, and the same for the other charged-particle absorption reactions. Space for the new material directory is

then allocated (`mfs`, `mts`, `ncs`). Section identification and card counts will be entered into these arrays as they are determined.

File 2 on the ENDF file is now checked using `s2sammy` (which was imported from the `samm` module) to see whether the sammy method is needed. This depends on whether RML resonance parameters are found, and whether conversion to RML format has been requested for Reich-Moore or Breit-Wigner data (see `Want_SAMRML_RM` and `Want_SAMRML_BW`). The variable `nmtres` flags the use of SAMMY processing. Because RML evaluations can include more than the normal elastic, fission, and capture reactions, a list of reactions identified is printed. Here is an example from an experimental ^{19}F evaluation.

```
resonance range information
---------------------------------
ier     energy-range     lru lrf  method
 1  1.000E-05 1.000E+06   1   7   sammy

  samm resonance reactions:    2  102   51   52
  samm max legendre order:  7
  generating File 4 for resonance angular distributions
```

The next step is to read File 2, which contains resolved and unresolved resonance parameters (if any). The array `res` is allocated to contain the File 2 data and `rdfil2` is called to read them. This routine uses the additional routines `rdf2bw`, `rdf2aa`, `rdf2hy`, `rdsammy`, `rdf2u0`, `rdf2u1`, and `rdf2u2` to read the different types of resonance parameters. The subroutine `rdsammy` is imported from the `samm` module. In addition, another imported routine `ppsammy` is used to prepare for the SAMMY calculation. While the resonance parameters are being stored, RECONR adds each resonance energy to its list of energy nodes (`enode`). In the unresolved energy range, RECONR uses the energies of tabulated parameters or fission widths if available. If the evaluation uses energy-independent parameters, or if the energy steps between the nodes are too large (see `wide`), `rdfil2` creates additional node energies at a density of approximately 13 points per decade (see `egridu`). Note that regions where the unresolved representation for an element overlaps the resolved or smooth ranges are found and marked by negative energy values. The energy nodes are sorted into order and duplications are removed.

If the SAMMY method is active, and if angular distributions have been re-

quested (see Want_Angular_Dist), the maximum Legendre order defined by the resonance data is printed out.

If unresolved data are present, subroutine genunr is called to compute the infinitely-dilute unresolved average cross sections on the unresolved energy grid using csunr1 or csunr2. Any backgrounds on File 3 are included, except in regions of resolved-unresolved or unresolved-smooth overlap. The computed cross sections are arranged in the order required by the special section with MF=2 and MT=152, which is written onto the PENDF tape by recout. Using the normal ENDF style, this format is defined by the following:

```
[MAT,2,152/ZA,AWR,LSSF,0,0,INTUNR]HEAD
[MAT,2,152/0.,0.,5,1,NW,NUNR/
           E1,STOT1,SELAS1,SFIS1,SCAP1,STRN1,
           E2,STOT2,SELAS2,SFIS2,SCAP2,STRN2,
           ...
           ENUNR,STOTNUNR,....]LIST
```

where NW = 6 + 6*NUNR. The definitions of the energy and cross section entries are fairly obvious, except STRN stands for the current-weighted total cross section. This format is specialized to "infinite dilution." The more general form used for self-shielded effective cross sections will be described in the UNRESR, PURR, and GROUPR chapters of this manual.

The subroutine lunion is used to linearize and unionize the ENDF data. Space is reserved for two buffers to be used by loada/finda, for the linearization stack (x and y), and for the ENDF scratch area (scr). The length of the stack (ndim) determines the smallest possible subdivision of each panel (energy points as close as $2^{-\texttt{ndim}}$ times the panel width can be generated). Since the number of energies in the union grid may soon exceed the capacity of any reasonable memory array, the existing list of energy nodes is copied to binary scratch storage (the loada/finda system). This storage system consists of the buffers bold and bnew and the scratch units iold and inew. The energy grid points will "ping-pong" back and forth between units 14 and 15 as the union grid is built up. Subroutine lunion now starts with MT=2 and checks each reaction in sequence to determine whether the current grid (on iold) is sufficient to represent the reaction to within the desired tolerance using linear interpolation. If not, RECONR adds additional points by adaptively halving the intervals. The new grid is stored on inew. The units inew and iold are swapped, and the next

MT is processed. When all nonredundant reactions have been examined, the list of energies in `loada/finda` storage is the desired linearized and unionized grid. The storage used is deallocated.

This grid is used as the starting point for resonance reconstruction in `resxs`. Subroutine `resxs` first reserves space for the `loada/finda` buffers `bufr` and `bufg`, the linearization stack (`x` and `y`), and the partial cross sections (`sig`). The length of the stack (`ndim`) determines the smallest possible subdivision of a panel between two nodes (energy points as close a $2^{-\texttt{ndim}}$ times the panel width can be generated). Subroutine `resxs` then examines the grid on `ngrid` (`iold` from `lunion`) panel by panel. Grid points are added and cross sections computed until the convergence criteria discussed in Section 3.3 are satisfied. The cross sections are copied to `nout` using `loada`, and `resxs` continues to the next panel. This procedure is continued until all panels are converged. The result is a tape (`nout`) containing the energy grid in the resonance region and the total, elastic, fission, capture, and possibly additional cross sections at each energy point.

Unionization is obtained automatically in the resonance region since all of the partials are computed simultaneously in `sigma`, using `cssblw` for SLBW parameters, `csmlbw` for MLBW parameters, `csaa` for multi-level Adler-Adler parameters, `csrmat` for Reich-Moore parameters, `cshyb` for Hybrid R-Function parameters, `cssammy` for Reich-Moore-Limited parameters, and `sigunr` for unresolved resonance parameters. This last routine retrieves the cross sections from the table prepared by `genunr`. Subroutine `cssammy` is imported from the `samm` module. A special feature of RECONR is the ability to reconstruct the cross sections at `tempr` by $\psi\chi$ broadening if SLBW or Adler-Adler parameters are given. This can also be done for MLBW using the GH method implemented by `csmlbw2`. The Doppler-broadened resonance shapes are obtained using `quickw` (see description in the UNRESR chapter), and the linearization procedure proceeds as before.

The resonance cross sections on `ngrid` are merged with the ENDF cross sections in `emerge`. First, the background grid from `lunion` is merged with the resonance grid from `resxs` and written onto the `loada/finda` file, which will accumulate the total cross section and any other redundant reactions required (`iold/inew`). A loop is then set up over all nonredundant reactions. For each grid point, the ENDF background cross section is obtained by interpolation. If this grid point has a resonance contribution on `nres`, it is added. The resulting net cross section at this point is added into the appropriate redundant cross sections on `iold/inew` and also saved on `ngrid`. When all the energies for this reaction have been processed, the cross sections on `ngrid` are converted into a

TAB1 record and written on `nscr`. This loop is continued until all reactions have been processed. When `emerge` is finished, `nscr` contains cross sections for all the nonredundant reactions, and `iold` contains the redundant summation reactions.

Control now passes to `recout`, which writes the new File 1 comments and dictionary. It also writes a default version of the section with MF=2 and MT=151 that gives no resonance parameters. The upper limit of the resolved energy range, `eresh`, is added to the "C2" field of the third card so that BROADR knows not to broaden into the unresolved energy range. For materials with unresolved data, a specially formatted section (MF=2, MT=152) is written containing the infinitely-dilute unresolved cross sections. This section can be used by BROADR and GROUPR to correct for resolved-unresolved overlap effects, if necessary. Subroutine `recout` then steps through the reactions on `nscr` and `iold`. Redundant summation reactions are converted to TAB1 records and inserted in the correct order. Nonredundant reactions are simply copied. Finally, a MEND record is added and control is returned to `reconr`.

Now `reconr` either directs that this process be repeated for another isotope or writes a TEND record and terminates. The result is a new file in ENDF format containing the desired pointwise cross sections. Normally, only Files 1, 2, 3, 10, and 13 are included for neutron files. However, if angular distribution processing has been requested, a File 4 containing the Legendre coefficients will also be written. Because the original ENDF File 4 was not copied to the PENDF file, the presence of sections of File 4 on the PENDF file provides a flag to subsequent modules that resonance angular distributions have been calculated. Only Files 1 and 23 are included for a photon file.

The SAMMY method is implemented in a separate Fortran-90 module `samm` defined by `sammy.f90`. It exports the subroutines `cssammy` (computes cross sections, angular distributions, and derivatives), `s2sammy` (scans File 2 to see if SAMMY method is needed and measure some sizes), `ppsammy` (sets up SAMMY calculation), `rdsammy` (reads in File 2 data with optional conversion of BW or RM data to RML form), and `desammy` (cleans up after the SAMMY calculation). It also exports some logical parameters, namely, `Want_Partial_Derivs`, `Want_Angular_Dist`, `Want_SAMRML_RM` and `Want_SAMRML_BW`. See ERRORR for the use of derivatives. If conversion from BW and/or RM was requested, it is possible to get the resulting File 2 values printed out for checking. Just set `imf2` in `sammy.f90` to 1.

The `cssammy` subroutine uses `abpart` to compute some energy-independent pieces of the cross sections and derivatives. The main work for cross sections,

angular distributions, and derivatives in done in `cross`. The results for cross sections are returned in `sigp` to be consistent with the other "cs" routines in RECONR. Angular distributions are returned in `siga`, and sensitivities are returned in `sigd` (derivatives of cross sections with respect to parameters). Subroutine `cross` starts by initializing the quantities being calculated (cross sections, maybe angular distributions, maybe derivatives), and then it sets up a loop over the spin-parity groups. It initializes the results for this spin group and then calls `setr` to compute the elements of the R-matrix (see Eq. 60 and other quantities, such as the Y matrix, penetrabilities (`rootp`), and phase shifts. It then inverts the Y matrix and calculates the X matrix of Eq. 61. See `setxqx`. It can then use the X matrix to compute the contributions to the cross sections (`sectio`), maybe angular distributions (`setleg`), and maybe derivatives from this spin group and add them into the sum over groups. When the loop over spin groups is complete, it normalizes things properly and returns its results.

Going back to subroutine `setr`, it computes the R-matrix first. It then computes the phase shift, penetrabilities, and shift factors. For non-Coulomb cases, the phase shifts come from `sinsix`, and the penetrability P and boundry condition $(S - B + iP)^{-1}$ come from `pgh`. For Coulomb cases, subroutine `pghcou` computes all of these quantities. The penetrability P is converted to `rootp` for use in Eq. 61.

3.6 Input Instructions

The input instructions for each module are given in the code as comment cards at the beginning of the source code for each module. The RECONR instructions are reproduced here for the convenience of the reader.

```
!---input specifications (free format)---------------------------
!
! card 1
!    nendf    unit for endf tape
!    npend    unit for pendf tape
! card 2
!    tlabel   66 character label for new pendf tape
!             delimited with quotes, ended with /.
! card 3
!    mat      material to be reconstructed
!    ncards   number of cards of descriptive data for new mf1
!             (default=0)
```

```
!    ngrid    number of user energy grid points to be added.
!             (default=0)
! card 4
!    err      fractional reconstruction tolerance used when
!             resonance-integral error criterion (see errint)
!             is not satisfied.
!    tempr    reconstruction temperature (deg kelvin)
!             (default=0)
!    errmax   fractional reconstruction tolerance used when
!             resonance-integral error criterion is satisfied
!             (errmax.ge.err, default=10*err)
!    errint   maximum resonance-integral error (in barns)
!             per grid point (default=err/20000)
!             (note: the max cross section difference for
!             linearization, errlim, and for reconstruction,
!             errmin, are also tied to errint.  to get maximum
!             accuracy, set errint to a very small number.
!             for economical production, use the defaults.)
! card 5
!    cards    ncards of descriptive comments for mt451
!             each card delimited with quotes, ended with /.
! card 6
!    enode    users energy grid points
!
!     cards 3, 4, 5, 6 must be input for each material desired
!     mat=0/ terminates execution of reconr.
!
!--------------------------------------------------------------------
```

A sample input for processing an isotope from ENDF/B-VII follows (the line numbers are for reference only and are not part of the input). First, mount the ENDF/B-VII file for ^{235}U on unit 20.

```
1.  reconr
2.  20 21/
3.  'pendf tape for U-235 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 tape will be on unit 20, and that the output PENDF tape will be on unit 21. Card 3 is a "TAPEID" label for the output PENDF file. Card 4 gives the MAT number for U-235 and says that two additional comment cards will be given. Card 5 sets the reconstruction tolerance to 0.1% (.001 as a fraction) with all its other parameters defaulted. Cards 6 and 7 are the two comment cards to be inserted into the PENDF files MF1/MT451 section. Finally, the "0/" terminates the RECONR input. The capability to loop over multiple isotopes in RECONR is rarely used for neutron files, but it is useful for photon interaction processing (see GAMINR). The resulting PENDF tape will contain the desired TAPEID card, followed by ^{235}U data, a MEND card and a TEND card.

3.7 Error Messages

`error in reconr***illegal nsub for reconr`

RECONR only processes sublibraries that contain cross section data. Check whether the right input ENDF input tape was mounted.

`error in anlyzd***too many redundant reactions`

Increase the global parameter `nmtmax`=10.

`error in xxxxxx***storage in enode exceeded`

Too many energy nodes including the user's nodes and the energies from MF=2. Increase the global parameter `nodmax`=800000. This message can come from `rdf2bw`, `rdf2aa`, `rdf2hy`, `rdf2u0`, `rdf2u1`, or `rdf2u2`.

`error in xxxxxx***res storage exceeded`

Too much resonance data. This should not occur for a conforming ENDF-format file, because `maxres` is computed from the MF=2 line count in the MF=1/MT=451 index. This message can come from `rdfil2`, `rdf2bw`, `rdf2aa`, `rdf2hy`, `rdf2u0`, `rdf2u1`, or `rdf2u2`.

`error in xxxxxx***storage in eunr exceeded.`

Increase the global parameter `maxunr`=500. This message can come from `rdfil2`, `rdf2u1`, or `rdf2u2`.

`error in rdfil2***illegal resonance mode.`

A resonance mode has been requested that RECONR does not understand.

`error in rdf2bw***energy-dep scattering radius ...`

This option is only used in MLBW for current evaluations.

`message from rdf2bw***calc... of angular distribution not...`

This option is only partially available in RECONR. This message can come from `rdf2bw` (for Reich-Moore cases) or from `rdf2hy` (Hybrid R function).

`error in rdf2hy***hybrid competing reactions not yet added`

This option is not yet available in RECONR.

`error in lunion***ill behaved threshold`

The routine is having trouble adjusting the threshold to agree with the Q value. Check the points near the threshold for this evaluation.

`error in lunion***exceeded stack`

Increase length of linearization stack `ndim` (currently 50).

`error in resxs***stack exceeded`

Increase length of reconstruction stack `ndim` (currently 50).

`error in sigma***general r-matrix not installed.`

This option is not yet available in RECONR.

`error in sigma***illegal option.`

There is a problem with the ENDF tape.

`error in csmlbw***not coded for temperature gt 0 deg k`

The $\psi\chi$ Doppler-broadening doesn't work for normal MLBW. This message shouldn't occur, because temperatures greater than zero will cause `csmlbw2` to be called.

`error in csrmat***not coded for temperature gt 0 deg k`

The $\psi\chi$ Doppler-broadening doesn't work for RM. Use `tempr`=0. only.

`error in cshybr***doppler broad'g not provided for hybrid`

The $\psi\chi$ Doppler-broadening doesn't work for hybrid parameters. Use `tempr`=0. only.

`error in csaa***bad li value`

There is an error in the evaluation format.

`message from emerge--negative elastic cross sects found`

Negative elastic cross sections can occur for SLBW evaluations.

`error in recout***for mf -- mt --`

Indexing and pair count for this section do not make sense.

`calculation of angular distribution not installed -`

Message comes from several resonance types that do not support the calculation of angular distributions. Some of them can be used if `Want_SAMRL_RM` or `Want_SAMRML_BW` are true.

`message from s2sammy***multiple isotopes... -`

Multiple isotopes for RM sections don't work with the SAMMY method. The code automatically reverts to normal RM processing.

`error in s2sammy***res storage exceeded. -`
Storage limited to `maxres`. Shouldn't occur.

`error in s2sammy***energy-dep scattering length... -`
This only works for MLBW parameters.

`error in rearrange***nres fault -`
Trouble while rearranging resonances into spin-group order.

`errorr in findsp***quantum numbers in file 2 do not... -`
Problems with the quantum numbers for the evaluation.

`error in checkqn***error in quantum numbers -`
Problems with the quantum numbers for the evaluation.

`error in lmaxxx***lllmax limit to 51 -`
Problems with the Clebsch-Gordan coefficients for the angular distributions.

`error in clbsch***did not count correctly -`
Problems with the Clebsch-Gordan coefficients for the angular distributions.

`error in pspcou***llmax larger than 100 -`
Problem computing the Coulomb phase shifts.

`error in bigeta***I0 sum failed -`
Problem for the Coulomb routine.

`error in bigeta***K0 sum failed -`
Problem for the Coulomb routine.

`error in bigeta***L1 sum failed -`
Problem for the Coulomb routine.

`error in bigeta***K1 sum failed -`
Problem for the Coulomb routine.

`error in setleg***nppx too large -`
Problem generating Legendre polynomials.

3.8 Input-Output Units

The following logical units are used:

10 `nscr1` in `reconr`, `nout` in `lunion`, and `nin` in `emerge`. Contains copy of nonredundant sections from original ENDF tape.

11 `nscr2` in `reconr`; `ngrid` in `lunion`, `resxs`, and `emerge`. Contains union grid for ENDF tape (not counting resonances).

12 `nscr3` in `reconr`, `nout` in `resxs`, and `nres` in `emerge`. Contains resonance grid and cross sections.

13	`nscr4` in `reconr` is used for two separate purposes. In `resxs` it is a binary scratch file `nscr` used for the unthinned resonance data. In `emerge` and `recout`, it is `nmerge` and contains the nonredundant reactions on the union grid.
14/15	`iold/inew` in `lunion`. Are used locally only to accumulate union grid for ENDF cross sections. Destroy after use.
14/15	`iold/inew` in `emerge`. Are used locally only to accumulate summation cross sections on union grid.
20-99	User's choice for ENDF (`nendf`) and PENDF (`npend`) tape numbers to link RECONR with other NJOY modules.
5,6,7	See the NJOY chapter for a description of the I/O units.

Note that 11, 12, 14, and 15 are always binary. Unit 10 has the same mode as `nendf`. Unit 13 is binary when used in RESXS, and it has the same mode as `npend` elsewhere. `npend` can have a different mode than `nendf`.

3.9 Storage Allocation

Storage allocation in RECONR is sensitive to (1) the amount of resonance parameter data, (2) the number of energy-grid node values, (3) the size of the resonance reconstruction stack, (4) the use of $\psi\chi$ broadening, and (5) the sizes of the `loada/finda` buffers. Other storage requirements are minor.

Buffer sizes can be reduced or increased at will. The result is a storage/speed tradeoff with no change in capability or accuracy. See the global parameters `nbufg`=2000, `nbufr`=2000, and `nbuf`=2000 at the beginning of the `reconm` module.

The $\psi\chi$ broadening option requires 7688 words of additional storage. Therefore, memory use can be reduced if $\psi\chi$ is not required. No code changes are needed — just avoid `tempr` greater than zero.

Resonance reconstruction in `resxs` uses 5×`ndim` words. The parameter `ndim` determines the smallest subdivision of a panel that can be obtained. Using `ndim`=30 allows points to be generated with spacing as small as one-billionth of the panel size (2^{30}).

The code currently allows for `nodmax`=800000 energy nodes and `maxunr`=500 unresolved points. These values tend to increase as additional very detailed resonance evaluations appear, but the current values seem to be sufficient for the evaluations existing as of this writing.