5 UNRESR

The UNRESR module is used to produce effective self-shielded cross sections for resonance reactions in the unresolved energy range. In ENDF-format evaluations, the unresolved range begins at an energy where it is difficult to measure individual resonances and extends to an energy where the effects of fluctuations in the resonance cross sections become unimportant for practical calculations. As described in the ENDF format manual,[9] resonance information for this energy range is given as average values for resonance widths and spacings together with distribution functions for the widths and spacings. This representation can be converted into effective cross sections suitable for codes that use the background cross section method, often called the Bondarenko method,[39] using a method originally developed for the MC2 code[37] and extended for the ETOX code[28]. This unresolved-resonance method has the following features:

- Flux-weighted cross sections are produced for the total, elastic, fission, and capture cross sections, including competition with inelastic scattering.
- A current-weighted total cross section is produced for calculating the effective self-shielded transport cross section.
- The energy grid used is consistent with the grid used by RECONR.
- The computed effective cross sections are written on the PENDF tape in a specially defined section (MF2, MT152) for use by other modules.
- The accurate quadrature scheme from the MC2-2 code[38] is used for computing averages over the ENDF statistical distribution functions.

This chapter describes the UNRESR module in NJOY2016.0.

5.1 Theory

In the unresolved energy range, it is not possible to define precise values for the cross sections of the resonance reactions $\sigma_x(E)$, where x stands for the reaction type (total, elastic, fission, or capture). It is only possible to define average values. Of course, these average values should try to preserve the reaction rate:

$$\overline{\sigma}_{0x}(E^*) = \frac{\displaystyle\int_{E_1}^{E_2} \sigma_x(E)\,\phi_0(E)\,dE}{\displaystyle\int_{E_1}^{E_2} \phi_0(E)\,dE} \,, \tag{100}$$

where $\phi_0(E)$ is the scalar flux, E^* is an effective energy in the range $[E_1, E_2]$, and the range $[E_1, E_2]$ is large enough to hold many resonances but small with

respect to slowly varying functions of E. In order to calculate effective values for the transport cross section, it is necessary to compute the current-weighted total cross section also. It is given by

$$\overline{\sigma}_{1t}(E^*) = \frac{\displaystyle\int_{E_1}^{E_2} \sigma_x(E)\,\phi_1(E)\,dE}{\displaystyle\int_{E_1}^{E_2} \phi_1(E)\,dE} \ , \tag{101}$$

where the P_1 component of the neutron flux, $\phi_1(E)$, is proportional to the neutron current. To proceed farther, it is necessary to choose a model for the shape of $\phi_\ell(E)$ in the vicinity of E^*. The model used in UNRESR is based on the B_0 approximation for large homogeneous systems and narrow resonances:

$$\phi_\ell(E) = \frac{C(E)}{[\,\Sigma_t(E)\,]^\ell} \ , \tag{102}$$

where $C(E)$ is a slowly varying function of E, and $\Sigma_t(E)$ is the macroscopic total cross section for the system. In order to use this result in Eq. 100, it is further assumed that the effects of other isotopes in the mixture can be approximated by a constant called σ_0 in the range $[E_1, E_2]$, or

$$\phi_\ell(E) = \frac{C(E)}{[\,\sigma_0 + \sigma_t(E)\,]^\ell} \ . \tag{103}$$

Therefore, the effective cross sections in the unresolved range are represented by

$$\overline{\sigma}_{0x}(E^*) = \frac{\displaystyle\int_{E_1}^{E_2} \frac{\sigma_x(E)}{\sigma_0 + \sigma_t(E)} C(E)\,dE}{\displaystyle\int_{E_1}^{E_2} \frac{1}{\sigma_0 + \sigma_t(E)} C(E)\,dE} \ , \tag{104}$$

with x being t for total, e for elastic, f for fission, and γ for capture, and

$$\overline{\sigma}_{1t}(E^*) = \frac{\displaystyle\int_{E_1}^{E_2} \frac{\sigma_x(E)}{[\,\sigma_0 + \sigma_t(E)\,]^2} C(E)\,dE}{\displaystyle\int_{E_1}^{E_2} \frac{1}{[\,\sigma_0 + \sigma_t(E)\,]^2} C(E)\,dE} \ . \tag{105}$$

This equation can also be written in the equivalent form

$$\overline{\sigma}_{1t}(E^*) = \frac{\displaystyle\int_{E_1}^{E_2} \frac{1}{\sigma_0 + \sigma_t(E)} C(E)\, dE}{\displaystyle\int_{E_1}^{E_2} \frac{1}{[\,\sigma_0 + \sigma_t(E)\,]^2} C(E)\, dE} - \sigma_0 \ . \tag{106}$$

The parameter σ_0 in Eq. 103 deserves more discussion. It can be looked at as a parameter that controls the depth of resonance dips in the flux. When σ_0 is large with respect to the peak cross sections of resonances in $\sigma_t(E)$, the shape of the flux is essentially $C(E)$. For smaller values of σ_0, dips will develop in the flux that correspond to peaks in σ_t. These dips will cancel out part of the reaction rate in the region of the peaks, thus leading to self-shielding of the cross section. Analysis shows that it is possible to use this single parameter to represent the effects of admixed materials or the effects of neutron escape from an absorbing region. See the GROUPR chapter of this manual for additional details.

The cross sections that appear in the above integrals can be written as the sum of a resonant part and a smooth part as follows:

$$\sigma_x(E) = b_x + \sigma_{Rx}(E) = b_x + \sum_s \sum_r \sigma_{xsr}(E - E_{sr}) \ , \tag{107}$$

where s is an index to a spin sequence, r is an index to a particular resonance in that spin sequence, and E_{sr} is the center energy for that resonance. The smooth part b_x can come from a smooth background given in the ENDF file, and it also includes the potential scattering cross section σ_p for the elastic and total cross sections (x=t and x=e). In terms of the smooth and resonant parts, the effective cross sections become

$$\overline{\sigma}_{0x}(E^*) = b_x + \frac{\displaystyle\int_{E_1}^{E_2} \frac{\sigma_{Rx}(E)}{\overline{\sigma} + \sigma_{Rt}(E)} C(E)\, dE}{\displaystyle\int_{E_1}^{E_2} \frac{1}{\overline{\sigma} + \sigma_{Rt}(E)} C(E)\, dE} \ , \tag{108}$$

and

$$\overline{\sigma}_{1t}(E^*) = \frac{\displaystyle\int_{E_1}^{E_2} \frac{1}{\overline{\sigma} + \sigma_{Rt}(E)} C(E)\, dE}{\displaystyle\int_{E_1}^{E_2} \frac{1}{[\,\overline{\sigma} + \sigma_{Rt}(E)\,]^2} C(E)\, dE} - \sigma_0 \ , \tag{109}$$

where $\overline{\sigma} = b_t + \sigma_0$. It is convenient to transform the denominators of Eqs. 108 and 109 into

$$\int \frac{1}{\overline{\sigma}+\sigma_t} C\,dE = \frac{1}{\overline{\sigma}} \left\{ \int C\,dE - \int \frac{\sigma_t}{\overline{\sigma}+\sigma_t} C\,dE \right\} \ , \tag{110}$$

and

$$\int \frac{1}{[\overline{\sigma}+\sigma_t]^2} C\,dE = \frac{1}{\overline{\sigma}^2} \left\{ \int C\,dE - \int \frac{\sigma_t}{\overline{\sigma}+\sigma_t} C\,dE - \int \frac{\overline{\sigma}\sigma_t}{[\overline{\sigma}+\sigma_t]^2} C\,dE \right\} \ . \tag{111}$$

Furthermore, since $C(E)$ is assumed to be a slowly-varying function of E, it can be pulled out through all integrals and dropped. The average cross sections become

$$\overline{\sigma}_{0x} = b_x + \frac{\overline{\sigma} I_{0x}}{1 - I_{0t}} \ , \tag{112}$$

and

$$\overline{\sigma}_{1t} = b_t + \frac{\overline{\sigma} I_{1t}}{1 - I_{0t} - I_{1t}} \ . \tag{113}$$

The last equation can also be written in the form

$$\overline{\sigma}_{1t} = \overline{\sigma} \left[\frac{1 - I_{0t}}{1 - I_{0t} - I_{1t}} \right] - \sigma_0 \ . \tag{114}$$

The average cross sections are thereby seen to depend on two types of "fluctuation integrals:"

$$I_{0x} = \frac{1}{E_2 - E_1} \int_{E_1}^{E_2} \frac{\sigma_{Rx}(E)}{\overline{\sigma} + \sigma_{Rt}(E)}\,dE \ , \tag{115}$$

and

$$I_{1t} = \frac{1}{E_2 - E_1} \int_{E_1}^{E_2} \frac{\overline{\sigma}\sigma_{Rt}(E)}{[\,\overline{\sigma} + \sigma_{Rt}(E)\,]^2}\,dE \ , \tag{116}$$

where x can take on the values t, n, f, or γ. Note that $I_{1t} \leq I_{0t}$, the difference increasing as σ_0 decreases from infinity.

Inserting the actual sums over resonances into the formula for I_{0x} gives

$$I_{0x} = \frac{1}{E_2 - E_1} \int_{E_1}^{E_2} \frac{\sum_{sr} \sigma_{xsr}(E - E_{sr})}{\overline{\sigma} + \sum_{sr} \sigma_{tsr}(E - E_{sr})}\,dE \ . \tag{117}$$

If the resonances were widely separated, only the "self" term would be important, and one would obtain

$$I_{0x} = \sum_{sr} \frac{1}{E_2 - E_1} \int_{E_1}^{E_2} \frac{\sigma_{xsr}(E - E_{sr})}{\overline{\sigma} + \sigma_{tsr}(E - E_{sr})} \, dE \ . \tag{118}$$

Since the range of integration is large with respect to the width of any one resonance, the variable of integration can be changed to ξ=E−E_{sr}, and the limits on ξ can be extended to infinity. For any one sequence, the interval E_2−E_1 is equal to the average spacing of resonances in that sequence times the number of resonances in the interval. Therefore,

$$I_{0x}^{I} = \sum_{s} \frac{1}{D_s} \frac{1}{N_s} \sum_{r} \int_{-\infty}^{\infty} \frac{\sigma_{xsr}(\xi)}{\overline{\sigma} + \sigma_{tsr}(\xi)} \, d\xi \tag{119}$$

where D_s is the average spacing, and the "I" superscript indicates that this is the "isolated resonance" result. Because there are assumed to be many resonances in the interval, the sum over resonances can be changed to a multiple integration over some characteristic set of parameters (such as widths) times the probability of finding a resonance with some particular values of the parameters:

$$\frac{1}{N} \sum_{r \in s} f_r = <f>_s = \int d\alpha P_s(\alpha) \int d\beta P_s(\beta) \cdots f(\alpha, \beta, \cdots) \ . \tag{120}$$

In the following text, this multiple integral (up to four fold) will be abbreviated by writing the α integral only. The final results for isolated resonances are as follows:

$$I_{0x}^{I} = \sum_{s} \frac{1}{D_s} \int P(\alpha) \int_{-\infty}^{\infty} \frac{\sigma_{xs\alpha}(\xi)}{\overline{\sigma} + \sigma_{ts\alpha}(\xi)} \, d\xi \, d\alpha \ , \tag{121}$$

and

$$I_{1t}^{I} = \sum_{s} \frac{1}{D_s} \int P(\alpha) \int_{-\infty}^{\infty} \frac{\overline{\sigma}\sigma_{ts\alpha}(\xi)}{[\, \overline{\sigma} + \sigma_{ts\alpha}(\xi) \,]^2} \, d\xi \, d\alpha \ . \tag{122}$$

If the effects of overlap are too large to be neglected, overlap corrections to the isolated resonance result can be constructed using the continued-fraction generator

$$\frac{1}{a+b} = \frac{1}{a} \left(1 - \frac{b}{a+b} \right) \ . \tag{123}$$

Starting with the I_0 integrals,

$$\frac{\sum_{sr}\sigma_{xsr}}{\overline{\sigma}+\sum_{sr}\sigma_{tsr}} = \sum_{sr}\frac{\sigma_{xsr}}{\overline{\sigma}+\sigma_{tsr}}\Big\{1 - \sum_{r'\neq r}\frac{\sigma_{tsr'}}{\overline{\sigma}+\sum\sigma_{tsr}} - \sum_{s'\neq s}\sum_{r'}\frac{\sigma_{ts'r'}}{\overline{\sigma}+\sum\sigma_{tsr}}\Big\}. \quad (124)$$

Expand the second term in the braces to get

$$\begin{aligned}\frac{\sum_{sr}\sigma_{xsr}}{\overline{\sigma}+\sum_{sr}\sigma_{tsr}} = \sum_{sr}\frac{\sigma_{xsr}}{\overline{\sigma}+\sigma_{tsr}}\Big\{1 &- \sum_{r'\neq r}\frac{\sigma_{tsr'}}{\overline{\sigma}+\sigma_{tsr}+\sigma_{tsr'}} \\ &\Big\{1-\sum_{\substack{r''\neq r\\ r''\neq r'}}\frac{\sigma_{tsr''}}{\overline{\sigma}+\sum\sigma_{tsr}} - \sum_{s'\neq s}\sum_{r'}\frac{\sigma_{ts'r'}}{\overline{\sigma}+\sum\sigma_{tsr}}\Big\} \\ &- \sum_{s'\neq s}\sum_{r'}\frac{\sigma_{ts'r'}}{\overline{\sigma}+\sum\sigma_{tsr}}\Big\}. \end{aligned} \quad (125)$$

Neglecting the products of three *different* resonances in sequence s gives

$$\begin{aligned}\frac{\sum_{sr}\sigma_{xsr}}{\overline{\sigma}+\sum_{sr}\sigma_{tsr}} &= \sum_{sr}\frac{\sigma_{xsr}}{\overline{\sigma}+\sigma_{tsr}} \\ &\times \Big\{1-\sum_{r'\neq r}\frac{\sigma_{tsr'}}{\overline{\sigma}+\sigma_{tsr}+\sigma_{tsr'}}\Big\} \\ &\times \Big[1-\sum_{s'\neq s}\sum_{r'}\frac{\sigma_{ts'r'}}{\overline{\sigma}+\sum\sigma_{tsr}}\Big]. \end{aligned} \quad (126)$$

The factor before the opening brace is the isolated resonance result, the factor in braces is the in-sequence overlap correction, and the factor in brackets is the sequence-sequence overlap correction. Note that recursion can be used to refine the sequence-sequence correction to any desired accuracy. Similarly, the I_1 integral requires

$$\frac{\sum_{sr}\overline{\sigma}\sigma_{xsr}}{[\,\overline{\sigma}+\sum_{sr}\sigma_{tsr}\,]^2} = \sum_{sr}\frac{\overline{\sigma}\sigma_{xsr}}{[\,\overline{\sigma}+\sigma_{tsr}\,]^2}\Big[1 - \sum_{r'\neq r}\frac{\sigma_{tsr'}}{\overline{\sigma}+\sum\sigma_{tsr}} - \sum_{s'\neq s}\sum_{r'}\frac{\sigma_{ts'r'}}{\overline{\sigma}+\sum\sigma_{tsr}}\Big]^2. \quad (127)$$

Once more, we expand the fraction and neglect terms that will result in products of three or more different resonances in the same sequence. The result is

$$\begin{aligned}\frac{\sum_{sr}\overline{\sigma}\sigma_{xsr}}{[\,\overline{\sigma}+\sum_{sr}\sigma_{tsr}\,]^2} &= \sum_{sr}\frac{\overline{\sigma}\sigma_{xsr}}{[\,\overline{\sigma}+\sigma_{tsr}\,]^2} \\ &\times \left\{1-2\sum_{r'\neq r}\frac{\sigma_{tsr'}}{\overline{\sigma}+\sigma_{tsr}+\sigma_{tsr'}}+\sum_{r'\neq r}\left(\frac{\sigma_{tsr'}}{\overline{\sigma}+\sigma_{tsr}+\sigma_{tsr'}}\right)^2\right\} \\ &\times \left[1-\sum_{s'\neq s}\sum_{r'}\frac{\sigma_{ts'r'}}{\overline{\sigma}+\sum\sigma_{tsr}}\right], \end{aligned} \tag{128}$$

where in-sequence and sequence-sequence overlap terms have been factored out.

The next step is to substitute these results back into the fluctuation integrals I_0 and I_1. The integrals over energy and the sums over different resonances in each sequence can be handled as described above for isolated resonances. This procedure will result in three different kinds of integrals. The first kind includes the isolated resonance integrals already considered above

$$\begin{aligned} B_{xs} &= \frac{1}{E_2-E_1}\int_{E_1}^{E_2}\sum_r\frac{\sigma_{xsr}}{\overline{\sigma}+\sigma_{tsr}}\,dE \\ &= \frac{1}{D_s}\int P(\alpha)\int_{-\infty}^{\infty}\frac{\sigma_{xs\alpha}(\xi)}{\overline{\sigma}+\sigma_{ts\alpha(\xi)}}\,d\xi\,d\alpha\,, \end{aligned} \tag{129}$$

and

$$\begin{aligned} D_{ts} &= \frac{1}{E_2-E_1}\int_{E_1}^{E_2}\sum_r\frac{\overline{\sigma}\sigma_{tsr}}{[\,\overline{\sigma}+\sigma_{tsr}\,]^2}\,dE \\ &= \frac{1}{D_s}\int P(\alpha)\int_{-\infty}^{\infty}\frac{\overline{\sigma}\sigma_{xs\alpha}(\xi)}{[\,\overline{\sigma}+\sigma_{ts\alpha}(\xi)\,]^2}\,d\xi\,d\alpha\,. \end{aligned} \tag{130}$$

Note that $D_t \leq B_t$, the difference increasing as σ_0 decreases from infinity.

The next kind are the in-sequence overlap integrals. The sum over r' is replaced by integrals over the probabilities of finding each partial width and the probability of finding a resonance r' at a distance η from resonance r.

$$
\begin{aligned}
V_{0xs} &= \frac{1}{E_2-E_1}\int_{E_1}^{E_2}\sum_r\sum_{r'\neq r}\frac{\sigma_{xsr}}{\overline{\sigma}+\sigma_{tsr}}\frac{\sigma_{tsr'}}{\overline{\sigma}+\sigma_{tsr}+\sigma_{tsr'}}\,dE \\
&= \frac{1}{D_s^2}\int P(\alpha)\int P(\beta)\int\int\Omega(\eta)\,\frac{\sigma_{xs\alpha}(\xi)}{\overline{\sigma}+\sigma_{ts\alpha}(\xi)} \\
&\quad \frac{\sigma_{ts\beta}(\xi-\eta)}{\overline{\sigma}+\sigma_{ts\alpha}(\xi)+\sigma_{ts\beta}(\xi-\eta)}\,d\eta\,d\xi\,d\beta\,d\alpha\ , \qquad (131)
\end{aligned}
$$

where $\xi = E - E_{sr}$ and $\eta = E_{sr'} - E_{sr}$. Similarly,

$$
\begin{aligned}
V_{1ts} &= \frac{1}{E_2-E_1}\int_{E_1}^{E_2}\sum_r\sum_{r'\neq r}\frac{\overline{\sigma}\sigma_{tsr}}{[\,\overline{\sigma}+\sigma_{tsr}\,]^2}\Big\{2\frac{\sigma_{tsr'}}{\overline{\sigma}+\sigma_{tsr}+\sigma_{tsr'}} \\
&\quad -\Big(\frac{\sigma_{tsr'}}{\overline{\sigma}+\sigma_{tsr}+\sigma_{tsr'}}\Big)^2\Big\}\,dE \\
&= \frac{1}{D_s^2}\int P(\alpha)\int P(\beta)\int\int\Omega(\eta)\,\frac{\overline{\sigma}\sigma_{ts\alpha}(\xi)}{[\,\overline{\sigma}+\sigma_{ts\alpha}(\xi)\,]^2} \\
&\quad \Big\{2\frac{\sigma_{ts\beta}(\xi-\eta)}{\overline{\sigma}+\sigma_{ts\alpha}(\xi)+\sigma_{ts\beta}(\xi-\eta)} \\
&\quad -\Big[\frac{\sigma_{ts\beta}(\xi-\eta)}{\overline{\sigma}+\sigma_{ts\alpha}(\xi)+\sigma_{ts\beta}(\xi-\eta)}\Big]^2\Big\}\,d\eta\,d\xi\,d\beta\,d\alpha\ . \qquad (132)
\end{aligned}
$$

The final class of integrals includes the sequence-sequence overlap corrections. They are simplified by noting that the positions of resonances in different spin sequences are uncorrelated. Therefore, $\Omega(\eta)$=1, and the integral of the product reduces to the product of the integrals.

Using the results and definitions from above, the fluctuation integrals become

$$
I_{0x} = \sum_s A_{xs}\ , \qquad (133)
$$

$$
A_{xs} = (B_{xs} - V_{0xs})\Big[\,1 - \sum_{s'\neq s} A_{ts'}\,\Big]\ , \qquad (134)
$$

and

$$
I_{1t} = \sum_s (D_{ts} - V_{1ts})\Big[\,1 - \sum_{s'\neq s} A_{ts'}\Big]^2\ , \qquad (135)
$$

where Eq. 134 provides a recursive definition of the A_{ts} for the sequence-sequence corrections as well as the normal value of A_{xs}.

These equations are formally exact for the sequence-sequence overlaps, but in-sequence overlaps only include the interactions between pairs of resonances. Three different approximations to this result are currently in use.

The MC2/ETOX Approximation The MC2 and ETOX codes use similar approximations to the results above, except that MC2 does not include a calculation of the current-weighted total cross section. Both codes explicitly neglect the in-sequence overlap corrections. This approximation was based on the assumption that resonance repulsion would reduce the overlap between resonances in a particular spin sequence, leaving the accidental close spacing of resonances in different sequences as the dominant overlap effect. In addition, both codes stop the recursion of Eq. 134 at $A_t = B_t$. Thus,

$$I_{0x} = \sum_s B_{xs}\Big(1 - \sum_{s' \neq s} B_{ts'}\Big) \ , \tag{136}$$

and

$$I_{1t} = \sum_s D_{ts}\Big(1 - \sum_{s' \neq s} B_{ts'}\Big)^2 \ . \tag{137}$$

The equations for the effective cross sections in the MC2/ETOX approximation become

$$\overline{\sigma}_{0x} = b_x + \frac{\overline{\sigma}\sum_s B_{xs}\Big(1 - \sum_{s' \neq s} B_{ts'}\Big)}{1 - \sum_s B_{ts}\Big(1 - \sum_{s' \neq s} B_{ts'}\Big)} \ , \tag{138}$$

and

$$\overline{\sigma}_{1t} = b_t + \frac{\overline{\sigma}\sum_s D_{ts}\Big(1 - \sum_{s' \neq s} B_{ts'}\Big)^2}{1 - \sum_s B_{ts}\Big(1 - \sum_{s' \neq s} B_{ts'}\Big) - \sum_s D_{ts}\big(1 - \sum_{s' \neq s} B_{ts'}\big)^2} \ , \tag{139}$$

or

$$\overline{\sigma}_{1t} = \overline{\sigma}\left[\frac{1-\sum_s B_{ts}\Big(1-\sum_{s'\neq s} B_{ts'}\Big)}{1-\sum_s B_{ts}\Big(1-\sum_{s'\neq s} B_{ts'}\Big)-\sum_s D_{ts}\Big(1-\sum_{s'\neq s} B_{ts'}\Big)^2}\right]-\sigma_0\ . \quad (140)$$

These are the equations that are used in the UNRESR module of NJOY. Note that the equation in the ETOX code and report corresponding to Eq. 140 is incorrect. The following equation was used in the ETOX code:

$$\overline{\sigma}_{1t} = \overline{\sigma}\left[\frac{1-\sum_s B_{ts}\Big(1-\sum_{s'\neq s} B_{ts'}\Big)}{1-\sum_s C_{ts}\Big(1-\sum_{s'\neq s} C_{ts'}\Big)}\right]-\sigma_0\ , \quad (141)$$

with $C_{ts} = B_{ts} + D_{ts}$.

The MC2-2 Approximation The MC2-2 code includes the in-sequence overlap corrections, which the authors found to be more important than previously thought. It uses additional approximations to obtain the equivalent of

$$\overline{\sigma}_{0x} = b_x + \overline{\sigma}\sum_s \frac{B_{xs}-V_{0xs}}{1-B_{ts}+V_{0ts}}\ . \quad (142)$$

The additional approximations used are

1. Set $A_{ts} = B_{ts} - V_{0ts}$ (first-order sequence-sequence overlap),
2. Neglect the factor $(1-\sum_{s'\neq s} A_{ts'})$ in the denominator, and
3. Use the approximation $1-\sum_i f_i \approx \prod_i (1-f_i)$ on the numerator and denominator.

These simplifications result in a loss of accuracy for the sequence-sequence overlap correction at relatively low values of σ_0. The $\overline{\sigma}_{1t}$ term is not calculated.

The UXSR Approximation The experimental UXSR module was developed at Oak Ridge (with some contributions from LANL) based on coding from the Argonne National Laboratory (ANL) in an attempt to include the sophisticated in-sequence overlap corrections from MC2-2 without approximating the sequence-sequence corrections so badly. It also implemented a calculation of the current-weighted total cross section, which was omitted in MC2-2. The additional cost of using the full expressions for Eqs. 133 and 135 is minimal, and

effective cross sections can be computed for lower values of σ_0 when in-sequence overlap is small (*e.g.,* ^{238}U).

Now that expressions have been chosen for computing the cross sections in terms of the isolated-resonance integrals, it is necessary to select an efficient numerical method for computing them. The resonant parts of the cross sections are given by

$$\sigma_{xsr}(E-E_{sr}) = \left[\sigma_m \frac{\Gamma_x}{\Gamma}\psi(\theta,X)\right]_{sr} , \tag{143}$$

and

$$\sigma_{tsr}(E-E_{sr}) = [\,\sigma_m\{\cos 2\phi_\ell\,\psi(\theta,X) + \sin 2\phi_\ell\,\chi(\theta,X)\}\,]_{sr} , \tag{144}$$

where x takes on the values γ, f, or c for capture, fission, or competition, and

$$\sigma_m = \frac{4\pi g_J}{k^2}\frac{\Gamma_n}{\Gamma} , \tag{145}$$

$$\theta = \sqrt{\frac{A}{4kTE_0}}\,\Gamma , \tag{146}$$

$$X = \frac{2(E-E_0)}{\Gamma} , \tag{147}$$

$$g_J = \frac{2J+1}{2(2I+1)} , \text{ and} \tag{148}$$

$$k = 2.196771\times 10^{-3}\frac{A}{1+A}\sqrt{E} . \tag{149}$$

The functions ψ and χ are the symmetric and antisymmetric components of the broadened resonance line shape:

$$\psi(\theta,X) = \frac{\theta\sqrt{\pi}}{2}\mathrm{Re}W\left(\frac{\theta X}{2},\frac{\theta}{2}\right) , \tag{150}$$

and

$$\chi(\theta,X) = \theta\sqrt{\pi}\mathrm{Im}W\left(\frac{\theta X}{2},\frac{\theta}{2}\right) , \tag{151}$$

where

$$W(x,y) = \exp[-(x+iy)^2]\,\mathrm{erfc}[-i(x+iy)] \tag{152}$$

is the complex probability integral. The methods for computing ψ and χ are well known (see `quikw`).

The first integral needed is

$$\begin{aligned} B_{xs} &= \frac{1}{D_s}\int P(\alpha)\int \frac{\sigma_{xs\alpha}(\xi)}{\overline{\sigma}+\sigma_{ts\alpha}(\xi)}\,d\xi\,d\alpha \\ &= \frac{1}{D_s}\int P(\alpha)\int \frac{\sigma_m(\Gamma_x/\Gamma)\psi(\theta,X)}{\overline{\sigma}+\sigma_m\{\cos 2\phi_\ell\,\psi(\theta,X)+\sin 2\phi_\ell\,\chi(\theta,X)\}}\,d\xi\,d\alpha \\ &= \frac{1}{D_s}\int P(\alpha)\frac{\Gamma_x}{2\cos 2\phi_\ell}\int \frac{\psi(\theta,X)}{\beta+\psi(\theta,X)+\tan 2\phi_\ell\,\chi(\theta,X)}\,dX\,d\alpha \end{aligned} \quad (153)$$

where

$$\beta = \frac{\overline{\sigma}}{\sigma_m\cos 2\phi_\ell}\ . \quad (154)$$

The second integral needed is

$$\begin{aligned} B_{ts} &= \frac{1}{D_s}\int P(\alpha)\int \frac{\sigma_{ts\alpha}(\xi)}{\overline{\sigma}+\sigma_{ts\alpha}\xi)}\,d\xi\,d\alpha \\ &= \frac{1}{D_s}\int P(\alpha)\,\frac{\Gamma}{2}\int \frac{\psi(\theta,X)+\tan 2\phi_\ell\,\chi(\theta,X)}{\beta+\psi(\theta,X)+\tan 2\phi_\ell\,\chi(\theta,X)}\,dX\,d\alpha\ . \end{aligned} \quad (155)$$

Both of these integrals can be expressed in terms of the basic J integral:

$$\begin{aligned} B_{xs} &= \frac{1}{D_s}\int P(\alpha)\frac{\Gamma}{\cos 2\phi_\ell}\,J(\beta,\theta,\tan 2\phi_\ell,0)\,d\alpha\ ,\ \text{and} \\ B_{ts} &= \frac{1}{D_s}\int P(\alpha)\,\Gamma\,J(\beta,\theta,\tan 2\phi_\ell,\tan 2\phi_\ell)\,d\alpha\ , \end{aligned} \quad (156)$$

where

$$J(\beta,\theta,a,b) = \frac{1}{2}\int_{-\infty}^{\infty}\frac{\psi(\theta,X)+b\,\chi(\theta,X)}{\beta+\psi(\theta,X)+a\,\chi(\theta,X)}\,dX\ . \quad (157)$$

The D integral can be handled in the same way, but only total reaction is required.

$$
\begin{aligned}
D_{ts} &= \frac{1}{D_s}\int P(\alpha)\int \frac{\overline{\sigma}\sigma_{ts\alpha}(\xi)}{[\,\overline{\sigma}+\sigma_{ts\alpha}(\xi)\,]^2}\,d\xi\,d\alpha \\
&= \frac{1}{D_s}\int P(\alpha)\,\frac{\Gamma}{2}\int \frac{\beta\psi(\theta,X)+\tan 2\phi_\ell\,\chi(\theta,X)}{[\,\beta+\psi(\theta,X)+\tan 2\phi_\ell\,\chi(\theta,X)\,]^2}\,dX\,d\alpha \\
&= \frac{1}{D_s}\int P(\alpha)\,\Gamma\,K(\beta,\theta,\tan 2\phi_\ell,\tan 2\phi_\ell)\,d\alpha\ , \qquad (158)
\end{aligned}
$$

where

$$
K(\beta,\theta,a,b)=\frac{1}{2}\int_{-\infty}^{\infty}\frac{\beta\,[\,\psi(\theta,X)+b\,\chi(\theta,X)\,]}{[\,\beta+\psi(\theta,X)+a\,\chi(\theta,X)\,]^2}\,dX\ . \qquad (159)
$$

A method for computing J, including the interference effects, has been developed by Hwang for MC2-2[38]. However, this method was not available in the days when MC2 and ETOX were developed. Therefore, UNRESR uses only $J(\beta,\theta,0,0)$ and $K(\beta,\theta,0,0)$ in computing the isolated-resonance fluctuation integrals. A direct integration is used over most of the X range, but the part of the integral arising from large X is handled using analytic integrations of the asymptotic forms of the arguments (see `ajku`).

The final step is to do the n-fold integration over the probability distributions for the resonance widths. This integration has been abbreviated as a single integration over α in the above equations. The method used was originally developed for MC2-2 and is based on Gauss-Jacobi quadratures. A set of 10 quadrature points and weights is provided for each of the χ^2 probability distributions with 1 through 4 degrees of freedom. These quadratures convert the n-fold integral into an n-fold summation. The value of n can be as large as 4 when Γ_n, Γ_f, Γ_γ, and Γ_c (competitive width) are all present.

Although UNRESR neglects the effects of overlap between resonances in the same spin sequence and the effects of interference in the elastic and total cross sections, it still gives reasonable results for the background cross section values needed for most practical problems. Modern evaluations are steadily reducing the need for accurate unresolved calculations by extending the resolved resonance range to higher and higher energies. Ultimately, UNRESR should be upgraded to use the UXSR approach. Another alternative is to generate self-shielded effective cross sections from ladders of resonances chosen statistically (see PURR). This avoids many of the approximtions in the overlap corrections.

In NJOY2016, running the PURR module after UNRESR overwrites the UNRESR output with the PURR results. In fact, UNRESR can be omitted

from the processing stream. To use UNRESR results, either omit PURR from the processing stream or run it before running UNRESR.

5.2 Implementation

In implementing this theory in UNRESR, there are special considerations involving the choice of an incident energy grid, what to do if the unresolved range overlaps the resolved range or the range of smooth cross sections, the choice of the σ_0 grid, how to interpolate on σ_0, and how to communicate the results to other modules.

Choice of Energy Grid. The same logic is used to choose the incident energy grid in UNRESR and RECONR. It is complicated, because of the several different representations available for unresolved data, and because of the existence of evaluations that have been carried over from previous versions of ENDF/B or ENDF/B-VII evaluations with inadequate energy grids. Even many modern evaluations have inadequate energy grids.

For evaluations that give energy-independent unresolved-resonance parameters, there is still an energy dependence to the cross sections. Because this dependence is normally somewhere between constant and a $1/v$ law, a fairly coarse grid with about 13 points per decade should be sufficient to allow the cross sections to be computed reliably using linear-linear interpolation.

If the evaluation uses energy-dependent parameters, the normal rule would be to use the energies that were provided by the evaluator and to obtain intermediate cross sections by interpolation. Unfortunately, some of the evaluations carried over from earlier days contain some energy intervals that are quite large (for example, steps by factors of 10). The evaluators for these isotopes assumed that the user would use parameter interpolation and compute the cross sections at a number of intermediate energies in these long steps. Even some newer evaluations contain large jumps in the energy grid. UNRESR will detect such evaluations and add additional energy points in the large energy steps using an algorithm similar to the one used for the cases with energy-independent parameters. For NJOY2016, large jumps in the energy grid are any with step ratios greater than `wide`, which is currently set to 1.26.

The final energy grid can be observed by scanning the printed output from UNRESR.

Resolved-Unresolved Overlap. Elemental evaluations include separate energy ranges in MF2/MT151 for each of the isotopes of the element, and these energy ranges do not have to be the same for each isotope. This means that the lower end of an unresolved range may overlap the resolved range from another isotope, and the upper end of the unresolved range for an isotope can overlap the smooth range of another isotope. These overlap regions are detected by UNRESR as the resonance data are read in, and they are marked by making the sign of the incident energy value negative.

Choosing a σ_0 Grid. There are two factors to consider, namely, choosing values that will represent the shape adequately, and limiting the range of σ_0 to the region where the theory is valid. The $\sigma_x(\sigma_0)$ curves start out decreasing from the infinite dilution value as $1/\sigma_0$ as σ_0 decreases from infinity (1×10^{10} in the code). The curve eventually goes through an inflection point at some characteristic value of σ_0, becomes concave upward, and approaches a limiting value at small σ_0 that is smaller than the infinite-dilution value. Decade steps are often used, but the user should try to select values that include the inflection point and not waste values on the $1/\sigma_0$ region. Half-decade values are useful near the inflection point (*e.g.,* 100, 300, 1000 for ^{235}U). The grid interval chosen should be consistent with the interpolation method used (see below).

Choosing the lower limit for σ_0 is a more difficult problem. As shown in the theory section (5.1), resonance overlap effects are developed as a series in $1/\sigma_0$, and the series is truncated after only one step of recursion in Eq. 134. This means that the overlap correction should be most accurate for large σ_0 and gradually get worse as σ_0 decreases. Experience shows that the correction can actually get large enough to produce negative cross sections for small σ_0. (This problem can also show up as a failure in interpolation when a log scheme has been selected.) For isotopes that have relatively narrow resonances spaced relatively widely, such as ^{238}U, UNRESR gives reasonable results to σ_0 values as low as 0.1. For materials with stronger overlap, such as ^{235}U, a lower limit around 100 is more reasonable. A few of the heavy actinide evaluations have been seen to break down for σ_0 values lower than 1000. This problem is not too serious in practice. The fertile materials, which appear in large concentrations in reactors, allow the necessary small values of σ_0. The fissile materials have to be more dilute, and the larger σ_0 limit needed for them is not usually a problem.

The UXSR approximation discussed above allows one to reach somewhat smaller σ_0 values.

Interpolating on σ_0. It turns out that these functions are difficult to interpolate because they have a limited radius of convergence. Although approximate schemes have been developed based on using functions of similar shape such as the tanh function[40], better results can be obtained by using different interpolation schemes for the low- and high-σ_0 ranges. The TRANSX-CTR code[12] used interpolation in $1/\sigma_0$ for high σ_0, Lagrangian interpolation of $\ln\sigma_x$ *vs* $\ln\sigma_0$, for intermediate values, and a σ_0^2 extrapolation for very low σ_0. Unfortunately, schemes like this sometimes give ridiculous interpolation excursions when the polynomials are not suitable. Therefore, TRANSX-2[41] has had to revert to using simple linear interpolation, which is always bounded and predictable, but which requires relatively fine σ_0 grids.

Communicating Results to Other Modules. NJOY tries to use ENDF-like files for all communications between the different calculational modules. Because the unresolved effective cross sections were originally derived from the resonance parameters in File 2, it seemed reasonable to create a new section in File 2 to carry the unresolved cross sections onto other modules, and a special MT number of 152 was selected for this purpose. RECONR creates an MT152 that contains only the infinitely-dilute unresolved cross sections. UNRESR overwrites it with self-shielded unresolved cross sections. GROUPR can then use these cross sections in its calculation of the multigroup constants. The format used for MT152 is given below using the conventional ENDF style.

```
[MAT,2,152/ ZA, AWR, LSSF, 0, 0, INTUNR ] HEAD
[MAT,2,152/ TEMZ, 0, NREAC, NSIGZ, NW, NUNR/
            SIGZ(1), SIGZ(2),...,SIGZ(NSIGZ),
            EUNR(1),
            SIGU(1,1,1), SIGU(1,2,1),...,SIGU(1,NSIGZ,1),
            SIGU(2,1,1),...
            ...
            SIGU(NREAC,1,1),...,SIGU(NREAC,NSIGZ,1),
            EUNR(2),...
              <continue for energies through EUNR(NUNR)>
            ...SIGU(NREAC,NSIGZ,NUNR) ] LIST
```

where NREAC is always 5 (for the total, elastic, fission, capture, and current-weighted total reactions, in that order), NSIGZ is the number of σ_0 values, NUNR is the number of unresolved energy grid points, and NW is given by

```
NW=NSIGZ+NUNR*(1+5*NSIGZ)
```

5.3 User Input

The following summary of the user input instructions was copied from the comment cards at the beginning of the UNRESR module in the NJOY2016 source file.

```
!---input specifications (free format)--------------------------
!
! card 1
!   nendf   unit for endf tape
!   nin     unit for input pendf tape
!   nout    unit for output pendf tape
! card 2
!   matd    material to be processed
!   ntemp   no. of temperatures (default=1)
!   nsigz   no. of sigma zeroes (default=1)
!   iprint  print option (0=min, 1=max) (default=0)
! card 3
!   temp    temperatures in Kelvin (including zero)
! card 4
!   sigz    sigma zero values (including infinity)
!       cards 2, 3, 4 must be input for each material desired
!       matd=0/ terminates execution of unresr.
!
!-----------------------------------------------------------------------
```

Card 1, as usual, specifies the input and output units for the module. The input PENDF file on `nin` must have been processed through RECONR and BROADR. It will contain default versions of the special unresolved section with MF=2 and MT=152 that gives the infinitely-dilute unresolved cross sections for each temperature. The output PENDF file `nout` will contain revised sections MF2, MT152 that give effective cross sections *vs* σ_0 for all temperatures.

Card 2 is used to specify the material desired (`matd`), the number of temperatures and background cross sections desired (`ntemp` and `nsigz`), and the print

option (`iprint`). The actual temperature and σ_0 values are given on Cards 3 and 4. Temperatures should be chosen to be adequate for the planned applications. The temperature dependence of the effective cross sections is usually monotonic and fairly smooth. Polynomial interpolation schemes using T are often used, and roughly uniform spacing for the temperature grid (or spacing that increases modestly as T increases) is usually suitable.

The choice of a grid for σ_0 is more difficult. The curves of cross section versus σ_0 have an inflection point, and it is important to choose the grid to represent the inflection point fairly well. A log spacing for σ_0 is recommended. Very small values of σ_0 should not be used. These considerations are discussed in more detail in the "Implementation" section (5.2) above. Unfortunately, the best choice for a grid can only be found by trial and error.

5.4 Output Example

The following portion of UNRESR output is for ^{238}U from ENDF/B-VII.0.

```
unresr...calculation of unresolved resonance cross sections          494.4s

                                                    storage   8/   20000

unit for input endf/b tape ...........        -21
unit for input pendf tape ............        -23
unit for output pendf tape ...........        -24

temperatures .........................  2.936E+02
                                        4.000E+02
                                         ...
sigma zero values ....................  1.000E+10
                                        1.000E+03
                                        1.000E+02
                                        5.000E+01
                                        2.000E+01
                                        1.000E+01
                                        5.000E+00
                                        2.000E+00
                                        1.000E+00
                                        5.000E-01
print option (0 min., 1 max.) ........          1
```

```
mat = 9237    temp =  2.936E+02                                          494.4s
energy =  2.0000E+04
  1.433E+01  1.428E+01  ... 1.311E+01  1.297E+01 1.291E+01  1.288E+01
  1.380E+01  1.375E+01  ... 1.264E+01  1.252E+01 1.247E+01  1.244E+01
  0.000E+00  0.000E+00  ... 0.000E+00  0.000E+00 0.000E+00  0.000E+00
  5.294E-01  5.280E-01  ... 4.654E-01  4.540E-01 4.491E-01  4.464E-01
  1.433E+01  1.424E+01  ... 1.246E+01  1.230E+01 1.223E+01  1.220E+01
energy =  2.3000E+04
  1.414E+01  1.411E+01  ... 1.307E+01  1.293E+01 1.288E+01  1.285E+01
  1.364E+01  1.361E+01  ... 1.262E+01  1.250E+01 1.245E+01  1.242E+01
  0.000E+00  0.000E+00  ... 0.000E+00  0.000E+00 0.000E+00  0.000E+00
  4.962E-01  4.951E-01  ... 4.426E-01  4.325E-01 4.281E-01  4.257E-01
  1.414E+01  1.407E+01  ... 1.246E+01  1.229E+01 1.223E+01  1.219E+01
  ...
energy =  1.4903E+05
  1.140E+01  1.140E+01  ... 1.124E+01  1.118E+01 1.115E+01  1.113E+01
  1.126E+01  1.126E+01  ... 1.110E+01  1.104E+01 1.101E+01  1.099E+01
  0.000E+00  0.000E+00  ... 0.000E+00  0.000E+00 0.000E+00  0.000E+00
  1.427E-01  1.427E-01  ... 1.406E-01  1.386E-01 1.374E-01  1.367E-01
  1.213E+01  1.212E+01  ... 1.174E+01  1.156E+01 1.147E+01  1.141E+01
generated cross sections at  18 points                                  494.9s
```

The display of the cross section table for each energy has σ_0 reading across (in decreasing order) and reaction type reading down (in the order of total, elastic, fission, capture, and current-weighted total). Four σ_0 values were removed from the table to make it narrower for this report.

5.5 Coding Details

The main entry point for UNRESR is subroutine `unresr`, which is exported by module `unresm`. The subroutine starts by reading in the user's input and output unit numbers and opening the files that will be needed during the UNRESR run. After printing the introductory timer line, storage is allocated for an array `scr`, which will be used for reading in ENDF records. The default size of this array is `maxscr`=1000, which has proved sufficient for all evaluations tried so far. UNRESR now prints out the user's unit numbers on the output listing and calls `uwtab` to prepare internal tables used by `uw` to compute the real and imaginary parts of the complex probability integral.

The next step is to read in the `tapeid` records of the ENDF and PENDF

tapes. The loop over materials starts at statement number 110 by reading in the user's values for the ENDF MAT number, number of temperatures `ntemp`, number of σ_0 values `nsigz`, and print flag `iprint`. If this is not the end of the material loop (`mat`=0), the actual values of the temperatures and background cross sections are read into the arrays `temp` and `sigz`. The input quantities are echoed to the output listing to help detect input errors.

The code then begins a loop over the requested temperatures. It writes the current values of material ID, temperature, and time on the output listing. It then reads the resonance parameters from the section with MT=151 in File 2 of the ENDF tape using `rdunf2`. The arrays `eunr` with length `maxeunr`=150 and `arry` with length with length `maxarry`=10000 are used to store these data. Next, it reads the background cross sections from File 3 for each of the resonance reactions using `rdunf3`. Here, `sb` is used to store the data. The array `b` is allocated with sufficient length to build the output record to be written in MT=152 on the new PENDF file.

The next loop is over all the energy grid points at this temperature. The grid points were determined in `rdunf2`, and the `nunr` points are stored in the array `eunr`. The background cross sections are stored in an array `sb(ie,ix)`. The energy index takes on `nunr` values, and the reaction index `ix` takes on four values. The calculation of the actual effective cross sections takes place in `unresl`. The results for each energy appear in the array `sigu(ix,is)`, where `is` takes on `nsigz` values. Each `unresl` array is stored into the accumulating output array `b` and printed on the output listing.

At this point, UNRESR checks to see if there is a previous version of MT152 on the PENDF tape. If so, these new data will replace it. If not, a new section with MT=152 will be created. In either case, a new section MT451 in File 1 is generated containing the current temperature and the correct entry in the directory for the PENDF tape. Finally, the new MT152 for this temperature is copied onto the output file from the `b` array, and the rest of the contents of this temperature on the old PENDF file are copied to the new PENDF file. After writing a report on the number of resonance points generated and the amount of computer time used, UNRESR branches back to continue the temperature loop. When the last temperature has been processed, the code closes the files, writes a final report on the output listing, and terminates.

Note that UNRESR takes special branches for materials with no resonance parameters or materials with no unresolved parameters. Therefore, the user can freely request an UNRESR run even when there are no unresolved resonance

data present on the ENDF tape. UNRESR simply copies `nin` to `nout` in this case.

The subroutine `rdunf2` is used to read in the unresolved resonance parameters from File 2 of the input ENDF tape. It is very similar to the corresponding coding in RECONR. The array `scr` is used to read in the ENDF record, the resonance parameter data are stored in the array `arry`, and the final grid of energy values is stored in `eunr`. Note that `rdunf2` will add extra energy nodes for evaluations with energy-independent parameters or for energy-dependent evaluations that have energy steps larger than the factor `wide`, which is currently set to 1.26. The subroutine also discovers resolved-unresolved or unresolved-smooth overlap, flags those energy values, and prints messages to the user on the output listing.

Subroutine `ilist` is used to insert a new energy into a list of energies in ascending order. It is used to manage the accumulation of the grid of energy nodes.

Subroutine `rdunf3` is used to read in the background cross sections in the unresolved range from File 3. The unresolved energy grid determined by `rdunf2` is used for the background cross sections.

The main calculation of the effective cross sections for the unresolved range is performed in subroutine `unresl`. The calculation is done in two passes: first, the potential scattering cross section is computed; second, the unresolved cross sections are computed. The passes are controlled by the parameter `ispot`. In both cases, resonance parameter data stored in `arry` by `rdunf2` are used. The coding is similar to that used in RECONR down to the point where the ETOX statistical averages are computed. The three loops over `kf`, `kn`, and `kl` carry out the n-fold quadrature represented as integrals over α in the text. They account for variations in the fission width, neutron width, and competitive width. The capture width is assumed to be constant. Note that `ajku` is called in the innermost loop to compute the J and K integrals in `xj` and `xk`, respectively. The K integral returned by this routine is actually $J + K$ in our notation. The final quantities are computed in the loops over `itp` and `is0`. Note that `tk` is equivalent to $B_{ts} + D_{ts}$ in our notation. Similarly, `tl` is equivalent to B_{xs}, and `tj` is equivalent to B_{ts}. The last step in this subroutine is to compute the average cross sections by summing over spin sequences. The loop over `ks` computes `xj` as $\sum_s B_{ts}$ and `xk` as $\sum_s C_{ts}$. With these quantities available, it is easy to finish the calculation of the effective cross sections as given by Eq. 138 and Eq. 140.

The subroutines `uunfac`, `intrf`, and `intr` are similar to corresponding routines from RECONR and don't require further discussion here. Subroutine `ajku`

is used to compute the J and K integrals without interference corrections. The subroutines `quikw`, `uwtab`, and `uw` implement a package for computing the complex probability integral efficiently. It was originally developed at ANL for the MC2 code. When it is used, a pair of 62×62 tables for the real and imaginary parts of the complex probability integral are precomputed using `uwtab` and `uw`. Values of W for small arguments are obtained by interpolating in these precomputed tables. Values of W for larger arguments are obtained using various asymptotic formulas.

5.6 Error Messages

`error in unresr***mode conversion between nin and nout`

Input and output files must both be in ASCII mode (positive unit numbers), or they must both be in binary mode (negative unit numbers).

`error in unresr***storage exceeded`

There is not enough room in the `b` array. Increase `nb`, which currently is 5000.

`error in rdunf2***energy dependent data undefined`

When using unresolved resonance formalisms with energy dependent parameters (e.g. the fission width, etc.), these data need to be defined over the entire unresolved resonance region. In rare cases, such as ENDF/B-VII.0 Pu239, this is not the case, leading to NaN cross section values. This is an evaluation error.

`error in rdunf2***storage exceeded`

There is not enough room in `arry`. Increase the value of `maxarry`, which currently is 10000.

`error in unresl***storage exceeded`

The code is currently limited to 50 spin sequences. For more spin sequences, it will be necessary to increase the dimensions of several arrays in this subroutine.