LA-UR- 08-1999

Approved for public release; distribution is unlimited.

Title:	The Processing of ENDF70 and ENDF70PROT: New ACE-Formatted Neutron and Proton Libraries Based on ENDF/B-VII.0
Author(s):	Holly R. Trellue Robert C. Little
Intended for:	memo

Los Alamos
NATIONAL LABORATORY
EST. 1943

Form 836 (7/06)

Los Alamos
NATIONAL LABORATORY
EST. 1943

memorandum

Nuclear, Atomic, and EOS Data, X-1-NAD

To/MS: Distribution
Thru/MS: M. Chadwick, X-1, MS F699
From/MS: H. Trellue, X-1/D-5, MS K575
R. Little, X-1, MS F663
M. B. Lee, X-1, MS F663
Phone: 5-9539 / 5-3487
Symbol: X-1: HT-08-25
Date: March 21, 2008
Subject: ENDF70 Processing

New ACE-Formatted Neutron and Proton Libraries Based on ENDF/B-VII.0

1.0 Introduction

The ENDF/B-VII.0 nuclear data evaluations contain neutron cross sections for 390 isotopes and 3 elements.[1] All evaluations with the exception of ^{7}Be, which was incomplete, have been processed into ACE format at five different temperatures: 293.6, 600, 900, 1200, and 2500 K in the ENDF70 library. With the exception of ^{1}H, the processing was done using NJOY99[2] Version 248 and several pre- and post-processing and quality assurance checking codes. The ENDF70 neutron library was broken into eleven separate files of less than 900 MB each and will be compressed and released with MCNP5 1.50. The major changes to ENDF70 not present in ENDF66 were more temperatures, inclusion of substantially more nuclides including metastable states, and a new XSDIR file with atomic weights for the new nuclides.

Forty-seven ENDF/B-VII proton evaluations were also processed at room temperature using the ACER module of NJOY version 248. Forty-eight proton evaluations were available through ENDF/B-VII.0; ^{13}C could not be processed with NJOY because the evaluation was incomplete and was thus omitted from the resulting library. The resulting ACE-formatted data were run through the same checking codes as the neutron evaluations and then combined into the library ENDF70PROT, which is about 81 MB. For the moment, both of these libraries are available in the directory /usr/projects/data/nuclear/mc/new on the open ICN systems.

2.0 Background

Nuclear data is distributed as part of MCNP; thus it is used extensively worldwide. The release of the ENDF66 neutron library based upon data from ENDF/B-VI.6 involved many quality assurance tests to make sure the best data possible were released. Checking codes were written and each evaluation was plotted and tested in MCNP to make sure no major problems existed. The same rigorous procedure was employed for release of the ENDF70 library based on ENDF/B-VII.0 data for both neutron and proton evaluations. Note that during the processing of the ENDF66 library, some limitations of NJOY99 Version 50 were found that related to unresolved resonance tables (the PURR module) and delayed neutron data.[3,4] The problems in PURR contributed to the calculation of incorrect heating cross sections and factors in some probability tables and the need for more than 64 ladders for some isotopes. The delayed neutron data was simply not yet added by NJOY to the ACE file. All of these issues have been resolved in version 248 of NJOY, which is what was used for most of ENDF70 processing. The processing of ^{1}H was done using simply Update 271 of NJOY to fix an error in deuteron production.

3.0 The Neutron Continuous Energy Library ENDF70

3.1 New Features

There are numerous factors that were considered when processing the ENDF70 library, including providing more temperature-dependent data, inclusion of metastable state isomers, and the addition of new isotopes that were not previously processed to the list of atomic weights in the XSDIR file. The next several sections discuss each of these issues.

3.1.1 Temperature-Dependence

The temperature-dependent cross sections in ENDF70 have the identifiers of .70c, .71c, .72c, .73c, and .74c respectively (see Table 1).

Table 1. Mapping of ZAID Identifiers to Temperatures

Identifier	Temperature (K)
*.70c	293.6
*.71c	600
*.72c	900
*.73c	1200
*.74c	2500

The purpose of providing more intermediate temperatures than previously released in data libraries is to make reactor-type calculations easier both directly and with the use of the Doppler code[5] released with MCNP. Given pre-existing ACE-formatted libraries, Doppler prepares cross sections for any temperature greater than a temperature in an existing library. However, Doppler interpolates values for the thermal scattering and unresolved-range probability tables and can do so more accurately with an "upper" and "lower" bound of the temperature range. Having multiple temperatures available aids users in obtaining more accurate results at temperatures of interest. The total size of the new type1 (i.e. uncompressed) neutron data is approximately 8.8 GB. Compressed, it comprises about 380 MB. The data is broken into eleven different files distinguished by atomic number (Z). Table 2 lists which elements/isotopes are contained in each data file.

The light elements are contained in ENDF70A, uranium, neptunium and plutonium in ENDF70J, and other actinides in ENDF70K. Intermediate-mass structural materials, fission products, and heavy non-actinides are contained in ENDF70B through ENDF70I. The wide range of fission products now available should help users avoid the need to lump fission products together.

3.1.2. Inclusion of Metastable States

Another major change to the ACE-formatted libraries is the inclusion of nuclides with metastable states and even both ground and metastable states of some isotopes. For most nuclides, the ZAID is simply the atomic number (Z) followed by the atomic mass number (A) in the format ZZAAA. If the Z number is only one digit, the first Z can be blank, but if the atomic mass number is not three digits, then zeros must be included. However, to emphasize the presence of metastable states, we have tried to form ZAID identifiers for nuclides with metastable states that involve unrealistic atomic mass numbers.

Table 2. List of Contents in Each ENDF70 file

file	Elements	Z numbers
endf70a	H, He, Li, Be, B, C, N, O, F, Na, Mg, Al, Si, P, S, Cl, Ar, K, Ca, Sc, Ti, V	1 through 23
endf70b	Cr, Mn, Fe, Co, Ni, Cu, Zn, Ga, Ge	24 through 32
endf70c	As, Se, Br, Kr, Rb, Sr, Y, Zr, Nb, Mo	33 through 42
endf70d	Tc, Ru, Rh, Pd, Ag, Cd	43 through 48
endf70e	In, Sn, Sb, Te, I	49 through 53
endf70f	Xe, Cs, Ba, La, Ce, Pr	54 through 59
endf70g	Nd, Pm, Sm, Eu	60 through 63
endf70h	Gd, Tb, Dy, Ho, Er	64 through 68
endf70i	Lu, Hf, Ta, W, Re, Ir, Au, Hg, Pb, Bi	71 through 83
endf70j	U, Np, Pu	92 through 94
endf70k	Ra, Ac, Th, Pa, Am, Cm, Bk, Cf, Es, Fm	88 through 91 95 through 100

Table 3 contains a list of nuclides for which metastable state data are available as well as their ZAID identifiers. Note that in some cases, data for both the ground and metastable states exist; in others, only metastable state data exists, but the ZAID is formed consistently either way. If one is trying to calculate the atomic mass from the ZAID of a nuclide in a metastable state, a different formula must be used (see Equation 1) or the incorrect answer will result.

Table 3. List of Nuclides with Metastable States for ENDF70

	ZAID
^{58}Co (ground)	27058
^{58m}Co (metastable)	27458
^{110m}Ag (metastable only)	47510
^{115m}Cd (metastable only)	48515
^{127m}Te (metastable only)	52527
^{129m}Te (metastable only)	52529
^{148}Pm (ground)	61148
^{148m}Pm (metastable)	61548
^{166m}Ho (metastable only)	67566
^{242}Am (ground)	95642
^{242m}Am (metastable)	95242
^{244}Am (ground)	95244
^{244m}Am (metastable)	95644

$ZAID_m = (ZAID + 300) + (m*100)$ [1]

Where: $ZAID_m$ is the metastable isomer ZAID

ZAID is the ground-state ZAID

m is the excited state corresponding to m=0,1,2,3…

The exception to this ZAID rule is ^{242m}Am, for which the identifier for the metastable state in the past has always been 95242. The ground state is more stable but produced from capture in ^{241}Am a minority of the time; the majority of the production goes to the metastable state. Now that cross sections exist for both the ground and metastable states, an unusual identifier for the ground state has been developed: 95642.

3.1.3 Changes to XSDIR For New Isotopes

To utilize the cross sections in MCNP, atomic weights of applicable isotopes must be available. The file "xsdir" contains information about the cross section files, including size, but also includes atomic weights of all available isotopes. Because cross sections for new isotopes are now available, atomic weights had to be added for the following isotopes:

^{22}Na, ^{58m}Co, ^{79}Se, ^{85}Kr, ^{86}Rb, ^{89}Sr, ^{90}Sr, ^{105}Ru, ^{106}Ru, ^{107}Pd, ^{110m}Ag, ^{111}Ag, ^{115m}Cd, ^{113}Sn, ^{123}Sn, ^{125}Sn, ^{126}Sn, ^{124}Sb, ^{125}Sb, ^{126}Sb, ^{127m}Te, ^{129m}Te, ^{132}Te, ^{130}I, ^{131}I, ^{123}Xe, ^{133}Xe, ^{133}Ba, ^{140}Ba, ^{140}La, ^{139}Ce, ^{141}Ce, ^{143}Ce, ^{144}Ce, ^{142}Pr, ^{143}Pr, ^{148m}Pm, ^{151}Pm, ^{153}Sm, ^{156}Eu, ^{157}Eu, ^{153}Gd, ^{160}Tb, ^{166m}Ho, ^{242m}Am, ^{244m}Am, ^{249}Cm, ^{250}Cm, ^{250}Bk, ^{253}Cf, ^{254}Cf, and ^{255}Fm.

These are not necessarily all the new isotopes available for ENDF70 that were not available from ENDF66; these are just the ones for which atomic weights were not previously available in the file "xsdir." Additionally, atomic weights were added for a large range of nuclides,[6] not just those listed above. The purpose of this addition is because MCNPX can generate cross sections for nuclides not represented by ENDF files. Having a wider range of available atomic weights is essential to such calculations.

3.2 Processing Codes

All processing of ENDF/B-VII data was done on the LINUX machine Flash in the directory /usr/projects/data/nuclear/mc/endf7 and is also stored on HPSS under /hpss/nucldata/archive/mc_data/Libs/endf70. The processing was run using a Fortran program called PRENJOY to set up a c-shell script that creates the NJOY input files, runs NJOY, stores all the data, and runs a checking script called CHECKACE.PL on the resulting ACE files. The code PRENJOY creates a c-shell script named RUNNJOY (see Appendix A for an example). PRENJOY, RUNNJOY, NJOY, and CHECKACE.PL were run separately for all evaluations in their own directory. They were named /usr/projects/data/nuclear/mc/endf7/xx*** on the ICN LINUX machine Flash and for the long-term are saved on HPSS under /hpss/nucldata/archive/mc_data/Libs/endf70/xx***. xx is the element and *** represents the atomic mass number of the isotope being requested. In cases where the element is only one character, only one 'x' is required, and in cases where the atomic mass number is less than 100, not all three '*'s are required (e.g., U235 for ^{235}U and H1 for ^{1}H). PRENJOY.F, CHECKACE.PL, and the other Fortran checking codes employed by CHECKACE.PL are currently located in the directory: /usr/projects/data/nuclear/mc/endf7/acs and on HPSS under /hpss/nucldata/archive/mc_data/Libs/endf70/acs. Note that using the default input files, the

resulting ACE-formatted data files did not have the appropriate heading, so the file addline.f under /usr/projects/data/nuclear/mc/endf7/acer was used to fix the descriptor line of the file for each isotope at each temperature.

3.2.1 PRENJOY

The code PRENJOY is run by typing "prenjoy ZZAAA" at the command line. PRENJOY first looks up the material identifier for the isotope being requested from a file called XSLIST7.VI located one directory up (i.e. /usr/projects/data/nuclear/mc/endf7 - see Appendix B for a listing). Then it creates a c-shell script called RUNNJOY with commands to copy files, create NJOY input files, run NJOY99.248, and execute the checking script CHECKACE.PL. All output from running RUNNJOY for an isotope was saved as xx***.log. Typically, numerous isotopes were run serially and/or in parallel using another c-shell program, with the following lines required for each isotope:

new run:	*repeat run for one isotope (such as evaluation change)*
mkdir U235	cd U235
cd U235	rm U235.log
../acs/prenjoy 92235	../acs/prenjoy 92235
csh –v runnjoy > U235.log	csh –v runnjoy > U235.log
cd ..	cd ..

The isotopes were broken into about ten different files to run on Flash; see the files in the directory /usr/projects/data/nuclear/mc/endf7/runfiles and /hpss/nucldata/archive/mc_data/Libs/endf70/runfiles for examples.

3.2.2 RUNNJOY

RUNNJOY first copies the ENDF file from the directory /usr/projects/data/nuclear/mc/endf7/endf (note that the ENDF files for which modifications were required are named xx***.mod (where xx is the element in lowercase) and RUNNJOY MUST be modified by hand to use these files). The ENDF files are also saved on HPSS under /hpss/nucldata/archive/mc_data/Libs/endf70/endf. Then it creates an NJOY input file for processing through the modules MODER, RECONR, BROADR, HEATR, PURR, THERMR, GASPR to create a PENDF file. BROADR is required to obtain temperature-dependent cross sections at all energies of interest, HEATR calculates energy deposition cross sections additional to those in the ENDF file, PURR creates unresolved resonance probability tables, and GASPR generates gas production data. THERMR is not required to generate ACE-formatted files; however, THERMR was included during this NJOY processing to avoid some photon production sum issues found by the CONSIS module of NJOY by providing more data points. Finally, RUNNJOY executes the Perl script CHECKACE.PL to perform quality assurance checking of the data generated.

The original ENDF file is named xx***.endf, and the resulting PENDF file is called xx***.pendf. The NJOY input and output files for this step are saved as xx***.njoyinput.pendf and xx***.njoyoutput.pendf. Then, for each of the five temperatures, RUNNJOY creates an input file to run the ACER module of NJOY three times; first to process data into ACE format, second to run consistency checking, and third to determine if the ACE files produced by NJOY

after consistency checking pass the consistency checking after being generated. The final ACE files are saved as xx***.acer.%%%% where xx is the element, *** is the isotope, and %%%% (either three or four digits) is the temperature (293 is given for 293.6 K, 600 represents 600 K, etc.). The NJOY input and output files from the ACER runs are stored as xx***.njoyinput.%%%%.acer and xxx***.njoyoutput.%%%%.acer. Note that almost all problems found to be significant by the internal consistency checking within NJOY were corrected by ACER with the exception of Eu153, which requires evaluation modifications in the future to solve this problem (see Section 3.3.5).

3.2.3 CHECKACE.PL

CHECKACE.PL runs eleven checking codes written during the processing of ENDF66, providing file names for output and modifications to the ACE files, and evaluating the output to see if possible problems exist. It prints messages about such problems to the xxx***.log file. The ten Fortran checking codes along with the output files and/or new ACE files created by each are given in Table 4. CHECKACE.PL is listed in Appendix C.

Table 4. Checking Codes Used During ENDF70, Including Names of Output and Modified ACE Files

	Output file	New ACE file
check0.f	xxx***.acer.%%%%.c0	xxx***.acer.%%%%.new3
check5.f	xxx***.acer.%%%%.c5	-
check61.f	xxx***.acer.%%%%.c61	xxx***.acer.%%%%.new2
check_heat.f	xxx***.acer.%%%%.cht	-
check_iso.f	xxx***.acer.%%%%.ciso	-
checknd.f	xxx***.acer.%%%%.cnd	-
checknd_neut.f	xxx***.acer.%%%%.cnd_n	xxx***.acer.%%%%.new
checkthresh.f	xxx***.acer.%%%%.cthr	-
check_ures.f	xxx***.acer.%%%%.cures	-
checkxs.f	xxx***.acer.%%%%.cxs	-
check_lownum.pl	N/A	xxx***.acer.%%%%.new4

The Fortran programs were compiled separately and run "stand-alone" from "checkace.pl". In addition, the capabilities of the Perl checking script "check_lownum.pl," which checks for low exponent values was built into "checkace.pl".

A spreadsheet (called "evals_final.xls") was maintained that contained information about whether or not each checking code found any possible errors for all isotopes. It also contained messages produced by MCNP for each isotope and highlighted isotopes for which files were changed as a result of certain checking codes. Final ACE files are also stored individually under: /usr/projects/data/nuclear/mc/endf7/acer and /hpss/nucldata/archive/mc_data/Libs/endf70/acer.

A description of each of the checking codes generated during processing of ENDF66 and only slightly modified for ENDF70 is given below.

CHECK0: This code reads a type 1 MCNP library and checks to make sure that all threshold reactions have a leading zero in their cross-section tabulation. It checks neutron cross sections, type-13 photon-production cross sections, and other particle production cross sections. An error is printed to the output file and carried over to the log file of the isotope/element being examined by CHECKACE.PL if any problems are found.
CHECK5: This code reads in a type 1 MCNP data library and checks the information for MT=5. The code prints out the threshold energy and the first pair (energy, yield) of data for secondary neutrons. It checks to see if the threshold for MT=5 is lower than the first data pair, in which case the first yield must be zero. If no MT=5 reaction data if found there is no printed output. CHECKACE.PL reviews the output file and prints all messages to the log file. These messages must be reviewed to assure no problems exist.

CHECKND: The program checknd reads in an MCNP type 1 library file and checks various aspects of the secondary photon energy distributions which use LAW=4 or 44. It informs the user if the secondary photon energies are discrete and if there are any negative discrete energies at adjacent incident neutron energies. All "errors" printed to the output file are transmitted to the log file by CHECKACE.PL.

CHECKND_NEUT: The program reads in an MCNP type 1 library file and checks various aspects of the secondary neutron energy distributions which use LAW=4 or 44. It verifies that interpolation schemes 1 or 2 are used, and identifies any negative probability density functions. The code checks to see if any secondary neutrons are produced with energy greater than the incident neutron energy and takes corrective action with the exception of fission. It also fixes negative probabilities by setting them to zero. Errors in the output file are transmitted to the log file by CHECKACE.PL. Though no corrective action is taken for fission, one still can get a "error" message for fission MT's. Note that problems existed in the methodology checknd_neut used to correct the incident energy. A new version of checknd_neut.f (checknd_neut2.f) was written to help avoid some problems but was not completely correct nor used. Thus, only ACE files modified by checknd_neut for negative probability density functions were included in ENDF70. Changes for outgoing neutron energy problems were small and considered insignificant.

CHECK61: The program check61 reads in an MCNP type 1 library file and checks various aspects of the secondary neutron energy distributions which use LAW=61. It verifies that interpolation schemes 1 or 2 are used, and identifies any negative probability density functions. The code checks to see if any secondary neutrons are produced with energy greater than the incident neutron energy and takes corrective action, again with the exception of fission.. Though no corrective action is taken for fission, one still can get a "error" message for fission MT's. It also fixes the negative probabilities by setting them to zero. All errors given in the output file are transmitted to the log file by CHECKACE.PL. However, for consistency with checknd_neut, only ACE files corrected for errors with negative probability density functions were included in ENDF70. Changes for outgoing neutron energy problems were small and considered insignificant.

CHECK_HEAT: This code reads in a type 1 MCNP data library and checks various aspects of heating numbers. First, it checks the main heating grid for zeros or negatives. Then, it checks

each of the partial particle heating arrays for the same features. Finally, it totals these partial heating arrays and does some comparisons of this total to the main grid. Again, errors found in the output file were carried over to the log file by CHECKACE.PL.

CHECK_ISO: This code reads in a type 1 MCNP data library and checks all neutron reactions that produce secondary neutrons. If the angular distributions are isotropic, it prints out the zaid, mt, and tyr. The reason for this code is that an error appeared in NJOY99.50 that gave a positive tyr for reactions with isotropic distributions, even if the isotropy is specified to be center-of-mass (an example from ENDF66 is MTs 56 and higher in ^{16}O). It also checks to make sure TYR is 19 for all fission reactions. For ENDF70, this test helped identify a problem in the evaluation for ^{242}Am (ground state) as will be discussed in Section 3.3.1. All values of TYR greater than zero plus all fission and inelastic MTs are printed to the output file and transmitted to the log file by CHECKACE.PL. These values must be verified by hand to assure they are not unreasonable.

CHECKTHRESH: This code reads in a type 1 MCNP data library and checks the reaction threshold energies with the kinematic thresholds for negative Q-value reactions. If the library threshold is lower than the kinematic threshold the codes outputs this information along with the magnitude of the discrepancy, the law specified for the secondary neutrons and the line number on the type 1 library file containing the problem energy value. If no errors are detected no print out is given. All errors given in the output file are transmitted to the log file by CHECKACE.PL.

CHECK_URES: This program analyzes the unresolved resonance probability tables generated by the PURR module of NJOY to assure that the partial cross sections or factors add up appropriately to totals and that no unrealistic heating values exist. It lists all the cross sections and factors for each incident energy and prints warning messages when partials do not add up correctly to the total within 1 and/or 5%, if some values are zero, or if unrealistic heating numbers exist. All errors given in the output file are transmitted to the log file by CHECKACE.PL as are the total number of potential problems identified by check_ures.

CHECKXS: This program reads in a Type 1 MCNP library, and for each zaid it compares the total cross section to the sum of the partials for neutrons and for photon production. This version of the code also verifies that the sum of the 600-800's series of MT's adds up to the corresponding total in MT=103-107 and checks for negative cross sections (such as MT91 for Y89). All errors given in the output file are transmitted to the log file by CHECKACE.PL.

CHECK_LOWNUM.PL: Checks an ACE file for numbers in scientific notation which are < 10^{-30}. The lines of the ACE files with such low numbers are printed to the log file and must be modified by hand if they are less than or equal to 10^{-37}. They are ignored if greater than 10^{-37}.

3.3 Results of Verification and Validation

Several issues were found with the processing/evaluations. The evaluation issues are discussed first followed by outcomes of the checking code results/changes for quality assurance.

3.3.1 Evaluation Changes

Evaluations were modified for the following isotopes: ^{1}H, ^{45}Sc, ^{89}Y, ^{96}Zr, ^{97}Mo, ^{233}U, ^{242}Am (ground state), and ^{242m}Am. Detailed changes to the evaluations are given in Appendix D but are summarized here.

H1: The value for the energy of the photon from radiative capture was modified to 2.2233 MeV from 2.2246. The new value is the actual energy of the photon from thermal capture. The previous evaluations gave the total energy and did not take into account the recoil of the nucleus (hence impacting heating as well). This was found in ENDF66 and did not get changed in ENDF/B-VII.

Sc45: In ENDF66, it was found that this evaluation had to be modified because File 13, MT=3 did not have the secondary distribution for photons beginning with a zero point at the threshold. Additionally, the evaluation had the incorrect reference frame specified for angular distributions of (n,2n) and (n,n*)a, (n,n*)p, and (n,n*c) as found by CONSIS/ACER. Both of these problems still existed in the ENDF/B-VII evaluation, so the same changes were made as were made previously.

Y89: Some negative cross sections existed from 1.7 to 4.5 MeV in reaction 91 (inelastic cross section not included in other channels). Thus, reaction 91 was changed from 35 lines with numerous negative cross sections to only 14 lines with values that were the difference of MT 4 (total) and all MT 51-90 (partial) reactions as summed on a spreadsheet.

Zr96 and Mo97:

Errors from the CONSIS checking option in the ACER module of NJOY indicated that the value of r in law44 kalbach was incorrect. It should be a fraction between 0 and 1, but values such as 9.999 were appearing. It turns out that there were type-o's in the evaluations for Zr96 and Mo97 that gave results such as 9.999… instead of 0.9999…. These two evaluations were thus modified accordingly.

U-233:

It turns out that there was an error in the exponent for the delayed- neutron yields for U-233 above 9 MeV. They should be e-3 instead of e-2.

Am242 (ground): The ^{242}Am ENDF file did not have an angular distribution for fission (i.e. no file 4 nor 6 for mt=18), which led to an incorrect value of TYR. The isotropic mf4, mt18 data from ^{242m}Am file was simply inserted into the ^{242}Am file as a modification.

Am242m: The inelastic cross sections (MT 4, 51, 52, 53, and 54) contained values of zero between 50 and 65 keV, whereas values above and below were non-zero. The cross sections below 100 keV were smoothed out by the evaluator in this modification.

3.3.2 Heating Problems

The checking code "check_heat" found some heating inconsistencies (the sum of the partials not adding up to the total or being greater than the total), most of which occur for isotopes with energy balance problems. It also found some cases of negative heating. The isotopes for which

these problems occur (with bolded values representing isotopes with negative heating cross sections) are:
^{47}Ti, ^{48}Ti, ^{49}Ti, ^{74}As, **^{92}Mo**, **^{94}Mo**, **^{96}Mo**, **^{97}Mo**, **^{98}Mo**, ^{101}Ru, ^{105}Pd, ^{113}Sn, ^{131}Xe, ^{133}Ba, **^{143}Ce**, **^{145}Nd**, **^{147}Nd**, ^{148}Nd, ^{150}Nd, **^{147}Sm**, ^{148}Sm, **^{149}Sm**, **^{151}Sm**, **^{153}Gd**, ^{154}Gd, **^{155}Gd**, ^{156}Gd, ^{157}Gd, ^{191}Ir, and ^{208}Pb.
Note that using Versions 224 and earlier of NJOY99 will lead to even more energy balance problems in the heating cross section than those listed here, which are associated with Version 248. During early testing of the ENDF/B-VII data, many problems were identified by "check_heat" and substantial upgrades were made to NJOY to correct them by Version 248.

3.3.3 Unresolved Resonance Range Issues
During ENDF66 processing of ENDF/B-VI Release 6 data using NJOY99 Version 50, problems were identified with heating cross sections and factors output in the probability tables not having the right units, and a separate code was written to fix these problems. Version 76 of NJOY99 corrected most of these problems, and Version 172 of NJOY99 corrected all the problems (especially heating factors). Minor changes were also added in Version 245 to help the sampling of narrow widely spaced resonances where "check_ures" also reported warnings.

There are still a number of isotopes for which some values in the probability tables are not summing up to the totals predicted (within 5%) and/or there are negative heating factors. Thus, probability tables have been excluded (i.e. NJOY was run without PURR) for the following isotopes:

^{22}Na, ^{36}Ar, ^{38}Ar, ^{74}As, ^{79}Se, ^{82}Kr, ^{90}Zr, ^{94}Nb, ^{95}Nb, ^{99}Mo, ^{106}Cd, ^{123}Sn, ^{126}Sn, ^{125}Sb, ^{131}I, ^{136}Cs, ^{139}Ce, ^{143}Pr, ^{144}Nd, ^{148}Pm, ^{151}Pm, ^{153}Sm, ^{152}Eu, ^{153}Eu, ^{154}Eu, ^{155}Eu, ^{156}Dy, ^{158}Dy, ^{181}Ta, ^{252}Cf, ^{253}Cf, and ^{253}Es.

Such modifications to the RUNNJOY c-shell script had to be made by hand.

3.3.4 Photon Production Sum Problems in CONSIS
As mentioned previously, the THERMR module in NJOY was included during this NJOY processing to avoid some photon production sum issues found by the CONSIS module of NJOY. All isotopes were processed with THERMR, but the specific isotopes for which these issues initially existed when THERMR was not included were: ^{1}H, ^{40}Ca, ^{42}Ca, ^{43}Ca, ^{44}Ca, ^{46}Ca, ^{48}Ca, ^{70}Ge, ^{72}Ge, ^{73}Ge, ^{74}Ge, ^{76}Ge, ^{74}As, ^{75}As, ^{85}Kr, ^{86}Rb, ^{84}Sr, ^{90}Y, ^{90}Zr, ^{91}Zr, ^{92}Zr, ^{94}Zr, ^{96}Zr, ^{95}Mo, ^{99}Tc, ^{101}Ru, ^{103}Rh, ^{105}Pd, ^{109}Ag, ^{111}Ag, ^{115m}Cd, ^{113}Sn, ^{125}Sn, ^{126}Sb, ^{132}Te, ^{130}I, ^{131}Xe, ^{133}Cs, ^{133}Ba, ^{140}La, ^{136}Ce, ^{138}Ce, ^{139}Ce, ^{143}Ce, ^{141}Pr, ^{142}Pr, ^{142}Nd, ^{143}Nd, ^{144}Nd, ^{145}Nd, ^{146}Nd, ^{147}Nd, ^{148}Nd, ^{150}Nd, ^{151}Pm, ^{144}Sm, ^{147}Sm, ^{148}Sm, ^{149}Sm, ^{150}Sm, ^{151}Sm, ^{152}Sm, ^{153}Sm, ^{154}Sm, ^{153}Eu, ^{157}Eu, ^{152}Gd, ^{153}Gd, ^{154}Gd, ^{155}Gd, ^{156}Gd, ^{157}Gd, ^{158}Gd, ^{160}Gd, ^{159}Tb, ^{160}Tb, ^{156}Dy, ^{158}Dy, ^{160}Dy, ^{161}Dy, ^{162}Dy, ^{163}Dy, ^{164}Dy, ^{165}Ho, ^{166m}Ho, ^{191}Ir, ^{193}Ir, ^{232}Th, ^{231}Pa, ^{233}Pa, and ^{237}Np.

3.3.5 Modifications to the ACE file Made by CHECK61 and CHECKND_NEUT
The checking code CHECK61 checks various aspects of the secondary neutron energy distributions that use LAW=61. It verifies that interpolation schemes 1 or 2 are used, identifies any negative probability density functions, and sets them to zero. The code also checks to see if any secondary neutrons are produced with energy greater than the incident neutron energy and

takes corrective action (except for fission, which it does not correct). The only isotope for which CHECK61 found negative probability density functions was ^{153}Eu, so these were set to zero in the corrected ACE file. There were some minor problems found by CHECKND_NEUT and CHECK61 related to incident/secondary neutron energies, but the modified ACE files for neutrons were not used because the changes were 1) not significant and 2) often modified incorrectly (especially by CHECKND_NEUT, which is why the Fortran program CHECKND_NEUT2.F was created to try and solve these problems but needs more work if used in the future).

3.3.6 Modifications to the ACE file Resulting from Output from CHECK0 and CHECK_LOWNUM.PL

Changes to the ACE file resulting from the Perl script CHECK_LOWNUM.PL were made for seven isotopes with exponents less than or equal to e-37: ^{16}O, ^{40}Ca, ^{42}Ca, ^{43}Ca, ^{44}Ca, ^{46}Ca, and ^{204}Pb. The exponents were simply changed to e-35 for these cases. The checking code CHECK0 only found a problem for one isotope, ^{10}B. Note also that although one change was made to the ACE file by CHECK0 for a leading non-zero threshold photon-production cross section value, the total photon production cross section at that energy had to be decremented by hand in the file. For example, for isotope ^{10}B, the cross section was 1.94222e-09, but it was changed to 0.0, and the sum 0.189405100000 became 0.189405098058.

3.3.7 Problems with ^{1}H

3.3.7.1 Warning Found by CHECKND

Using the ENDF/B-VII data for ^{1}H, NJOY chose to use an MCNP Law = 4 representation for the energy distribution of the photon (now in File 6), whereas it used to represent it with Law = 2 with the ENDF/B-VI data. It has essentially tabulated the function "E' = 2.2233 + Eneut * (AWR)/(AWR + 1)" by using a tabulated energy distribution as a function of incident energy, where each tabulated distribution contains only the one value calculated from above (specified as discrete) with linear-linear interpolation between incident energies. This new technique gives the same result as the Law=2 one but leads to the NJOY warning message: "discrete anisotropic photon treated as simple primary photon" and warnings in the checking code CHECKND.

3.3.7.2 Deuteron Production Not Calculated Correctly

Using Version 248 of NJOY with ENDF/B-VII data for ^{1}H, there was a problem with deuteron production following neutron capture on hydrogen. ^{1}H was the only nuclide affected because in ENDF/B-VII, the 1h capture photon data were removed from mf12 & mf14 and put into mf6. Basically, at 1.e-11 MeV, the cross section for neutron absorption (16.70111) equals both the mt102 cross section and the mt 204 cross section. However, in the deuterium production portion of the file, the cross section at 1.e-11 eV is twice what it should be (33.40222). Update 271 of NJOY contains a solution to this problem, so this update by itself was added to Version 248 of NJOY and ^{1}H was reprocessed to correct it.

3.3.8 MCNP "Hoho" Test

In addition to the other quality assurance checking codes, resulting ACE files for all isotopes were run through a sample MCNP file to assure the data is read in and processed correctly. The sample file involves a source starting over a large energy range up to 20 MeV for a sphere with a

density of 20 g/cc of the isotope/element being tested (see Appendix E for an example). Some warning messages from MCNP include:

"lacks gamma ray production cross sections"
"1 coincident energy grid point"
"no photon-production mt found in acegam" (looped failed no more than 2 times for any isotope)
"nubar may be prompt or total"
"material 1 has been set to a conductor"
"comment. 1 cross sections modified by free gas thermal treatment"

The messages relevant for each isotope are listed in the spreadsheet "evals_final.xls". Also, some MCNP input and output files are saved under /usr/projects/data/nuclear/mc/hoho/mcnp and long-term under /hpss/nucldata/archive/mc_data/Libs/endf70/hoho. There are five files for each isotope with extensions h1 through h5 that correspond to each of the five temperatures 293.6 K, 600 K, 900 K, 1200 K, and 2500 K respectively. Intermediate Fortran files used to set up the MCNP input files correctly and run them are found in the directory /usr/projects/data/nuclear/mc/hoho and /hpss/nucldata/archive/mc_data/Libs/endf70/hoho.

3.3.9 Plotting

The final "test" performed on the data was to plot all reactions for all isotopes using the program "xsplot." This program pulls up all available reactions for an isotope from various formats and plots it at all energies. The ENDF70 library was directly plotted at all energies on the machine "xfiles", for which the executable XSPLOT2 was valid. There were many cases where there were discontinuities in the data at a certain energy (i.e. some of the Ac isotopes and other actinides evaluations had assumed values below 0.6 eV and other models above 0.6 eV, so two very different cross sections existed at the same energy). There were also numerous cases of sharps dips in the heating cross section (MT 301) at an energy, often corresponding to negative heating values. Such anomalies were ignored at this time. However, the only major problem found thus far was the inelastic cross section of ^{242m}Am, which was modified. Most significant findings are documented in Appendix F.

4.0 The Proton Continuous Energy Library ENDF70

The forty-seven proton evaluations processed with NJOY at room temperature (293.6 K) were given identifiers of ".70h" and combined into one file, ENDF70PROT.

4.1 Processing

Only the ACER module of NJOY was used (again three times to allow for consistency checking and modifications), but the same procedure was used as for the neutron evaluations. The program PRENJOY2.F was created for the proton processing based on PRENJOY.F for the neutron processing except it created different NJOY input files (PRENJOY2.F is stored under /hpss/nucldata/archive/mc_data/Libs/endf70prot/acs). RUNNJOY was used to copy files to the correct places and run NJOY and CHECKACE.PL. Results from these calculations were stored in the spreadsheet "evals_pro.xls". One problem found when processing the proton evaluations with NJOY is that the ZAID was incremented by a value of 1 after ACER was run. The ZAIDs in the final files had to be modified by hand to be consistent with the incoming value.

4.2 Validation and Verification

For ^{28}Si, there were CONSIS errors related to photon production sum mismatches that were not resolved by CONSIS. In addition, the checking code CHECKND found problems with ^{2}H and ^{9}Be that were not resolved. For ^{2}H, there was a negative value of eprime at 0.1 MeV which was modified by hand to being positive in the ACE file; ^{9}Be just displayed warnings. Additionally, CHECKND_NEUT found some negative probability density functions for ^{9}Be and set them to zero (kept separately in the file Be9.new).

Many of the resulting ACER files had numbers less than or equal to e-37. For the following isotopes, values less than or equal to e-37 were modified by hand to be e-35: ^{6}Li, ^{56}Fe, ^{182}W, ^{183}W, ^{184}W, ^{186}W, ^{200}Hg, ^{201}Hg, ^{202}Hg, ^{206}Pb, and ^{207}Pb. All isotopes in the resulting proton continuous energy library were run through an MCNPX "hoho" test similar to that run for the neutron library with no problems found.

5.0 Remaining Issues/Future Work

In comparing some ENDF/B-VII.0 ACE-formatted files with multi-group data both from LANL and from Livermore National Laboratory, Robert MacFarlane (LANL) and Culler (LLNL) found some discrepancies at high energies. These are discussed below. Resolution of these issues should occur sometime in the future.

5.1 Fission spectrum interpolation

In U-235 and Pu-239, the energy grid for the fission spectra changes from 0.1 MeV below 10 MeV to 1 MeV above. Because linear-linear interpolation is specified, the deviations between the realistic exponential shape in the region above 10 MeV and the linear interpolate get quite large between the secondary-energy grid points (deviations up to 9%). Patches have been developed to alleviate this problem but were not implemented in ENDF70. Changes to these evaluations would be beneficial in the future.

5.2 Incident-energy interpolation

MF6/MT91 for the major actinides specifies linear interpolation on incident energy. Apparently, MCNP automatically changes this to some kind of a unit base interpolation in order to smooth out the shapes for energies between the incident energies given. In the future, either MCNP needs to be modified or the ENDF/B-VII evaluations should be changed to reflect an explicit int=22 interpolation law for the MF6 sections.

5.3 Histograms for continuum distributions

Most of the actinides, and many of the other materials based on model calculations, describe outgoing spectra in MF6 using histograms. The models normally use bins based on delta E. Of course, the realistic shape in the low-energy range of such spectra is sqrt(E'). The difference between sqrt(E') and the histogram shape leads to some edges in the computed flux for metallic assemblies like Godiva or Jezebel. These edges become evident when MCNP flux calculations are compared to TART, which makes an attempt to smooth out the histograms. The effect on k-eff is fairly small. A patch for ACE processing in NJOY has been developed to analyze the histograms to find the region that acts like sqrt(E') and inserts additional histogram segments using log energy spacing with small enough steps to minimize the edges in the computed flux. This patch was not implemented in the ENDF70 processing

5.4 Delayed Neutron Spectra

The delayed neutron spectra also use histograms. Since the low break is at 10 keV, the effect of that break can usually be seen as a step in the computed flux. It would be beneficial to change the shape of the spectrum so we could get very high delayed neutron sources into the low energy region in addition to the evident step in the flux at 10 keV. A patch could be applied to NJOY to do this shape changing or modified ENDF/B-VII evaluations could be prepared with smoother distributions for the MF5/MT455 sections.

6.0 Conclusions

In summary, the new neutron library ENDF70 offers more variety in the number of temperatures and available isotopes than in the past. The large amount of effort devoted to developing checking codes and other testing in ENDF66 helped the ENDF70 library be released in approximately a year after the evaluated ENDF/B-VII data became available. The resulting data takes up more space than in the past but provides improved cross sections for numerous calculations. The proton library ENDF70PROT contains improved data for almost all isotopes and was validated using the same codes as the neutron library. These libraries should benefit many users and will be released with MCNP5 1.50.

References:

1. M.B. Chadwick, P. Oblozinsky, M. Herman, N.M. Greene, R.D. McKnight, D.L.Smith, P.G. Young, R.E. MacFarlane, G.M. Hale, S.C. Frankle, A.C. Kahler, T. Kawano, R.C. Little, D.G. Madland, P. Moller, R.D. Mosteller, P.R. Page, P. Talou, H. Trellue, M.C. White, W.B. Wilson, R. Arcilla, C.L. Dunford, S.F. Mughabghab, B. Pritychenko, D. Rochman, A.A. Sonzogni, C.R. Lubitz, T.H. Trumbull, J.P. Weinman, D.A. Brown, D.E. Cullen, D.P. Heinrichs, D.P. McNabb, H. Derrien, M.E. Dunn, N.M. Larson, L.C. Leal, A.D. Carlson, R.C. Block, J.B. Briggs, E.T. Cheng, H.C. Huria, M.L. Zerkle, K.S. Kozier, A. Courcelle, V.Pronyaev and S.C. van der Marck, "ENDF/B-VII.0: Next Generation Evaluated Nuclear Data Library for Nuclear Science and Technology", Nuclear Data Sheets, vol. 107, (2006) pp. 2931-3060.

2. R. E. MacFarlane, and D. W. Muir, "The NJOY Nuclear Data Processing System Version 91," Los Alamos National Laboratory manual LA-12740-M (October 1994).

3. J. M. Campbell, S. C. Frankle, and R. C. Little, "ENDF66: A Continuous-Energy Neutron Data Library for MCNP4C," Los Alamos National Laboratory report LA-UR-02-3000 (2002).

4. J. M. Campbell, S. C. Frankle, and R. C. Little, "ENDF66: A Continuous-Energy Neutron Data Library Based on ENDF/B-VI Release 6," Los Alamos National Laboratory report LA-UR-03-0954 (2003).

5. J. L. Conlin, F. B. Brown, and R. D. Mosteller, "Temperature Corrections in MCNP For Calculating the Doppler Defect," Los Alamos National Laboratory report LA-UR-05-6225 (August 2005).

6. G. AUDI, A.H. WAPSTRA, C. THIBAULT, J. BLACHOT and O. BERSILLON "Ame2003: Atomic Mass Evaluation," November 2003, <http://www.nndc.bnl.gov/amdc/web/masseval.html> (November 2007).

ADC: MCW

Distribution:

X-1 files

M. Gray	X-1-NAD	MS F663	P. Talou	T-16	MS B283
M. White	X-1-NAD	MS F663	R. MacFarlane	T-16	MS B243
K. Parsons	X-1-NAD	MS F663	A. C. Kahler	T-16	MS B243

Appendix A. Sample RUNNJOY c-shell script

```
date
echo processing 92235
cat>input <<EOF
 moder
 20 -21
 reconr
 -21 -22
 pendf tape for U -235  from endf b-vii njoy99.182
        9228  7 /
 .001 /
 'the following reaction types are added where available'/
 '     mt152   bondarenko unresolved'/
 '     mt153   unresolved probability tables'/
 '     mt20x   gas production'/
 '     mt221   free thermal scattering'/
 '     mt301   total heating kerma factor'/
 '     mt444   total damage energy production'/
 0 /
 broadr
 -21 -22 -23
        9228  5 0 0 /
  .001 /
 293.6  600.0  900.0 1200.0 2500.0
  0 /
 heatr
 -21 -23 -25 /
        9228  7 /
 302 303 304 318 402 443 444 /
 heatr
 -21 -23 -22 24/
        9228  7 0 1 0 2/
 302 303 304 318 402 443 444 /
 purr
 -21 -25 -26
        9228  5 1 20 64 /
 293.6  600.0  900.0 1200.0 2500.0  /
 1e10 /
 0 /
 thermr
 0 -26 -27
 0          9228  16 5 1 0 1 221 1 /
 293.6  600.0  900.0 1200.0 2500.0  /
 0.001 10. /
 gaspr
 -21 -27 -28
 moder
 -28 30
 stop
EOF
cp /usr/projects/data/nuclear/mc/endf7/endf/n-092_U_235.endf tape20
cp tape20 U235.endf
bpsh $NODES ../../../njoy/njoy248/intel-9.1.037-f-O2/xnjoy<input
cp output U235.njoyoutput.pendf
cp input U235.njoyinput.pendf
```

```
cp tape30 U235.pendf
date
echo processing 92235
cat>input <<EOF
 moder
 20 -21
 moder
 30 -31
 acer
-21 -28 0 50 51
 1 0 1 .70 /
 acer tape for U -235 at       293.6  K from endf b-vii njoy99.182
   9228      293.6 /
 /
 /
 acer
 0 50 0 54 55
  7 1 1 -1 /
 acer tape for U -235 at       293.6  K from endf b-vii njoy99.182
 acer
 0 54 0 56 57
  7 1 1 -1 /
 acer tape for U -235 at       293.6  K from endf b-vii njoy99.182
 stop
EOF
bpsh $NODES ../../../njoy/njoy248/intel-9.1.037-f-O2/xnjoy<input
date
cp output U235.njoyoutput.293.acer
cp input U235.njoyinput.293.acer
mv tape54 U235.acer.293
../acs/t24 55 U235.acer.293
cp U235.acer.293 ../acer
../acs/checkace.pl U235.acer.293
if (-e xsdir.tmp) then
cp xsdir.tmp xsdir.tmp.tmp
cat xsdir.tmp.tmp xsdir.t24 > xsdir.tmp
rm xsdir.tmp.tmp xsdir.t24
else
mv xsdir.t24 xsdir.tmp
rm xsdir.t24
endif
date
echo processing 92235
cat>input <<EOF
 moder
 20 -21
 moder
 30 -31
 acer
-21 -28 0 60 61
 1 0 1 .71 /
 acer tape for U -235 at       600.0  K from endf b-vii njoy99.182
   9228      600.0 /
 /
 /
 acer
 0 60 0 64 65
```

```
 7 1 1 -1 /
 acer tape for U -235 at       600.0  K from endf b-vii njoy99.182
 acer
 0 64 0 66 67
 7 1 1 -1 /
 acer tape for U -235 at       600.0  K from endf b-vii njoy99.182
 stop
EOF
bpsh $NODES ../../../njoy/njoy248/intel-9.1.037-f-O2/xnjoy<input
date
cp output U235.njoyoutput.600.acer
cp input U235.njoyinput.600.acer
mv tape64 U235.acer.600
../acs/t24 65 U235.acer.600
cp U235.acer.600 ../acer
../acs/checkace.pl U235.acer.600
if (-e xsdir.tmp) then
cp xsdir.tmp xsdir.tmp.tmp
cat xsdir.tmp.tmp xsdir.t24 > xsdir.tmp
rm xsdir.tmp.tmp xsdir.t24
else
mv xsdir.t24 xsdir.tmp
rm xsdir.t24
endif
date
echo processing 92235
cat>input <<EOF
 moder
 20 -21
 moder
 30 -31
 acer
-21 -28 0 70 71
 1 0 1 .72 /
 acer tape for U -235 at       900.0  K from endf b-vii njoy99.182
   9228      900.0 /
 /
 /
 acer
 0 70 0 74 75
 7 1 1 -1 /
 acer tape for U -235 at       900.0  K from endf b-vii njoy99.182
 acer
 0 74 0 76 77
 7 1 1 -1 /
 acer tape for U -235 at       900.0  K from endf b-vii njoy99.182
 stop
EOF
bpsh $NODES ../../../njoy/njoy248/intel-9.1.037-f-O2/xnjoy<input
date
cp output U235.njoyoutput.900.acer
cp input U235.njoyinput.900.acer
mv tape74 U235.acer.900
../acs/t24 75 U235.acer.900
cp U235.acer.900 ../acer
../acs/checkace.pl U235.acer.900
if (-e xsdir.tmp) then
```

```
cp xsdir.tmp xsdir.tmp.tmp
cat xsdir.tmp.tmp xsdir.t24 > xsdir.tmp
rm xsdir.tmp.tmp xsdir.t24
else
mv xsdir.t24 xsdir.tmp
rm xsdir.t24
endif
date
echo processing 92235
cat>input <<EOF
 moder
 20 -21
 moder
 30 -31
 acer
-21 -28 0 80 81
 1 0 1 .73 /
 acer tape for U -235 at      1200.0  K from endf b-vii njoy99.182
   9228     1200.0 /
 /
 /
 acer
 0 80 0 84 85
  7 1 1 -1 /
 acer tape for U -235 at      1200.0  K from endf b-vii njoy99.182
 acer
 0 84 0 86 87
  7 1 1 -1 /
 acer tape for U -235 at      1200.0  K from endf b-vii njoy99.182
 stop
EOF
bpsh $NODES ../../../njoy/njoy248/intel-9.1.037-f-O2/xnjoy<input
date
cp output U235.njoyoutput.1200.acer
cp input U235.njoyinput.1200.acer
mv tape84 U235.acer.1200
../acs/t24 85 U235.acer.1200
cp U235.acer.1200 ../acer
../acs/checkace.pl U235.acer.1200
if (-e xsdir.tmp) then
cp xsdir.tmp xsdir.tmp.tmp
cat xsdir.tmp.tmp xsdir.t24 > xsdir.tmp
rm xsdir.tmp.tmp xsdir.t24
else
mv xsdir.t24 xsdir.tmp
rm xsdir.t24
endif
date
echo processing 92235
cat>input <<EOF
 moder
 20 -21
 moder
 30 -31
 acer
-21 -28 0 90 91
 1 0 1 .74 /
```

```
 acer tape for U -235 at      2500.0  K from endf b-vii njoy99.182
   9228     2500.0 /
 /
 /
 acer
 0 90 0 94 95
  7 1 1 -1 /
 acer tape for U -235 at      2500.0  K from endf b-vii njoy99.182
 acer
 0 94 0 96 97
  7 1 1 -1 /
 acer tape for U -235 at      2500.0  K from endf b-vii njoy99.182
 stop
EOF
bpsh $NODES ../../../njoy/njoy248/intel-9.1.037-f-O2/xnjoy<input
date
cp output U235.njoyoutput.2500.acer
cp input U235.njoyinput.2500.acer
mv tape94 U235.acer.2500
../acs/t24 95 U235.acer.2500
cp U235.acer.2500 ../acer
../acs/checkace.pl U235.acer.2500
if (-e xsdir.tmp) then
cp xsdir.tmp xsdir.tmp.tmp
cat xsdir.tmp.tmp xsdir.t24 > xsdir.tmp
rm xsdir.tmp.tmp xsdir.t24
else
mv xsdir.t24 xsdir.tmp
rm xsdir.t24
endif
 rm tape*
```

Appendix B. File "xslist7.vi" – list of material identifiers used by "prenjoy.f"

```
 1001  125  1 H   1
 1002  128  1 H   2
 1003  131  1 H   3
 2003  225  2 He  3
 2004  228  2 He  4
 3006  325  3 Li  6
 3007  328  3 Li  7
 4007  419  4 Be  7
 4009  425  4 Be  9
 5010  525  5 B  10
 5011  528  5 B  11
 6000  600  6 C   0
 7014  725  7 N  14
 7015  728  7 N  15
 8016  825  8 O  16
 8017  828  8 O  17
 9019  925  9 F  19
11022 1122 11 Na 22
11023 1125 11 Na 23
12024 1225 12 Mg 24
12025 1228 12 Mg 25
12026 1231 12 Mg 26
13027 1325 13 Al 27
14028 1425 14 Si 28
14029 1428 14 Si 29
14030 1431 14 Si 30
15031 1525 15 P  31
16032 1625 16 S  32
16033 1628 16 S  33
16034 1631 16 S  34
16036 1637 16 S  36
17035 1725 17 Cl 35
17037 1731 17 Cl 37
18036 1825 18 Ar 36
18038 1831 18 Ar 38
18040 1837 18 Ar 40
19039 1925 19 K  39
19040 1928 19 K  40
19041 1931 19 K  41
20040 2025 20 Ca 40
20042 2031 20 Ca 42
20043 2034 20 Ca 43
20044 2037 20 Ca 44
20046 2043 20 Ca 46
20048 2049 20 Ca 48
21045 2125 21 Sc 45
22046 2225 22 Ti 46
22047 2228 22 Ti 47
22048 2231 22 Ti 48
22049 2234 22 Ti 49
22050 2237 22 Ti 50
23000 2300 23 V   0
24050 2425 24 Cr 50
24052 2431 24 Cr 52
24053 2434 24 Cr 53
24054 2437 24 Cr 54
25055 2525 25 Mn 55
26054 2625 26 Fe 54
26056 2631 26 Fe 56
26057 2634 26 Fe 57
26058 2637 26 Fe 58
27058 2722 27 Co 58
27458 2723 27 Co 58m
27059 2725 27 Co 59
28058 2825 28 Ni 58
28059 2828 28 Ni 59
28060 2831 28 Ni 60
28061 2834 28 Ni 61
```

```
28062 2837 28 Ni 62
28064 2843 28 Ni 64
29063 2925 29 Cu 63
29065 2931 29 Cu 65
30000 3000 30 Zn  0
31069 3125 31 Ga 69
31071 3131 31 Ga 71
32070 3225 32 Ge 70
32072 3231 32 Ge 72
32073 3234 32 Ge 73
32074 3237 32 Ge 74
32076 3243 32 Ge 76
33074 3322 33 As 74
33075 3325 33 As 75
34074 3425 34 Se 74
34076 3431 34 Se 76
34077 3434 34 Se 77
34078 3437 34 Se 78
34079 3440 34 Se 79
34080 3443 34 Se 80
34082 3449 34 Se 82
35079 3525 35 Br 79
35081 3531 35 Br 81
36078 3625 36 Kr 78
36080 3631 36 Kr 80
36082 3637 36 Kr 82
36083 3640 36 Kr 83
36084 3643 36 Kr 84
36085 3646 36 Kr 85
36086 3649 36 Kr 86
37085 3725 37 Rb 85
37086 3728 37 Rb 86
37087 3731 37 Rb 87
38084 3825 38 Sr 84
38086 3831 38 Sr 86
38087 3834 38 Sr 87
38088 3837 38 Sr 88
38089 3840 38 Sr 89
38090 3843 38 Sr 90
39089 3925 39 Y  89
39090 3928 39 Y  90
39091 3931 39 Y  91
39089 3925 39 Y  89
40090 4025 40 Zr 90
40091 4028 40 Zr 91
40092 4031 40 Zr 92
40093 4034 40 Zr 93
40094 4037 40 Zr 94
40095 4040 40 Zr 95
40096 4043 40 Zr 96
41093 4125 41 Nb 93
41094 4128 41 Nb 94
41095 4131 41 Nb 95
42092 4225 42 Mo 92
42094 4231 42 Mo 94
42095 4234 42 Mo 95
42096 4237 42 Mo 96
42097 4240 42 Mo 97
42098 4243 42 Mo 98
42099 4246 42 Mo 99
42100 4249 42 Mo100
43099 4325 43 Tc 99
44096 4425 44 Ru 96
44098 4431 44 Ru 98
44099 4434 44 Ru 99
44100 4437 44 Ru100
44101 4440 44 Ru101
44102 4443 44 Ru102
44103 4446 44 Ru103
44104 4449 44 Ru104
44105 4452 44 Ru105
```

```
44106 4455 44 Ru106
45103 4525 45 Rh103
45105 4531 45 Rh105
46102 4625 46 Pd102
46104 4631 46 Pd104
46105 4634 46 Pd105
46106 4637 46 Pd106
46107 4640 46 Pd107
46108 4643 46 Pd108
46110 4649 46 Pd110
47107 4725 47 Ag107
47109 4731 47 Ag109
47510 4735 47 Ag110m
47111 4737 47 Ag111
48106 4825 48 Cd106
48108 4831 48 Cd108
48110 4837 48 Cd110
48111 4840 48 Cd111
48112 4843 48 Cd112
48113 4846 48 Cd113
48114 4849 48 Cd114
48515 4853 48 Cd115m
48116 4855 48 Cd116
49113 4925 49 In113
49115 4931 49 In115
50112 5025 50 Sn112
50113 5028 50 Sn113
50114 5031 50 Sn114
50115 5034 50 Sn115
50116 5037 50 Sn116
50117 5040 50 Sn117
50118 5043 50 Sn118
50119 5046 50 Sn119
50120 5049 50 Sn120
50122 5055 50 Sn122
50123 5058 50 Sn123
50124 5061 50 Sn124
50125 5064 50 Sn125
50126 5067 50 Sn126
51121 5125 51 Sb121
51123 5131 51 Sb123
51124 5134 51 Sb124
51125 5137 51 Sb125
51126 5140 51 Sb126
52120 5225 52 Te120
52122 5231 52 Te122
52123 5234 52 Te123
52124 5237 52 Te124
52125 5240 52 Te125
52126 5243 52 Te126
52527 5247 52 Te127m
52128 5249 52 Te128
52529 5253 52 Te129m
52130 5255 52 Te130
52132 5261 52 Te132
53127 5325 53 I 127
53129 5331 53 I 129
53130 5334 53 I 130
53131 5337 53 I 131
53135 5349 53 I 135
54123 5422 54 Xe123
54124 5425 54 Xe124
54126 5431 54 Xe126
54128 5437 54 Xe128
54129 5440 54 Xe129
54130 5443 54 Xe130
54131 5446 54 Xe131
54132 5449 54 Xe132
54133 5452 54 Xe133
54134 5455 54 Xe134
54135 5458 54 Xe135
```

```
54136 5461 54 Xe136
55133 5525 55 Cs133
55134 5528 55 Cs134
55135 5531 55 Cs135
55136 5534 55 Cs136
55137 5537 55 Cs137
56130 5625 56 Ba130
56132 5631 56 Ba132
56133 5634 56 Ba133
56134 5637 56 Ba134
56135 5640 56 Ba135
56136 5643 56 Ba136
56137 5646 56 Ba137
56138 5649 56 Ba138
56140 5655 56 Ba140
57138 5725 57 La138
57139 5728 57 La139
57140 5731 57 La140
58136 5825 58 Ce136
58138 5831 58 Ce138
58139 5834 58 Ce139
58140 5837 58 Ce140
58141 5840 58 Ce141
58142 5843 58 Ce142
58143 5846 58 Ce143
58144 5849 58 Ce144
59141 5925 59 Pr141
59142 5928 59 Pr142
59143 5931 59 Pr143
60142 6025 60 Nd142
60143 6028 60 Nd143
60144 6031 60 Nd144
60145 6034 60 Nd145
60146 6037 60 Nd146
60147 6040 60 Nd147
60148 6043 60 Nd148
60150 6049 60 Nd150
61147 6149 61 Pm147
61148 6152 61 Pm148
61548 6153 61 Pm148m
61149 6155 61 Pm149
61151 6161 61 Pm151
62144 6225 62 Sm144
62147 6234 62 Sm147
62148 6237 62 Sm148
62149 6240 62 Sm149
62150 6243 62 Sm150
62151 6246 62 Sm151
62152 6249 62 Sm152
62153 6252 62 Sm153
62154 6255 62 Sm154
63151 6325 63 Eu151
63152 6328 63 Eu152
63153 6331 63 Eu153
63154 6334 63 Eu154
63155 6337 63 Eu155
63156 6340 63 Eu156
63157 6343 63 Eu157
64152 6425 64 Gd152
64153 6428 64 Gd153
64154 6431 64 Gd154
64155 6434 64 Gd155
64156 6437 64 Gd156
64157 6440 64 Gd157
64158 6443 64 Gd158
64160 6449 64 Gd160
65159 6525 65 Tb159
65160 6528 65 Tb160
66156 6625 66 Dy156
66158 6631 66 Dy158
66160 6637 66 Dy160
```

```
66161 6640 66 Dy161
66162 6643 66 Dy162
66163 6646 66 Dy163
66164 6649 66 Dy164
67165 6725 67 Ho165
67566 6729 67 Ho166m
68162 6825 68 Er162
68164 6831 68 Er164
68166 6837 68 Er166
68167 6840 68 Er167
68168 6843 68 Er168
68170 6849 68 Er170
71175 7125 71 Lu175
71176 7128 71 Lu176
72174 7225 72 Hf174
72176 7231 72 Hf176
72177 7234 72 Hf177
72178 7237 72 Hf178
72179 7240 72 Hf179
72180 7243 72 Hf180
73181 7328 73 Ta181
73182 7331 73 Ta182
74182 7431 74 W 182
74183 7434 74 W 183
74184 7437 74 W 184
74186 7443 74 W 186
75185 7525 75 Re185
75187 7531 75 Re187
77191 7725 77 Ir191
77193 7731 77 Ir193
79197 7925 79 Au197
80196 8025 80 Hg196
80198 8031 80 Hg198
80199 8034 80 Hg199
80200 8037 80 Hg200
80201 8040 80 Hg201
80202 8043 80 Hg202
80204 8049 80 Hg204
82204 8225 82 Pb204
82206 8231 82 Pb206
82207 8234 82 Pb207
82208 8237 82 Pb208
83209 8325 83 Bi209
88223 8825 88 Ra223
88224 8828 88 Ra224
88225 8831 88 Ra225
88226 8834 88 Ra226
89225 8925 89 Ac225
89226 8928 89 Ac226
89227 8931 89 Ac227
90227 9025 90 Th227
90228 9028 90 Th228
90229 9031 90 Th229
90230 9034 90 Th230
90232 9040 90 Th232
90233 9043 90 Th233
90234 9046 90 Th234
91231 9131 91 Pa231
91232 9134 91 Pa232
91233 9137 91 Pa233
92232 9219 92 U 232
92233 9222 92 U 233
92234 9225 92 U 234
92235 9228 92 U 235
92236 9231 92 U 236
92237 9234 92 U 237
92238 9237 92 U 238
92239 9240 92 U 239
92240 9243 92 U 240
92241 9246 92 U 241
93235 9340 93 Np235
```

```
93236 9343 93 Np236
93237 9346 93 Np237
93238 9349 93 Np238
93239 9352 93 Np239
94236 9428 94 Pu236
94237 9431 94 Pu237
94238 9434 94 Pu238
94239 9437 94 Pu239
94240 9440 94 Pu240
94241 9443 94 Pu241
94242 9446 94 Pu242
94243 9449 94 Pu243
94244 9452 94 Pu244
94246 9458 94 Pu246
95241 9543 95 Am241
95642 9546 95 Am242
95242 9547 95 Am242m
95243 9549 95 Am243
95244 9552 95 Am244
95644 9553 95 Am244m
96241 9628 96 Cm241
96242 9631 96 Cm242
96243 9634 96 Cm243
96244 9637 96 Cm244
96245 9640 96 Cm245
96246 9643 96 Cm246
96247 9646 96 Cm247
96248 9649 96 Cm248
96249 9652 96 Cm249
96250 9655 96 Cm250
97249 9752 97 Bk249
97250 9755 97 Bk250
98249 9852 98 Cf249
98250 9855 98 Cf250
98251 9858 98 Cf251
98252 9861 98 Cf252
98253 9864 98 Cf253
98254 9867 98 Cf254
99253 9913 99 Es253
99254 9914 99 Es254
99255 9915 99 Es255
00255 9936100 Fm255
```

Appendix C. Checking script "checkace.pl"

```
#! /usr/bin/perl

# checkace.pl
#
# Runs a series of checking codes on an ACE file.

###############################################################################

###############################################################################

use File::Copy;
use FindBin qw($Bin);          # find absolute path

#
# Usage:  perl checkace.pl
#
# Output:


# Main script starts here

my $infile;
my $outfile1;
my $outfile2;
my $tolerance = 0.001;

# Currently this is the only whale input file (whin) which can be used
#   TODO:  change script to be able to user user-specified whin
my $whin_body = <<EOF;          #print everything from here down to EOF
-1.0e-4 1.0e6 1.0e-5             ! low sigma, high sigma, repeat tolerance
34 500                           ! 34 groups  100 MTs
1.390E-04       1.520E-01       4.140E-01       1.130E+00       3.060E+00       8.320E+00
2.260E+01       6.140E+01       1.670E+02       4.540E+02       1.235E+03       3.350E+03
9.120E+03       2.480E+04       6.760E+04       1.840E+05       3.030E+05       5.000E+05
8.230E+05       1.353E+06       1.738E+06       2.232E+06       2.865E+06       3.680E+06
6.070E+06       7.790E+06       1.000E+07       1.200E+07       1.350E+07       1.500E+07
1.700E+07       2.000E+07       2.500E+07       3.000E+07       3.500E+07
48                                 ! 48 energy-flux pairs
1.3900E-04  3.0190E+06   5.00E-04  1.0700E+07   1.00E-03  2.0980E+07
5.00E-03  8.9390E+07     1.00E-02  1.4638E+08   2.50E-02  2.0080E+08
4.00E-02  1.7635E+08     5.00E-02  1.4780E+08   1.00E-01  4.00E+07
1.40E-01  1.1300E+07     1.50E-01  7.60E+06     4.1400E-01  2.7900E+06
1.1300E+00  1.0200E+06   3.0600E+00  3.7700E+05         8.3200E+00  1.3900E+05
2.2600E+01  5.1100E+04   6.1400E+01  1.8800E+04         1.6700E+02  6.9100E+03
4.5400E+02  2.5400E+03   1.2350E+03  9.3500E+02         3.3500E+03  3.4500E+02
9.1200E+03  1.2660E+02   2.4800E+04  4.6500E+01         6.7600E+04  1.7100E+01
1.8400E+05  6.2700E+00   3.0300E+05  3.8800E+00         5.00E+05  3.60E+00
8.2300E+05  2.8700E+00   1.3530E+06  1.7500E+00         1.7380E+06  1.1300E+00
2.2320E+06  7.30E-01     2.8650E+06  4.00E-01   3.6800E+06  2.0500E-01
6.0700E+06  3.90E-02     7.7900E+06  1.6300E-02         1.00E+07  6.50E-03
1.20E+07  7.60E-03       1.30E+07  1.2300E-02   1.3500E+07  2.6400E-02
1.40E+07  1.1400E-01     1.4100E+07  1.1400E-01         1.4200E+07  1.0100E-01
1.4300E+07  6.50E-02     1.4600E+07  1.4900E-02         1.50E+07  4.00E-03
1.60E+07  1.5400E-03     1.70E+07  8.50E-04     4.00E+07  8.50E-04

EOF



foreach $acefile (@ARGV) {

    print "\nPROCESSING FILE $acefile.\n";

    # CHECKND
      print "*   RUNNING CHECKND........\n";
      $outfile1 = $acefile . ".cnd";
```

```
    system("$Bin/checknd $acefile $outfile1");
   &check_cnd($outfile1);

  #CHECKND_NEUT
     print "*   RUNNING CHECKND_NEUT........\n";
     $outfile1 = $acefile . ".cnd_n";
     $outfile2 = $acefile . ".new";
     system("$Bin/checknd_neut $acefile $outfile1 $outfile2");
     &check_cnd_n($outfile1);

  #CHECK_HEAT
     print "*   RUNNING CHECK_HEAT........\n";
     $outfile1 = $acefile . ".cht";
     system("$Bin/check_heat $acefile $outfile1");
     &check_cht($outfile1);

  #CHECK_ISO
     print "*   RUNNING CHECK_ISO........\n";
     $outfile1 = $acefile . ".ciso";
     system("$Bin/check_iso $acefile $outfile1");
     &check_ciso($outfile1);

  #CHECKTHRESH
    print "*   RUNNING CHECKTHRESH........\n";
    $outfile1 = $acefile . ".cthr";
    system("$Bin/checkthresh $acefile $outfile1");
    &check_cthr($outfile1);

  #CHECK0
    print "*   RUNNING CHECK0........\n";
    $outfile1 = $acefile . ".c0";
    $outfile2 = $acefile . ".new3";
    system("$Bin/check0 $acefile $outfile1 $outfile2");
    &check_c0($outfile1);

  #CHECK5
    print "*   RUNNING CHECK5........\n";
    $outfile1 = $acefile . ".c5";
    system("$Bin/check5 $acefile $outfile1");
    &check_c5($outfile1);

  #CHECK61
    print "*   RUNNING CHECK61........\n";
    $outfile1 = $acefile . ".c61";
    $outfile2 = $acefile . ".new2";
    system("$Bin/check61 $acefile $outfile1 $outfile2");
    &check_c61($outfile1);

  #CHECKXS
    print "*   RUNNING CHECKXS........\n";
    $outfile1 = $acefile . ".cxs";
    system("$Bin/checkxs $acefile $outfile1 $tolerance");
    &check_xs($outfile1);

  #CHECK_URES
   print "*   RUNNING CHECK_URES........\n";
   $outfile1 = $acefile . ".cures";
#   copy( $acefile, "tape1" );
   system("$Bin/check_ures $acefile $outfile1");
    &check_cures($outfile1);
#   copy( "tape3", $outfile1 );
#   system("rm tape1 tape3");

#   checklownum
  my $line_count = 0;

  print "\nCHECKING FILE $acefile for low numbers.\n";
  open ( INFILE, $acefile );
  $line = <INFILE>;
  @words = split " ", $line, 2;
  while (<INFILE>) {
```

```
      $line_count++;
      chomp;
      if ( ( /[0-9]\.[0-9]+(E|e)\-([3-9][0-9]+)/ ) ||
           ( /[0-9]\.[0-9]+(E|e)\-([0-9][0-9][0-9]+)/ ) ) {
        print "Line $line_count: $_\n";
      }
    }
  close( INFILE );
  print "DONE CHECKING FILE $acefile.\n\n";

# WHALE
#     $whalenum  = 1;
#     $whalefile = substr ($acefile, 0, index($acefile, "_"));
##     WARNING: erasing current whin
#     open ( WHIN, ">whin" );
#     print "*   RUNNING WHALE........\n";
#     print WHIN "$whalenum \'$whalefile\'   2 \'$acefile\'         ! 1 is ENDF, 2 is ACE  0 is
ignore\n";
#     print WHIN "\'$acefile.wout\'   \'$acefile.cmp\'  \!general output, compare output\n";
#     print WHIN "$tolerance                  ! tolerance for comparing data files 1.0 = 100%
rel. diff.\n";
#     print WHIN $whin_body;
#     close WHIN;
#     system("whale");
#
#   set up MCNP input file for Ho-Ho test
     print "*   Setting up MCNP input file........\n";
     $iso[0] = $words[0];
     for ($i = 0; $i < 4; $i++) {
       $iso[i+1] = chop($iso[i]);
     }
     $mcnpfile = $iso[i-2] . h1;
     if (-e $mcnpfile) {
       $mcnpfile = $iso[i-2] . h2;
     }
     if (-e $mcnpfile) {
       $mcnpfile = $iso[i-2] . h3;
     }
     if (-e $mcnpfile) {
       $mcnpfile = $iso[i-2] . h4;
     }
     if (-e $mcnpfile) {
       $mcnpfile = $iso[i-2] . h5;
     }
     open ( MCIN, ">$mcnpfile" );
     print MCIN "test of ENDF/B-VII data   \n";
     print MCIN "1  1  .1  -1  \n";
     print MCIN "2  0       1  \n";
     print MCIN "  \n";
     print MCIN "1  so 1  \n";
     print MCIN "  \n";
     print MCIN "mode n p \n";
     print MCIN "sdef erg=14.0  \n";
     print MCIN "m1  $words[0] \n";
     print MCIN "imp:n 1 0 \n";
     close MCIN;

   print "\n";
  }

sub check_cnd
{
    my ($outfile) = @_;

   open( OUTFILE, $outfile );

    while (<OUTFILE>) {
       if ( /^error/ ) {
           # Negative pdf
           $nextline = <OUTFILE>;
           print "ERROR found in cnd--\n   $_$nextline";
```

```
        } elsif ( /^[ a-zA-Z\.0-9\+\-]{10}     nxs\(1\)=/ ) {
            # NXS(1) > 2500000
#           print "Length of data block (nxs(1)) is too large.\n";

        } elsif ( /^[ a-zA-Z\.0-9\+\-]{10}[ 0-9\-]{10}  law=[ 0-9]{5}$/ ) {
            #Do nothing--normal law 4 or 44
#           print "1\n";

        } elsif ( /^[ a-zA-Z\.0-9\+\-]{10}[ 0-9\-]{10}  lnw=[ 0-9]{8}$/ ) {
            #Do nothing--normal law 4 or 44
#           print "2\n";

        } elsif ( /^[ a-zA-Z\.0-9\+\-]{10}[ 0-9\-]{10}  ne=[ 0-9]{5}$/ ) {
            # ne > 200
            print "Number of energies (ne) is too large.\n";

        } elsif ( /^[ a-zA-Z\.0-9\+\-]{10}[ 0-9\-]{10} einc=[ eE\+\-\.0-9]{1,20}  all discrete[
0-9\-]{5}$/ ) {
            #Do nothing--all discrete
#           print "3\n";

        } elsif ( /^[ a-zA-Z\.0-9\+\-]{10}[ 0-9\-]{10}  ein=[ eE\+\-\.0-9]{20}  eprime=[ eE\+\-
\.0-9]{20}$/ ) {
            # eprime < 0
             print "eprime <= 0.\n";

        } elsif ( /^                    at first energy, nd=[ 0-9\-]{5}$/ ) {
             #Do nothing--there are discrete photon lines  ( j==1 (1st energy?), nd(j) != 0 )
#           print "4\n";

        } elsif ( /^[ a-zA-Z\.0-9\+\-]{10}[ 0-9\-]{10}  [ eE\+\-\.0-9]{20}  [ \-0-9]{3}  [ eE\+\-
\.0-9]{20}  [ \-0-9]{3}$/ ) {
            # nd(j+1) != nd(j)
            print "Unequal number of discrete photons at adjacent energies.\n";

        } elsif ( /^[ a-zA-Z\.0-9\+\-]{10}[ 0-9\-]{10} ein\(1\)=[ eE\+\-\.0-9]{20} edisc=[ eE\+\-
\.0-9]{20} ein\(2\)=[ eE\+\-\.0-9]{20} edisc=[ eE\+\-\.0-9]{20}$/ ) {
            # e1 at e(j) != e2 at e(j+1)
            print "e1 != e2.\n";
        } else {
            print "UNKNOWN message: $_\n";
        }

    }
    close OUTFILE;

}


sub check_cnd_n
{
    my ($outfile) = @_;

    open( OUTFILE, $outfile ) | "Unable to open file $outfile.\n";

    while (<OUTFILE>) {
        if ( /^error/ ) {
            # Negative pdf
            $nextline = <OUTFILE>;
            print "ERROR found in cnd_n--\n  $_$nextline";

        } elsif ( /^[ a-zA-Z\.0-9\+\-]{10}     nxs\(1\)=/ ) {
            # NXS(1) > 2500000
            print "Length of data block (nxs(1)) is too large.\n";

        } elsif ( /Secondary energies are in the (\w{2,3}) frame/ ) {
            #Energy > Emax
            $nextline = <OUTFILE>;
            chomp;
```

```
            chomp $nextline;
            print "ERROR found in cnd_n--\n  $nextline ($1 frame)";

        } elsif ( /^          skipping correction - fissionable - nxs\(2\)=[ 0-9]{7}$/ ) {
            #For last energy > emax, the isotope is fissionable (i.e. nxs(2)>90000)
            print " (Fis.-UNCHANGED)\n";

        } elsif ( /^     changing energy from[ eE\+\-\.0-9]{20} to[ eE\+\-\.0-9]{20}$/ ) {
            #For last energy > emax, the isotope is not fissionable
            print " (Not fis.-CHANGED)\n";

        } elsif ( /^          bad news - another eprime needs to be changed$/ ) {
            #For last energy > emax, the isotope is not fissionable
            $nextline = <OUTFILE>;
            print "Another bad eprime value -- library has been changed\n";

        } elsif ( /^[ a-zA-Z\.0-9\+\-]{10}[ 0-9\-]{10}  law=[ 0-9\-]{5}$/ ) {
            #Do nothing--normal law 4 or 44
#           print "1\n";

        } elsif ( /^[ a-zA-Z\.0-9\+\-]{10}[ 0-9\-]{10}  lnw=[ 0-9]{8}$/ ) {
            #Do nothing--normal law 4 or 44
#           print "2\n";

        } elsif ( /^[ a-zA-Z\.0-9\+\-]{10}[ 0-9\-]{10}  ne=[ 0-9]{5}$/ ) {
            # ne > 200
            print "Number of energies (ne) is too large.\n";

        } elsif ( /^[ a-zA-Z\.0-9\+\-]{10}[ 0-9\-]{10} einc=[ eE\+\-\.0-9]{20} iint=[ 0-9]{5}$/ )
{
            # iint != 1 and iint !=2
            print "Bad interpolation scheme.\n";

        } elsif ( /^[ a-zA-Z\.0-9\+\-]{10}[ 0-9\-]{10} einc=[ eE\+\-\.0-9]{20} nsec\>1000[ 0-
9]{6}$/ ) {
            # nsec > 1000
            print "Checknd_neut stopped because nsec > 1000.\n";

        } elsif ( /^          renorming distribution$/ ) {
            # Do nothing -- eprimes were changed so the distribution was renormalized
#           print "2a\n";

        } elsif ( /^          renorm factor=[ eE\+\-\.0-9]{14}$/ ) {
            # Do nothing -- eprimes were changed so the distribution was renormalized
#           print "2b\n";

        } elsif ( /^[ 0-9\-]{5} replacing[ eE\+\-\.0-9]{14} with[ eE\+\-\.0-9]{14}$/ ) {
            # Do nothing -- renorming info
#           print "2c\n";

        } elsif ( /^[ a-zA-Z\.0-9\+\-]{10}[ 0-9\-]{10} einc=[ eE\+\-\.0-9]{20}  all discrete[ 0-
9\-]{5}$/ ) {
            #Do nothing--all discrete
#           print "3\n";

        } elsif ( /^[ a-zA-Z\.0-9\+\-]{10}[ 0-9\-]{10}  ein=[ eE\+\-\.0-9]{20}  eprime=[ eE\+\-
\.0-9]{20}$/ ) {
            # eprime < 0
             print "eprime <= 0.\n";

        } elsif ( /^                    at first energy, nd=[ 0-9\-]{5}$/ ) {
             #Do nothing--there are discrete photon lines  ( j==1 (1st energy?), nd(j) != 0 )
            print "4\n";

        } elsif ( /^[ a-zA-Z\.0-9\+\-]{10}[ 0-9\-]{10}  [ eE\+\-\.0-9]{20}  [ \-0-9]{3}  [ eE\+\-
\.0-9]{20}  [ \-0-9]{3}$/ ) {
            # nd(j+1) != nd(j)
            print "Unequal number of discrete photons at adjacent energies.\n";

        } elsif ( /^[ a-zA-Z\.0-9\+\-]{10}[ 0-9\-]{10} ein\(1\)=[ eE\+\-\.0-9]{20} edisc=[ eE\+\-
\.0-9]{20} ein\(2\)=[ eE\+\-\.0-9]{20} edisc=[ eE\+\-\.0-9]{20}$/ ) {
```

```
            # e1 at e(j) != e2 at e(j+1)
            print "e1 != e2.\n";
        } elsif ( /^warning:/ ) {
            $nextline = <OUTFILE>;
            $nextline = <OUTFILE>;
        } else {
            print "UNKNOWN message: $_ \n";
        }

    }
    close OUTFILE;

}


sub check_cthr
{
    my ($outfile) = @_;

    open( OUTFILE, $outfile );

    $mult_flag = 0;
    while (<OUTFILE>) {
        if ( /^          biggest change=([ eE\.0-9\+\-]{12})$/ ) {
            # Report on the biggest change -- do nothing
#           print "0:  $1\n";
            if ( $1 != 0.0 ) {
                if ($mult_flag == 1) {
                    print "CHECKTHRESH error: MULTIPLE energies found below threshold. (biggest
change: $1)\n"
                } else {
                    print "CHECKTHRESH error: Energies found below threshold. (biggest change:
$1)\n";
                }
            }
            $mult_flag = 0;

        } elsif ( /^[ a-zA-Z\.0-9\+\-]{10}     nxs\(1\)=/ ) {
            # NXS(1) > 2500000
            print "Length of data block (nxs(1)) is too large.\n";

        } elsif ( /^          big trouble \-\- next energy in grid =[ eE\.0-9\+\-]{20}$/ ) {
            # Next energy is below threshold too
            #  Write this out???? -- the error's already been flagged
#           print "Multiple energies found below threshold.\n";
            $mult_flag = 1;

        } elsif  ( /^[ a-zA-Z\.0-9\+\-]{10} mt=[ 0-9]{3} q=[ \.0-9\+\-]{9} egiven=[ eE\.0-9\+\-
]{20} should be [ eE\.0-9\+\-]{20} diff=[ eE\.0-9\+\-]{8} law=[ 0-9]{2} line=[ 0-9]{7}$/ ) {
            # eact < emin
#           print "2  $1\n";

        } else {
            print "UNKNOWN message: $_\n";
        }
    }
    close OUTFILE;

 }

sub check_c0
{
    my ($outfile) = @_;

    open( OUTFILE, $outfile );

    while (<OUTFILE>) {
        if ( /trouble -- [ a-zA-Z\.0-9\+\-]{50} / ) {
            # CHECK0 error found
            print "ERROR found in CHECK0--\n  $_ \n";
```

```
        }
        elsif (/^zaid/) {
        }
        else {
            print "UNKNOWN message: $_\n";
        }
    }
    close OUTFILE;

}

sub check_c5
{
    my ($outfile) = @_;

    open( OUTFILE, $outfile );

    while (<OUTFILE>) {
       if ( /^[ a-zA-Z\.0-9\+\-]{10}  MT= [ \-0-9]{5} xs thresh=[ eE\+\-\.0-9]{20} first
energy=[ eE\+\-\.0-9]{20} first mult=[ eE\+\-\.0-9]{20}$/ ) {
           # CHECK5 error found
           print "CHECK5 error found.\n";

       } elsif ( /^[ a-zA-Z\.0-9\+\-]{10}     nxs\(1\)=/ ) {
           # NXS(1) > 2500000
           print "Length of data block (nxs(1)) is too large.\n";

       } else {
           print "UNKNOWN message: $_\n";
       }
    }
    close OUTFILE;

}


sub check_c61
{
    my ($outfile) = @_;

    open( OUTFILE, $outfile );

    while (<OUTFILE>) {
       if ( /^[ a-zA-Z\.0-9\+\-]{10}  MT= [ \-0-9]{5} xs thresh=[ eE\+\-\.0-9]{20} first
energy=[ eE\+\-\.0-9]{20} first mult=[ eE\+\-\.0-9]{20}$/ ) {
           # CHECK5 error found
           print "CHECK5 error found.\n";



       } elsif ( /^[ a-zA-Z\.0-9\+\-]{10}     nxs\(1\)=/ ) {
           # NXS(1) > 2500000
           print "Length of data block (nxs(1)) is too large.\n";

       } elsif ( /^[ a-zA-Z\.0-9\+\-]{10}[ 0-9\-]{10}  ne=[ 0-9]{5}$/ ) {
           # ne > 200
           print "Number of energies (ne) is too large.\n";

       } elsif ( /^[ a-zA-Z\.0-9\+\-]{10}[ 0-9\-]{10} einc=[ eE\+\-\.0-9]{20} iint=[ 0-9]{5}$/ )
{
           # iint != 1 and iint !=2
           print "Bad interpolation scheme.\n";

       } elsif ( /^error:[ a-zA-Z\.0-9\+\-]{10}[ 0-9\-]{10} einc=[ eE\+\-\.0-9]{20} negative
pdf$/ ) {
           # Negative pdf
           $nextline = <OUTFILE>;
           print "Neg. pdf found in check61.\n";

# line 122 eprimemax < einc
       } elsif ( /^          skipping correction - fissionable - nxs\(2\)=[ 0-9]{7}$/ ) {
```

```
            #For last energy > emax, the isotope is fissionable (i.e. nxs(2)>90000)
            print " (Fis.-UNCHANGED)\n";

        } elsif ( /^     changing energy from[ eE\+\-\.0-9]{20} to[ eE\+\-\.0-9]{20}$/ ) {
            #For last energy > emax, the isotope is not fissionable
            print " (Not fis.-CHANGED)\n";

        } elsif ( /^          bad news - another eprime needs to be changed$/ ) {
            #For last energy > emax, the isotope is not fissionable
            $nextline = <OUTFILE>;
            print "Another bad eprime value -- library has been changed\n";

        } elsif ( /^[ a-zA-Z\.0-9\+\-]{10}[ 0-9\-]{10} einc=[ eE\+\-\.0-9]{20} nsec\>1000[ 0-
9]{6}$/ ) {
            # nsec > 1000
            print "Checknd_neut stopped because nsec > 1000.\n";

        } elsif ( /^[ a-zA-Z\.0-9\+\-]{10}[ 0-9\-]{10}  ein=[ eE\+\-\.0-9]{20}  eprime=[ eE\+\-
\.0-9]{20}$/ ) {
            # eprime < 0
             print "eprime <= 0.\n";

        } elsif ( /^[ a-zA-Z\.0-9\+\-]{10}[ 0-9\-]{10} ein\(1\)=[ eE\+\-\.0-9]{20} edisc=[ eE\+\-
\.0-9]{20} ein\(2\)=[ eE\+\-\.0-9]{20} edisc=[ eE\+\-\.0-9]{20}$/ ) {
            # e1 at e(j) != e2 at e(j+1)
            print "e1 != e2.\n";

        } elsif ( /^          renorming distribution$/ ) {
            # Do nothing -- eprimes were changed so the distribution was renormalized
#           print "2a\n";

        } elsif ( /^          renorm factor=[ eE\+\-\.0-9]{14}$/ ) {
            # Do nothing -- eprimes were changed so the distribution was renormalized
#           print "2b\n";

        } elsif ( /^[ 0-9\-]{5} replacing[ eE\+\-\.0-9]{14} with[ eE\+\-\.0-9]{14}$/ ) {
            # Do nothing -- renorming info
#           print "2c\n";


        } elsif ( /^[ a-zA-Z\.0-9\+\-]{10}[ 0-9\-]{10}  law=[ 0-9\-]{5}$/ ) {
            #Do nothing--normal law 61
#           print "1\n";

        } elsif ( /^[ a-zA-Z\.0-9\+\-]{10}[ 0-9\-]{10} einc=[ eE\+\-\.0-9]{20}  all discrete[ 0-
9\-]{5}$/ ) {
            #Do nothing--all discrete
#           print "3\n";

        } elsif ( /^                    at first energy, nd=[ 0-9\-]{5}$/ ) {
             #Do nothing--there are discrete photon lines  ( j==1 (1st energy?), nd(j) != 0 )
#           print "4\n";

        } elsif ( /^[ a-zA-Z\.0-9\+\-]{10}[ 0-9\-]{10}  [ eE\+\-\.0-9]{20}  [ \-0-9]{3}  [ eE\+\-
\.0-9]{20}  [ \-0-9]{3}$/ ) {
            # nd(j+1) != nd(j)
            print "Unequal number of discrete photons at adjacent energies.\n";

        } elsif ( /^[ a-zA-Z\.0-9\+\-]{10}[ 0-9\-]{10}  lnw=[ 0-9]{8}$/ ) {
            #Do nothing--normal law 4 or 44
#           print "2\n";

        } else {
            print "UNKNOWN message: $_\n";
        }
    }
    close OUTFILE;

 }

sub check_cht
```

```
{
    my ($outfile) = @_;

    open( OUTFILE, $outfile );

    $blank_line = <OUTFILE>;
    $iflag = 0;
    while (<OUTFILE>) {


        if ( /^[ a-zA-Z\.0-9\+\-]{10}     nxs\(1\)=/ ) {
            # NXS(1) > 2500000
            print "Length of data block (nxs(1)) is too large.\n";
        }
        elsif ( /^[ a-zA-Z\.0-9\+\-]{10}     nxs\(3\)=/ ) {
            # NXS(3) > 250000
            print "Length of data block (nxs(3)) is too large.\n";
        }
        elsif (/^\s*[0-9]{5}.[0-9]{2}c numzero =    0 numneg=    0/) {
            # not a CHECKHT error
        }
        elsif (/^\s*[0-9]{4}.[0-9]{2}c numzero =    0 numneg=    0/) {
            # not a CHECKHT error
        }
        elsif (/^\s*[0-9]{5}.[0-9]{2}c numzero =/) {
            print "ERROR found in CHECKHT numzero/negs \n $_ ";
            # CHECKHT error found
        }
        elsif (/^\s*[0-9]{4}.[0-9]{2}c numzero =/) {
            print "ERROR found in CHECKHT numzero/negs \n $_ ";
            # CHECKHT error found
        }
        elsif ( /^\s*ipt=[ 0-9]{3} numzero =    0 numneg=    0/ ) {
            # not a CHECKHT error
        }
        elsif ( /^\s*ipt=[ 0-9]{3} numzero=[ 0-9]{5}/ ) {
            # CHECKHT error found
            print "ERROR found in CHECKHT ipt --\n  $_ ";
        }
        elsif (/^\s*num comp=/ ) {
            # CHECKHT error found
            print "ERROR found in CHECKHT total --\n  $_ ";
        }
        elsif (/^\s*no partial heating/) {
            # not a CHECKHT error
        }
        elsif (/^ $/) {
            # not a CHECKHT error
        }
        else {
            print "UNKNOWN message: $_\n";
        }
    }
    close OUTFILE;

 }

sub check_ciso
{
    my ($outfile) = @_;

    open( OUTFILE, $outfile );

    while (<OUTFILE>) {
        if (/^\s*[0-9]{5}.[0-9]{2}c mt=   18/) {
            # not a CHECKISO error
        }
        elsif (/^\s*for reaction/) {
            # not a CHECKISO error
        }
       # CHECK_ISO message found
```

```
        else {
           print "UNKNOWN CHECKISO ERROR -- $_ \n";
        }
    }
    close OUTFILE;

 }

sub check_xs
{
    my ($outfile) = @_;

    open( OUTFILE, $outfile );

    while (<OUTFILE>) {
      if (/^\s*[0-9]{5}.[0-9]{2}c/) {
       # not a CHECKXS error
      }
      elsif (/^\s*[0-9]{4}.[0-9]{2}c/) {
       # not a CHECKXS error
      }
      elsif (/^\s*END OF PROCESSING ZAID/) {
       # not a CHECKXS error
      }
      elsif (/^\s*No Photon Production data for this ZAID/) {
       # not a CHECKXS error
      }
      elsif (/^\s*Index      Energy/) {
       # not a CHECKXS error
      }
      elsif (/^\s*Checking/) {
       # not a CHECKXS error
      }
      elsif (/^$/) {
       # not a CHECKHT error
      }
      else {
       # CHECK_XS message found
         print "ERROR found in CHECK_XS--\n  $_  ";
      }
    }
    close OUTFILE;

 }

sub check_cures
{
    my ($outfile) = @_;

    open( OUTFILE, $outfile );

    $blank_line = <OUTFILE>;
    while (<OUTFILE>) {

        if ( /^[ a-zA-Z\.0-9\+\-]{10}     nxs\(1\)=/ ) {
            # NXS(1) > 2500000
            print "Length of data block (nxs(1)) is too large.\n";
        }
        elsif ( /^\s*fatal --/ ) {
            # CHECK_CURES error found
            print "ERROR found in CHECK_CURES--\n  $_ ";
        }
        elsif (/^\s*possible problem/) {
            # CHECK_CURES error found
            print "ERROR found in CHECK_CURES with the energy grid.\n";
        }
        elsif ( /^\s*warning: at i=/ ) {
            # CHECK_CURES error found
            print "ERROR found in CHECK_CURES--\n  $_ ";
            print "The infinite total does not equal the sum of infinite partials with 1 part in
a million tolerance \n";
```

```
        }
        elsif ( /^\s*bad --/ ) {
            # CHECK_CURES error found
            print "ERROR found in CHECK_CURES--\n  $_ ";
        }
        elsif (/^\s*looks like non-physical heating value/) {
            # CHECK_CURES error found
            print "ERROR found in CHECK_CURES in that a heating value looks too big.\n";
        }
        elsif ( /^\s*row sum,tot disagree/ ) {
            # CHECK_CURES error found
            print "ERROR found in CHECK_CURES--\n  $_ ";
        }
        elsif ( /^\s*capture:/ |  /^\s*elastic:/ |  /^\s*fission:/ |  /^\s*sum:/ |  /^\s*heat:/
) {
            # CHECK_CURES error found
            print "ERROR found in CHECK_CURES--\n  $_ ";
            print "pt averages don't equal infinitely dilute values (within a tolerance of 1e-4)
\n";
        }
    }
    close OUTFILE;
```

Appendix D. Evaluation Changes

Evaluations were modified for the following isotopes: ^{1}H, ^{45}Sc, ^{89}Y, ^{96}Zr, ^{97}Mo, ^{233}U, ^{242m}Am, and ^{242}Am (ground state).

H1: A new value for the energy of the photon from radiative capture was modified from 2.2233 MeV from 2.2246. The new value is the actual energy of the photon from thermal capture. The previous evaluations gave the total energy and did not take into account the recoil of the nucleus (hence impacting heating as well). This was found in ENDF66 and did not get changed in ENDF/B-VII.

```
Line 399

0.000000+0 2.224631+6          0          2          1          2 125 6102    2


Became:
0.000000+0 2.223300+6          0          2          1          2 125 6102    2
```

Sc45: In ENDF66, it was found that this evaluation had to be modified because File 13, MT=3 did not have the secondary distribution for photons beginning with a zero point at the threshold. Additionally, the evaluation had the incorrect reference frame specified for angular distributions of (n,2n) and (n,n*)a, (n,n*)p, and (n,n*c) as found by CONSIS/ACER. Both of these problems still existed in the ENDF/B-VII evaluation, so the following changes were made:

```
Reaction 16 Line 802
0.000000+0 4.456790+1          1          2          0          02125 4 16    1

Was changed to:
0.000000+0 4.456790+1          1          1          0          02125 4 16    1

Reaction 22 Line 805
0.000000+0 4.456790+1          1          2          0          02125 4 22    1

Was changed to:
0.000000+0 4.456790+1          1          1          0          02125 4 22    1

Reaction 28 Line 808
0.000000+0 0.000000+0          1          2          0          02125 4 28    1

Was changed to:
0.000000+0 0.000000+0          1          1          0          02125 4 28    1

Reaction 91 Line 853
0.000000+0 4.456790+1          1          2          0          02125 4 91

Was changed to:
0.000000+0 4.456790+1          1          1          0          02125 4 91    1
```

Reaction MT13 MF 3 (lines 2975 to 2982)

```
0.000000+0 0.000000+0          0          1          1          17212513  3    2
        17          2                                            212513  3    3
1.000000+6 9.117200-1 2.000000+6 2.750500+0 3.000000+6 3.343000+0212513  3    4
3.500000+6 3.320100+0 4.000000+6 3.295200+0 5.000000+6 3.146800+0212513  3    5
6.000000+6 3.185700+0 8.000000+6 3.675400+0 1.000000+7 4.289900+0212513  3    6
1.100000+7 4.376100+0 1.200000+7 4.165900+0 1.300000+7 3.943300+0212513  3    7
1.400000+7 3.733600+0 1.500000+7 3.444500+0 1.600000+7 3.409100+0212513  3    8
1.800000+7 3.450100+0 2.000000+7 3.299400+0                    212513  3    9
```

Was changed to:

```
0.000000+0 0.000000+0          0          1          1         18212513  3    2
        18          2                                                212513  3    3
9.999990+5 0.000000+0 1.000001+6 9.117200-1 2.000000+6 2.750500+0212513  3    4
3.000000+6 3.343000+0 3.500000+6 3.320100+0 4.000000+6 3.295200+0212513  3    5
5.000000+6 3.146800+0 6.000000+6 3.185700+0 8.000000+6 3.675400+0212513  3    6
1.000000+7 4.289900+0 1.100000+7 4.376100+0 1.200000+7 4.165900+0212513  3    7
1.300000+7 3.943300+0 1.400000+7 3.733600+0 1.500000+7 3.444500+0212513  3    8
1.600000+7 3.409100+0 1.800000+7 3.450100+0 2.000000+7 3.299400+0212513  3    9
```

Y89: Reaction 91 was changed from 35 lines with numerous negative cross sections to only 14 lines with values that were the difference of MT 4 and all MT 51-90 reactions (I summed this on a spreadsheet)

Previous Lines 1956 to 1989

```
0.000000+0-9.788940+5          0          0          1          943925 3 91    1
        94          2                                                3925 3 91    2
9.900000+5 0.000000+0 1.000000+6 5.474100-3 1.100000+6 7.030000-53925 3 91    3
1.200000+6 1.040000-4 1.500000+6 1.053600-2 1.524360+6 4.168522-23925 3 91    4
1.700000+6-1.278960-1 1.764360+6-6.510780-2 2.000000+6-2.126790-13925 3 91    5
2.200000+6-2.527910-1 2.247600+6-2.298065-1 2.500000+6-2.568390-13925 3 91    6
2.558570+6-2.268702-1 2.595355+6-2.172159-1 2.595360+6-2.172148-13925 3 91    7
2.651788+6-2.026715-1 2.651790+6-2.026712-1 2.904380+6-1.663576-13925 3 91    8
2.913888+6-1.678272-1 2.913890+6-1.678286-1 2.927540+6-1.765479-13925 3 91    9
3.000000+6-2.282748-1 3.102400+6-1.879899-1 3.142310+6-1.913324-13925 3 91   10
3.174510+6-2.086664-1 3.200000+6-2.355770-1 3.284340+6-2.257808-13925 3 91   11
3.381008+6-2.229665-1 3.381010+6-2.229665-1 3.449190+6-2.242318-13925 3 91   12
3.490355+6-2.338753-1 3.490360+6-2.338775-1 3.500000+6-2.379197-13925 3 91   13
3.543350+6-2.088434-1 3.555485+6-2.029732-1 3.555490+6-2.029722-13925 3 91   14
3.597456+6-1.912481-1 3.597460+6-1.912480-1 3.600389+6-1.913103-13925 3 91   15
3.600390+6-1.913104-1 3.662080+6-1.977303-1 3.671487+6-2.006924-13925 3 91   16
3.671490+6-2.006934-1 3.700000+6-2.106652-1 3.757450+6-1.779832-13925 3 91   17
3.790219+6-1.645979-1 3.790220+6-1.645976-1 3.795275+6-1.620672-13925 3 91   18
3.795280+6-1.620653-1 3.891758+6-1.329360-1 3.891760+6-1.329358-13925 3 91   19
3.906119+6-1.301399-1 3.906120+6-1.301398-1 3.968519+6-1.328377-13925 3 91   20
3.968520+6-1.328379-1 4.000000+6-1.389370-1 4.022019+6-1.171123-13925 3 91   21
4.022020+6-1.171114-1 4.036886+6-1.033714-1 4.036890+6-1.033681-13925 3 91   22
4.060850+6-8.353687-2 4.068440+6-7.711653-2 4.500000+6 2.379666-13925 3 91   23
5.000000+6 7.981554-1 5.500000+6 1.221206+0 6.000000+6 1.432097+03925 3 91   24
6.500000+6 1.553025+0 7.000000+6 1.620094+0 7.500000+6 1.657880+03925 3 91   25
8.000000+6 1.680306+0 8.500000+6 1.691661+0 9.000000+6 1.694888+03925 3 91   26
9.500000+6 1.692033+0 1.000000+7 1.685082+0 1.050000+7 1.668128+03925 3 91   27
1.100000+7 1.643995+0 1.150000+7 1.612189+0 1.200000+7 1.489110+03925 3 91   28
1.250000+7 1.271664+0 1.300000+7 1.048079+0 1.350000+7 8.560107-13925 3 91   29
1.400000+7 6.996135-1 1.450000+7 5.637885-1 1.500000+7 4.580743-13925 3 91   30
1.550000+7 3.779990-1 1.600000+7 3.196506-1 1.650000+7 2.776628-13925 3 91   31
1.700000+7 2.445734-1 1.750000+7 2.187781-1 1.800000+7 1.969604-13925 3 91   32
1.850000+7 1.802930-1 1.900000+7 1.660907-1 1.950000+7 1.530276-13925 3 91   33
2.000000+7 1.424113-1                                                3925 3 91   34
```

Became 1956 to 1968:

```
0.000000+0-9.788940+5          0          0          1          333925 3 91    1
        33          2                                                3925 3 91    2
4.068440+6 0.000000+0 4.500000+6 2.379666-1 5.000000+6 7.981554-13925 3 91    3
5.500000+6 1.221206+0 6.000000+6 1.432097+0 6.500000+6 1.553025+03925 3 91    4
7.000000+6 1.620094+0 7.500000+6 1.657880+0 8.000000+6 1.680306+03925 3 91    5
8.500000+6 1.691661+0 9.000000+6 1.694888+0 9.500000+6 1.692033+03925 3 91    6
1.000000+7 1.685082+0 1.050000+7 1.668128+0 1.100000+7 1.643995+03925 3 91    7
1.150000+7 1.612189+0 1.200000+7 1.489110+0 1.250000+7 1.271664+03925 3 91    8
1.300000+7 1.048079+0 1.350000+7 8.560107-1 1.400000+7 6.996135-13925 3 91    9
1.450000+7 5.637885-1 1.500000+7 4.580743-1 1.550000+7 3.779990-13925 3 91   10
1.600000+7 3.196506-1 1.650000+7 2.776628-1 1.700000+7 2.445734-13925 3 91   11
1.750000+7 2.187781-1 1.800000+7 1.969604-1 1.850000+7 1.802930-13925 3 91   12
```

```
1.900000+7 1.660907-1 1.950000+7 1.530276-1 2.000000+7 1.424113-13925 3 91   13
```

In addition, Line 281 which was:

```
                              3          91          35          13925 1451  280
```

Became:

```
                              3          91          14          13925 1451  280
```

Zr96 and Mo97:

These evaluations contained the following errors from CONSIS:

```
checking energy distributions
consis: bad law44 kalbach r for (n,xd) at   1.800000E+01 -> 6.609760E+00
consis: bad law44 kalbach r for (n,xd) at   1.900000E+01 -> 7.588980E+00
consis: bad law44 kalbach r for (n,xd) at   1.900000E+01 -> 8.568210E+00
consis: bad law44 kalbach r for (n,xd) at   1.900000E+01 -> 9.057820E+00
consis: bad law44 kalbach r for (n,xd) at   2.000000E+01 -> 8.078600E+00
consis: bad law44 kalbach r for (n,xd) at   2.000000E+01 -> 8.568210E+00
consis: bad law44 kalbach r for (n,xd) at   2.000000E+01 -> 9.057820E+00
consis: bad law44 kalbach r for (n,xd) at   2.000000E+01 -> 9.547430E+00
consis: bad law44 kalbach r for (n,xd) at   2.000000E+01 -> 1.003700E+01
```

The value r is supposed to be in the range 0-1, as a sort of fraction. Values outside 0-1 such as what NJOY found for Zr96 are therefore illegal. The reality is that the exponents on values of r equal to 0.9999… were e+1, so the numbers were 9.999… instead of 0.9999….

Zr96 was modified as follows:

```
Line 3847
6.120150+6 2.234700-7 9.999660-1 6.609760+6 2.919190-7 0.999998+14043 6204   58

Became:
6.120150+6 2.234700-7 9.999660-1 6.609760+6 2.919190-7 0.999998+04043 6204   58

Lines 3858 to 3860
7.099370+6 2.264410-7 9.999860-1 7.588980+6 2.701220-7 0.999999+14043 6204   69
8.078600+6 3.152390-7 0.999997+0 8.568210+6 2.419700-7 0.999999+14043 6204   70
9.057820+6 3.805770-7 0.999999+1 9.607040+6 0.000000+0 0.000000+04043 6204   71

Became:
7.099370+6 2.264410-7 9.999860-1 7.588980+6 2.701220-7 0.999999+04043 6204   69
8.078600+6 3.152390-7 0.999997+0 8.568210+6 2.419700-7 0.999999+04043 6204   70
9.057820+6 3.805770-7 0.999999+0 9.607040+6 0.000000+0 0.000000+04043 6204   71

Lines 3870 to 3872
8.078600+6 2.233940-7 0.999999+1 8.568210+6 2.545820-7 0.999999+14043 6204   81
9.057820+6 1.939780-7 0.999999+1 9.547430+6 2.137780-7 0.999999+14043 6204   82
1.003700+7 3.377100-7 0.999999+1 1.059660+7 0.000000+0 0.000000+04043 6204   83

Became:
8.078600+6 2.233940-7 0.999999+0 8.568210+6 2.545820-7 0.999999+04043 6204   81
9.057820+6 1.939780-7 0.999999+0 9.547430+6 2.137780-7 0.999999+04043 6204   82
1.003700+7 3.377100-7 0.999999+0 1.059660+7 0.000000+0 0.000000+04043 6204   83
```

Mo97 was modified as follows:

```
Line 6116
1.573730+7 1.558500-8 0.999998+1 1.600180+7 0.000000+0 0.000000+04240 6203  331

Became:
1.573730+7 1.558500-8 0.999998+0 1.600180+7 0.000000+0 0.000000+04240 6203  331

Line 6148
1.632600+7 1.738630-8 0.999998+1 1.660050+7 1.379760-8 1.000000+04240 6203  363

Became:
```

```
1.632600+7 1.738630-8 0.999998+0 1.660050+7 1.379760-8 1.000000+04240 6203  363

Lines 6180 and 6181
1.634620+7 1.901120-8 9.999910-1 1.663040+7 1.344000-8 0.999999+14240 6203  395
1.691470+7 7.624100-9 0.999996+0 1.719900+7 1.510560-8 0.999999+14240 6203  396

Became
1.634620+7 1.901120-8 9.999910-1 1.663040+7 1.344000-8 0.999999+04240 6203  395
1.691470+7 7.624100-9 0.999996+0 1.719900+7 1.510560-8 0.999999+04240 6203  396
```

U-233: delayed-neutron yield

It turns out that there is an error in the exponent for the delayed- neutron yields for U-233 above 9 MeV. The effect of this is trivial but still important.

```
Line 599
1.400000+7 4.700000-2 1.500000+7 4.200000-2 3.000000+7 4.200000-29222 1455    6

Became:
1.400000+7 4.700000-3 1.500000+7 4.200000-3 3.000000+7 4.200000-39222 1455    6
```

Am242ground: This looks like it is caused by an omission (error) in the endf file - namely the 242Am file does not have an angular distribution (no file 4 nor file 6, mt=18) for fission. NJOY fills in the tyr array after reading the angular distribution data so that it knows whether to use a + or - value; if there's no data for a given mt, then the tyr value for that reaction simply retains the zero value that the array was initialized with. Skip tested this by stealing the isotropic mf4, mt18 data from the 242mAm file and inserting it into the 242Am file. This modification was also implemented in my processing.

```
Line 5
0.000000+0 0.000000+0          0          0        118         849546 1451    4

Became:
0.000000+0 0.000000+0          0          0        118         859546 1451    4

Line 124
                               1        451        206          09546 1451  123

Became:
                               1        451        207          09546 1451  123


Line 171 was added:
                               4         18          2           09546 1451  169

And lines 1477 to 1479 were added:
9.524200+4 2.399801+2          0          0          0           09546 4 18    0
0.000000+0 2.399801+2          1          1          0           09546 4 18    1
0.000000+0 0.000000+0          0          0          0           09546 4  099999
```

Am242m: The inelastic cross sections (MT 4, 51, 52, 53, and 54) contained values of zero between 50 and 65 keV, whereas values above and below were non-zero. The cross sections below 100 keV were smoothed out by the evaluator in this modification.

```
Lines 927 to 933:
1.500000+4 0.000000+0 2.000000+4 0.000000+0 2.500000+4 1.903490-39547 3  4    4

2.731335+4 1.887096-3 3.000000+4 1.967976-3 3.500000+4 1.723383-39547 3  4    5

4.000000+4 1.124790-3 4.500000+4 1.330016-4 5.000000+4 0.000000+09547 3  4    6
```

```
5.061002+4 0.000000+0 5.500000+4 0.000000+0 6.000000+4 0.000000+09547 3  4    7
6.500000+4 0.000000+0 6.567252+4 3.153566-3 7.500000+4 7.644508-29547 3  4    8
8.000000+4 6.936488-2 8.500000+4 6.248683-2 9.000000+4 5.770698-29547 3  4    9
9.500000+4 5.654312-2 9.981420+4 5.997318-2 1.000000+5 6.161738-29547 3  4   10
```

Became:

```
1.500000+4 0.000000+0 2.000000+4 0.000000+0 2.500000+4 2.003490-39547 3  4    4
2.731335+4 2.202442-3 3.000000+4 2.524756-3 3.500000+4 3.401936-39547 3  4    5
4.000000+4 4.666732-3 4.500000+4 6.350844-3 5.000000+4 8.485973-39547 3  4    6
5.061002+4 8.778871-3 5.500000+4 1.110382-2 6.000000+4 1.423608-29547 3  4    7
6.500000+4 1.791447-2 6.567252+4 1.845262-2 7.500000+4 2.703640-29547 3  4    8
8.000000+4 3.254335-2 8.500000+4 3.872322-2 9.000000+4 4.560772-29547 3  4    9
9.500000+4 5.322854-2 9.981420+4 6.129154-2 1.000000+5 6.161738-29547 3  4   10
```

Lines 1030 to 1034:

```
2.000000+4 0.000000+0 2.500000+4 1.183308-4 3.000000+4 9.640164-59547 3 51    4
3.500000+4 8.002040-5 4.000000+4 5.060946-5 4.500000+4 5.880629-69547 3 51    5
5.000000+4 0.000000+0 5.500000+4 0.000000+0 6.000000+4 0.000000+09547 3 51    6
6.500000+4 0.000000+0 7.500000+4 2.335926-4 8.000000+4 2.019060-49547 3 51    7
8.500000+4 1.762582-4 9.000000+4 1.593463-4 9.500000+4 1.542507-49547 3 51    8
```

Became:

```
2.000000+4 0.000000+0 2.500000+4 1.183308-4 3.000000+4 2.198553-59547 3 51    4
3.500000+4 3.214386-5 4.000000+4 4.230738-5 4.500000+4 5.247540-59547 3 51    5
5.000000+4 6.264722-5 5.500000+4 7.282216-5 6.000000+4 8.299951-59547 3 51    6
6.500000+4 9.317860-5 7.500000+4 1.135392-4 8.000000+4 1.237193-49547 3 51    7
8.500000+4 1.338984-4 9.000000+4 1.440758-4 9.500000+4 1.542507-49547 3 51    8
```

Lines 1064 to 1068:

```
2.000000+4 0.000000+0 2.500000+4 2.585243-6 3.000000+4 1.440153-69547 3 52    4
3.500000+4 1.316165-6 4.000000+4 9.509153-7 4.500000+4 1.252221-79547 3 52    5
5.000000+4 0.000000+0 5.500000+4 0.000000+0 6.000000+4 0.000000+09547 3 52    6
6.500000+4 0.000000+0 7.500000+4 7.804414-6 8.000000+4 7.206291-69547 3 52    7
8.500000+4 6.693263-6 9.000000+4 6.436060-6 9.500000+4 6.577657-69547 3 52    8
```

Became:

```
2.000000+4 0.000000+0 2.500000+4 1.216820-7 3.000000+4 2.000000-79547 3 52    4
3.500000+4 3.229684-7 4.000000+4 5.063105-7 4.500000+4 7.364973-79547 3 52    5
5.000000+4 1.000000-6 5.500000+4 1.306234-6 6.000000+4 1.676140-69547 3 52    6
```

```
6.500000+4 2.113481-6 7.500000+4 3.205509-6 8.000000+4 3.867720-69547 3 52    7

8.500000+4 4.612410-6 9.000000+4 5.443341-6 9.500000+4 6.364275-69547 3 52    8


Lines 1097 to 1101:
2.000000+4 0.000000+0 2.500000+4 1.782574-3 3.000000+4 1.770214-39547 3 53    4

3.500000+4 1.536417-3 4.000000+4 9.956295-4 4.500000+4 1.168439-49547 3 53    5

5.000000+4 0.000000+0 5.500000+4 0.000000+0 6.000000+4 0.000000+09547 3 53    6

6.500000+4 0.000000+0 7.500000+4 3.960658-2 8.000000+4 3.261879-29547 3 53    7

8.500000+4 2.713115-2 9.000000+4 2.337517-2 9.500000+4 2.157269-29547 3 53    8


Became:

2.000000+4 0.000000+0 2.500000+4 2.520574-3 3.000000+4 3.770214-39547 3 53    4

3.500000+4 5.063993-3 4.000000+4 6.364531-3 4.500000+4 7.670220-39547 3 53    5

5.000000+4 8.979451-3 5.500000+4 1.029062-2 6.000000+4 1.160211-29547 3 53    6

6.500000+4 1.291231-2 7.500000+4 1.552244-2 8.000000+4 1.681914-29547 3 53    7

8.500000+4 1.810813-2 9.000000+4 1.938779-2 9.500000+4 2.065651-29547 3 53    8


Lines 1131 to 1135
:
2.731335+4 0.000000+0 3.000000+4 9.992005-5 3.500000+4 1.056291-49547 3 54    3

4.000000+4 7.760021-5 4.500000+4 1.015181-5 5.000000+4 0.000000+09547 3 54    4

5.500000+4 0.000000+0 6.000000+4 0.000000+0 6.500000+4 0.000000+09547 3 54    5

7.500000+4 4.876602-3 8.000000+4 4.142531-3 8.500000+4 3.540286-39547 3 54    6

9.000000+4 3.124191-3 9.500000+4 2.945715-3 1.000000+5 3.048565-39547 3 54    7


Became:

2.731335+4 0.000000+0 3.000000+4 6.992005-5 3.500000+4 2.056291-49547 3 54    3

4.000000+4 4.082459-4 4.500000+4 6.131619-4 5.000000+4 8.204804-49547 3 54    4

5.500000+4 1.030305-3 6.000000+4 1.242739-3 6.500000+4 1.457886-39547 3 54    5

7.500000+4 1.896732-3 8.000000+4 2.120638-3 8.500000+4 2.347671-39547 3 54    6

9.000000+4 2.577934-3 9.500000+4 2.811531-3 1.000000+5 3.048565-39547 3 54    7
```

Appendix E. Sample MCNP Input File for Hoho Test (U-235)

```
test of ENDF/B-VII data
1 1 -20.0 -1
2  0       1

1  so 1

mode n p
nps 3000000
sdef pos=0 0 0 erg=d1
phys:p 100.0 1 0
si1   h 0.000139-6 0.152-6 0.414-6 1.13-6
```

```
        3.06-6 8.32-6 22.6-6 61.4-6 167.-6 454.-6 0.001235
        0.00335 0.00912 0.0248 0.0676 0.184 0.303 0.5
        0.823 1.353 1.738 2.232 2.865 3.68
        6.07 7.79 10.0 12.0 13.5 15.0 17.0 20.0 150.0
sp1   d 0 1 30r 1
tmp 2.530e-08 2.530e-08
ctme 15.0
m1  92235.42c   1.0
imp:n 1 0
imp:p  1 0
```

Appendix F. Changes Found While Plotting ENDF70 Cross Sections with XSPLOT2

All room temperature neutron cross sections were plotted using the executable program XSPLOT2 on the machine XFILES. First, the actinides were plotted from the large files ENDF70J and ENDF70K. However, since XSPLOT2 reads the files sequentially, it took a long time for it to read the cross sections at the end of the files. The remaining isotopes (activation and fission products plus a few actinides) were broken into the smaller files ENDF70AA through ENDF70BE for easier viewing. Isotopes ^{127}I, ^{206}Pb, ^{231}Pa, ^{233}Pa, ^{235}U, and ^{238}U had file sizes that were too big for XSPLOT2 to process. All reactions for these isotopes were instead viewed through the T-16 web site (using T-16 ACER files) http://t2.lanl.gov/data/neutron7.html. All reactions for these isotopes appeared to be okay.

The problems that were found included:
- There are "breaks" in the data that appear to be zero cross sections in a small energy range for reaction MT 801 in ^{16}O and MT102 (n,gamma) for ^{58}Co. These are attributed to problems in the evaluation and nothing was done to fix them at this time.

- There were many instances of breaks/drops in the data of heating plots (MT 301) for the following isotopes: ^{22}Na, ^{59}Ni, ^{96}Zr, ^{93}Nb, **^{92}Mo, ^{94}Mo, ^{96}Mo, ^{97}Mo, ^{98}Mo**, ^{115m}Cd, ^{125}Sn, ^{133}Cs, **^{143}Ce, ^{145}Nd, ^{147}Nd, ^{147}Sm, ^{149}Sm, ^{151}Sm, ^{153}Gd, ^{155}Gd,** ^{165}Ho, ^{166m}Ho, ^{166}Er, ^{181}Ta, ^{183}W, ^{184}W, ^{197}Au, ^{196}Hg, ^{199}Hg, ^{202}Hg, and ^{209}Bi. Most of these look like small energy ranges were the cross section was small or negative. Some of these correspond to cases in which the checking program CHECK_HEAT found negative heating cross sections; the isotopes for which this is true are bolded above.

- Some isotopes have reactions for which the plot drops off at the last energy level (i.e. as a vertical line). For many cases, this occurs when there is data past 20 MeV for most but not all reactions. The reactions for which there is no data past 20 MeV often but not always show a sharp drop to zero. These include: ^{53}Cr, ^{57}Fe, ^{60}Ni, ^{196}Hg, ^{200}Hg, ^{204}Pb, and ^{209}Bi. There is also one isotope (^{176}Hf) where the data for one reaction stops at 2 MeV and data for other reactions continues to 20 MeV. The same "drop" is seen in this case as the others.

- In only one case did the observations from the plotting result in the formation of a new evaluation: some of the inelastic scatter reactions (MTs 4, 51, 52, 53, and 54) for ^{242m}Am. From 50 to 65 keV, these inelastic cross sections were set equal to zero whereas there was data above and below. Patrick Talou of T-16 helped smooth out the cross sections below 100 keV so that zero cross sections no longer exist in the middle.