XV. MATXSR

The MATXS material cross section format is a generalized CCCC-type interface format for neutron, photon, and charged-particle data, including cross sections, group-to-group matrices, temperature variations, self-shielding, and time-dependence. The CCCC standards are discussed in more detail in the CCCCR chapter of this manual and in the CCCC-III and CCCC-IV reports.[1, 2] MATXS libraries can be used with the TRANSX code[3, 4] to produce effective cross sections for a wide variety of application codes. This chapter describes version 91.0 of the MATXSR module. Caution: the MATXS format described here is significantly different from the previous version. Existing codes that use the MATXS format will not be compatible with libraries produced using NJOY 91.0.

A. Background

Even the very best nuclear cross section processing code would be useless if it were unable to deliver its products to users. This is the role of the interface file. There have been interface files since the beginning of calculational neutronics; examples include the DTF format (see the DTFR section of this manual) that was devised for the early discrete-ordinates transport code DTF-IV,[5] and the CCCC ISOTXS format[2] (see the CCCCR section of this report). Both of these interface formats are still in use today, but both of them have problems and show their age. Some of these problems result from the increase in the capabilities of computer systems (capabilities that allow us to consider much more complex problems), some arise from the many new kinds of nuclear systems that are being studied today, and some come from 20/20 hindsight, which makes it easy to see the design flaws in earlier formats.

Based on the problems seen with existing interface files, an ideal interface file should be

extendable, so that new cross section types, new incident or secondary particle types, or new energy ranges are easy to add without changing the basic format;

comprehensive, in order to be able to handle as many of the kinds of data produced by the processing code as possible (results should not be lost just because there are no places for them);

generalizable, to allow common methods to be used for similar kinds of data (for example, nn matrices and $\gamma\gamma$ matrices) in order to transfer the experience gained in one field to another, and in order to simplify coding by allowing components to be reused;

self-contained, because it should not be necessary to provide additional information that is not in the file in order to use or interpret the file;

compact, because nuclear data often have many zeros or very small numbers in tables (for example, threshold reactions, scattering matrices), and these zeros must be removed effectively for economic storage and fast transfer of libraries; and

efficient, thus implying that binary mode should be used, that the records have a well-defined maximum size, and that there is a minimum number of records to reduce the number of I/O operations.

Comparing the DTF format to these principles gives the following results: it is fairly *extendable* because it has no fixed particles, energy limits, or reaction types; it is not very *comprehensive* because it can only transmit the total scattering matrix; it is fairly *generalizable* because of the lack of fixed types; it is not at all *self-contained* in that it requires outside definitions like table length, position of the total, group structure, and identity of edit cross sections; it is not very *compact* because most zeros must be given explicitly in the tables; and it is not very *efficient* because it uses coded card-image records.

Similarly, studying the ISOTXS format gives the following results: it is not *extendable* because it works for neutrons only and allows only very limited types of reactions to be included; it is not *comprehensive* because it works for neutrons only and allows only very limited types of reactions to be included; it is not *generalizable* because of its specialization to fast-reactor problems (as proof, note that the CCCC files for γ cross sections use completely different formats); it is reasonably *self-contained* because all the parameters are internal, the group structure is given, and all names needed for labeling an interpreted listing are well determined; it is fairly *compact* because many zeros are removed (but too many still remain); it is fairly *efficient* because it uses binary mode, but record sizes are poorly predictable and very nonuniform, thereby needlessly increasing the size of application codes and the number of I/O operations.

B. The MATXS Format

Following these general principles, the MATXS material cross section file was designed to extend and generalize the existing interface formats while still using the CCCC approach for efficiency and familiarity (see the CCCCR section of this report for more details). The first design principle was that all information would be identified using lists of Hollerith names. As an example, if the list of reactions included in the file contains entries such as `NF`, `NG`, and `N2N`, it is trivial to add additional reactions such as `KERMA` or `DPA`. This approach is much more extendable than the fixed set of reaction flags used in ISOTXS. The second design principle was that the file would be designed to hold sets of vectors and rectangular matrices

Table 1: Standard Particle Names

Name	Particle
N	neutron
G	gamma
P	proton
D	deuteron
T	triton
H	^{3}He nucleus
A	alpha
B	beta
R	residual or recoil (heavier than α)

and that the same format would be used regardless of the contents of the vectors and matrices. As a consequence, once a code can handle $n \rightarrow n$, it can also handle $\gamma \rightarrow \gamma$; once a code can handle $n \rightarrow \gamma$, it can also handle $\gamma \rightarrow n$, $n \rightarrow \beta$, or even $d \rightarrow p$. This approach is an example of generalization. Each material is now divided into data types identified by input and output particle. As an example, $n \rightarrow \gamma$ is a data type characterized by input particle equals neutron, and output particle equals photon. The matrices for this data type contain cross sections for producing photons in photon group γ due to reactions of neutrons in group n. The vectors, if any, would contain photon production cross sections versus neutron group. The use of completely general data types helps make the format comprehensive.

The names for materials are written in the forms `U235`, `FE56`, `TINAT`, or `H2A`. Note that "`NAT`" is used explicitly for elements; names like `BE` or `C` should be avoided. Suffixes "`A`", "`B`", or "`C` are used to label different versions of a material in a library. In order to keep names to six characters, isomers should be identified by incrementing the "thousands" digit in the atomic number field; for example, `NB193A` would be the second version of the first isomer of ^{93}Nb. The standard names for MATXS particles are given in Table 1.

The standard names for the data types (`HTYPE`) are mostly based on these particle names; the use of the terms `SCAT`, `DK`, `THERM` are exceptions. Table 2 illustrates the scheme used.

Reactions names are constructed in MATXSR from the ENDF MT number, the LR flag (if present), and the incident particle name. Examples of the standard names are given in Tables 3–11. Note that the first `N` is omitted from the last three reactions in Table 3. It is implicit in the data-type name. This convention saves space in the name for possible breakup products (see Table 4). The first `N` is also implicit in the reactions of Table 4. No multiplicity is used in the breakup

Table 2: Standard Data-Type Names

Name	Data Type
NSCAT	neutron scattering
NG	neutron-induced gamma production
NP	neutron-induced proton production
NR	neutron-to-recoil matrix
GSCAT	gamma scattering
PSCAT	proton scattering
PN	proton-induced neutron production
...	...
NTHERM	thermal scattering data
DKN	delayed-neutron data
DKHG	decay heat and gamma data
DKB	decay beta data

Table 3: Simple Neutron-Emitting Reactions

Name	MT	Description
NELAS	2	neutron elastic scattering
NNONEL	3	neutron nonelastic (MT1–MT2)
NINEL	4	neutron inelastic sum (MT51–91)
N2N	16	(n,2n)
N3N	17	(n,3n)
NNA	22	(n,n$'\alpha$)
NNP	28	(n,n$'$p)
N01	51	(n,n$_1$)
N02	52	(n,n$_2$)
NCN	91	(n,n$'$) to continuum

product strings in order to avoid possible confusions with the discrete-level number; just count the like letters to get the multiplicity. The names used for the most common neutron absorption reactions are given in Table 5, and the names used for the fission reactions are given in Table 6. MATXS libraries typically give the total fission cross section and all the partial cross sections (when available) in the vector blocks, but they do not give the total fission matrix when the partial matrices are available.

Table 7 gives some additional reaction names, some of which are special NJOY names. As discussed in the GROUPR section of this report, the total cross section can be averaged with the ℓth order of the flux to obtain the multigroup total cross section components $\sigma_{t\ell,g}$. These total cross section components and the corresponding Legendre fluxes are given names like the first four shown in Table 7. Related names with first letters `G`, `P`, `D`, etc., will also be found in MATXS libraries.

Table 4: Breakup Reactions (LR flags)

Name	MT	LR	Description
N07A	57	22	(n,n_7)α
N15P	65	28	(n,n_{15})p
N02AA	52	29	(n,n_2)2α
NCNAAA	91	23	(n,n$'$)3α
N06NA	56	24	(n,n_5)nα
N01EE	51	40	(n,n_1)ee

Table 5: Neutron-Absorption Reactions

Name	MT	Description
NABS	101	total absorption
NG	102	radiative capture
NP	103	(n,p)
NA	107	(n,α)

Table 6: Fission Reactions

Name	MT	Description
NFTOT	18	total fission
NF	19	(n,f) first-chance fission
NNF	20	(n,n$'$f) second-chance fission
N2NF	21	(n,n2f) third-chance fission
N3NF	38	(n,n3f) fourth-chance fission
NUDEL	455	delayed-neutron yield (MF3)
CHID	455	delayed-neutron spectrum (MF5)

The average inverse velocities are defined to preserve the time term of the time-dependent Boltzmann equation:

$$\left\langle \frac{1}{v} \right\rangle = \frac{\displaystyle\int_g (1/v)\phi(E)\,dE}{\displaystyle\int_g \phi(E)\,dE}\ . \tag{1}$$

The meaning of the terms energy-balance heat production and kinematic KERMA factor are discussed in more detail in the HEATR chapter of this manual. Briefly, the energy-balance heating (MT301) is computed by subtracting the energy carried away by neutrons and photons from the energy available for a reaction. The result should be the energy deposited by charged particles and the recoil nucleus, that is, the local heating. Unfortunately, problems with the energy-balance consistency of evaluations, the difficulty of determining the available energy for elements, and the inaccuracy in the difference between relatively large numbers sometimes cause this value to have unphysical values (for example, negative heating). These values do have the property of always conserving energy for large systems. The kinematic value (MT443) is computed from reaction kinematics alone. It is very accurate at low energies, but when three or more particles are involved in the reaction, it begins to fail. The results in `KERMA` are always larger than the correct heating value. Comparing the two estimates for the local heating can reveal problems in the evaluations.[6, 7] The MATXS user is free to choose whichever number is more appropriate for the problem. The reaction `DAME` is also generated using data from HEATR. As discussed in the HEATR section of this report, this "damage-energy production" cross section can be used to obtain the DPA (displacements per atom) parameter used in radiation damage studies.

The gas-production reaction names given in Table 8 can also appear with other particle names before the decimal point. The names for reactions induced by incident charged particles follow the neutron names in most cases, except that the first letter is changed to indicate the incident particle type. Discrete-level scattering reactions are exceptions; `N01` is used for both (n,n_1) and (p,n_1). Also note the `N00` cannot be used for incident neutrons; the name `NELAS` is used instead. Similarly, `P00` is not used for incident protons.

As discussed in more detail below, scattering from thermal moderators is treated like *materials* in ENDF/B libraries, but it is treated like *reactions* on the GENDF files. The free-gas scattering reaction can appear in any material, but the other thermal MT numbers can only appear in the material corresponding to the dominant scattering isotope. For example, `H20` only appears in H1. There are

Table 7: Special NJOY Names

Name	MT	Description
NTOT0	1	P_0 total cross section
NTOT1	1	P_1 total cross section
NWT0	1	P_0 weight function (flux)
NWT1	1	P_1 weight function (flux)
MUBAR	251	scattering $\bar{\mu}$
XI	252	scattering ξ
INVEL	259	inverse velocity (sec/m)
HEAT	301	energy-balance heat production
KERMA	443	kinematic KERMA factor
DAME	444	damage-energy production

two versions of ZRHYD; one appears in H1 and the other in Zr. The coherent and incoherent terms in the thermal cross section are kept separate for the convenience of applications; all the coherent names end with $. Note that MATXS files contain two different representations for the scattering cross sections at low energies:

static, where the cross section and group-to-group matrix are obtained from NSCAT, which is derived from MT2 on the ENDF evaluation (This is scattering for "static" nuclei; energy loss from recoil is included.); and

thermal, where the cross section and group-to-group matrix are obtained from one of the thermal reactions in the NTHERM data type. (The scattering nuclei are in motion with a distribution described by the Maxwell-Boltzmann law; both energy loss and energy gain events are possible.)

The TRANSX code gives the user the choice of static or thermal scattering, and it also allows the user to choose which binding state is desired for a particular moderator material.

The ENDF representation of photoatomic reactions is described in the GAMINR section of this report. The GHEAT reaction is constructed in GAMINR to represent the local heating due to atomic recoil and the photo-electric production of electrons. Flourescence photons from photoelectric interactions are assumed to deposit their energy locally.

The CCCC standards have always used 6-character Hollerith strings for names. These kinds of names are represented as "REAL*8" double precision variables on 32-bit machines (IBM, VAX, Sun, etc.) and as single-precision variables on 60- to 64-bit machines (CDC, Cray). However, a double-precision variable on a short-word machine can hold 8 characters. So can single-precision variables on CDC and Cray machines. There do not seem to be any computer systems currently in use that require 6-character words. Therefore, the latest versions of the MATXS

Table 8: Gas-Production Reactions

Name	MT	Description
N.NEUT	201	total neutron production
N.GAM	202	total γ production
N.H1	203	hydrogen production
N.H3	205	tritium production
N.HE4	207	helium production

Table 9: Incident-Proton Reactions

Name	MT	Description
PELAS	2	proton elastic scattering
P01	601	discrete-level (p,p_1) scattering
N00	50	discrete-level (p,n_0)
N01	51	discrete-level (p,n_1)
P2N	16	(p,2n)
PG	102	(p,γ)
PT	104	(p,t)

Table 10: Thermal Cross Sections

Name	MT	Description
FREE	221	free-gas scattering
H2O	222	H in H_2O
POLY	223	H in polyethylene (CH_2) incoherent
POLY$	224	H in polyethylene (CH_2) coherent
ZRHYD	225	H in ZrH incoherent
ZRHYD$	226	H in ZrH coherent
BENZ	227	Benzene incoherent
D2O	228	D in D_2O
GRAPH	229	C in graphite incoherent
GRAPH$	230	C in graphite coherent
BE	231	Be metal incoherent
BE$	232	Be metal coherent
BEO	233	BeO incoherent
BEO$	234	BeO coherent
ZRHYD	235	Zr in ZrH incoherent
ZRHYD$	236	Zr in ZrH coherent

Table 11: Photoatomic Cross Sections

Name	MT	Description
GTOT0	501	P_0 total
GWT0	501	P_0 weight function (flux)
GCOH	502	coherent scattering
GINCH	504	incoherent scattering
GPAIR	516	pair production $(\gamma, 2\gamma)$
GABS	522	photoelectric absorption
GHEAT	525	heating

format and the MATXSR module have been written to handle 8-character names.

The formal format specification for the MATXS material cross section file follows, using the standard CCCC presentation:

```
C***********************************************************************
C               PROPOSED 09/09/77                                      -
C                       (MODIFIED 09/80)                               -
C                       (NOMENCLATURE CHANGED 06/88)                   -
C                       (MODIFIED FOR CONST SUB-BLOCKS 06/90)          -
C                       (ORDERING CHANGED 10/90)                       -
C                                                                      -
CF            MATXS                                                    -
CE            MATERIAL CROSS SECTION FILE                              -
C                                                                      -
CN                       THIS FILE CONTAINS CROSS SECTION              -
CN                       VECTORS AND MATRICES FOR ALL                  -
CN                       PARTICLES, MATERIALS, AND REACTIONS;          -
CN                       DELAYED NEUTRON SPECTRA BY TIME GROUP;        -
CN                       AND DECAY HEAT AND PHOTON SPECTRA.            -
C                                                                      -
CN            FORMATS GIVEN ARE FOR FILE EXCHANGE ONLY                 -
C                                                                      -
C***********************************************************************
C
C
C-----------------------------------------------------------------------
CS           FILE STRUCTURE                                            -
CS                                                                     -
CS              RECORD TYPE                         PRESENT IF         -
CS              ==============================      ===============    -
CS              FILE IDENTIFICATION                   ALWAYS           -
CS              FILE CONTROL                          ALWAYS           -
CS              SET HOLLERITH IDENTIFICATION          ALWAYS           -
CS              FILE DATA                             ALWAYS           -
CS                                                                     -
CS   *************(REPEAT FOR ALL PARTICLES)                           -
CS   *          GROUP STRUCTURES                      ALWAYS           -
CS   *************                                                     -
CS                                                                     -
```

```
CS   *************(REPEAT FOR ALL MATERIALS)                                -
CS   *          MATERIAL CONTROL                    ALWAYS                  -
CS   *                                                                      -
CS   * ***********(REPEAT FOR ALL SUBMATERIALS)                             -
CS   * *        VECTOR CONTROL                      N1DB.GT.0               -
CS   * *                                                                    -
CS   * * *********(REPEAT FOR ALL VECTOR BLOCKS)                            -
CS   * * *      VECTOR BLOCK                        N1DB.GT.0               -
CS   * * *********                                                          -
CS   * *                                                                    -
CS   * * *********(REPEAT FOR ALL MATRIX BLOCKS)                            -
CS   * * *      MATRIX CONTROL                      N2D.GT.0                -
CS   * * *                                                                  -
CS   * * * *******(REPEAT FOR ALL SUB-BLOCKS)                               -
CS   * * * *    MATRIX SUB-BLOCK                    N2D.GT.0                -
CS   * * * *******                                                          -
CS   * * *                                                                  -
CS   * * *      CONSTANT SUB-BLOCK                  JCONST.GT.0             -
CS   * * *                                                                  -
CS   *************                                                          -
C                                                                           -
C---------------------------------------------------------------------------
C
C
C---------------------------------------------------------------------------
CR          FILE IDENTIFICATION                                             -
C                                                                           -
CL    HNAME,(HUSE(I),I=1,2),IVERS                                           -
C                                                                           -
CW    1+3*MULT                                                              -
C                                                                           -
CB    FORMAT(4H OV ,A8,1H*,2A8,1H*,I6)                                      -
C                                                                           -
CD    HNAME         HOLLERITH FILE NAME  - MATXS -  (A8)                    -
CD    HUSE          HOLLERITH USER IDENTIFICATION   (A8)                    -
CD    IVERS         FILE VERSION NUMBER                                     -
CD    MULT          DOUBLE PRECISION PARAMETER                              -
CD                       1- A8 WORD IS SINGLE WORD                          -
CD                       2- A8 WORD IS DOUBLE PRECISION WORD                -
C                                                                           -
C---------------------------------------------------------------------------
C
C
C---------------------------------------------------------------------------
CR          FILE CONTROL                                                    -
C                                                                           -
CL    NPART,NTYPE,NHOLL,NMAT,MAXW,LENGTH                                    -
C                                                                           -
CW    6                                                                     -
C                                                                           -
CB    FORMAT(4H 1D ,4I6)                                                    -
C                                                                           -
CD    NPART       NUMBER OF PARTICLES FOR WHICH GROUP                       -
```

```
CD                   STRUCTURES ARE GIVEN                              -
CD    NTYPE       NUMBER OF DATA TYPES PRESENT IN SET                  -
CD    NHOLL       NUMBER OF WORDS IN SET HOLLERITH                     -
CD                    IDENTIFICATION RECORD                            -
CD    NMAT        NUMBER OF MATERIALS ON FILE                          -
CD    MAXW        MAXIMUM RECORD SIZE FOR SUB-BLOCKING                 -
CD    LENGTH      LENGTH OF FILE                                       -
C                                                                      -
C-----------------------------------------------------------------------
C
C
C-----------------------------------------------------------------------
CR            SET HOLLERITH IDENTIFICATION                             -
C                                                                      -
CL    (HSETID(I),I=1,NHOLL)                                            -
C                                                                      -
CW    NHOLL*MULT                                                       -
C                                                                      -
CB    FORMAT(4H 2D ,8A8/(9A8))                                         -
C                                                                      -
CD    HSETID      HOLLERITH IDENTIFICATION OF SET (A8)                 -
CD                 (TO BE EDITED OUT 72 CHARACTERS PER LINE)           -
C                                                                      -
C-----------------------------------------------------------------------
C
C
C-----------------------------------------------------------------------
CR           FILE DATA                                                 -
C                                                                      -
CL    (HPRT(J),J=1,NPART),(HTYPE(K),K=1,NTYPE),(HMATN(I),I=1,NMAT),    -
CL   1(NGRP(J),J=1,NPART),(JINP(K),K=1,NTYPE,(JOUTP(K),K=1,NTYPE),     -
CL   2(NSUBM(I)I=1,NMAT),(LOCM(I),I=1,NMAT)                            -
C                                                                      -
CW    (NPART+NTYPE+NMAT)*MULT+2*NTYPE+NPART+2*NMAT                     -
C                                                                      -
CB    FORMAT(4H 3D ,8A8/(9A8))      HPRT,HTYPE,HMATN                   -
CB    FORMAT(12I6)                  NGRP,JINP,JOUTP,NSUBM,LOCM         -
C                                                                      -
CD    HPRT(J)     HOLLERITH IDENTIFICATION FOR PARTICLE J              -
CD                    N         NEUTRON                                -
CD                    G         GAMMA                                  -
CD                    P         PROTON                                 -
CD                    D         DEUTERON                               -
CD                    T         TRITON                                 -
CD                    H         HE-3 NUCLEUS                           -
CD                    A         ALPHA (HE-4 NUCLEUS)                   -
CD                    B         BETA                                   -
CD                    R         RESIDUAL OR RECOIL                     -
CD                              (HEAVIER THAN ALPHA)                   -
CD    HTYPE(K)     HOLLERITH IDENTIFICATION FOR DATA TYPE K            -
CD                    NSCAT     NEUTRON SCATTERING                     -
CD                    NG        NEUTRON INDUCED GAMMA PRODUCTION       -
CD                    GSCAT     GAMMA SCATTERING                       -
```

```
CD                      PN        PROTON INDUCED NEUTRON PRODUCTION     -
CD                        .          .                                  -
CD                        .          .                                  -
CD                        .          .                                  -
CD                      DKN       DELAYED NEUTRON DATA                  -
CD                      DKHG      DECAY HEAT AND GAMMA DATA             -
CD                      DKB       DECAY BETA DATA                       -
CD    HMATN(I)    HOLLERITH IDENTIFICATION FOR MATERIAL I               -
CD    NGRP(J)      NUMBER OF ENERGY GROUPS FOR PARTICLE J               -
CD    JINP(K)     TYPE OF INCIDENT PARTICLE ASSOCIATED WITH             -
CD                   DATA TYPE K.  FOR DK DATA TYPES, JINP IS 0.        -
CD    JOUTP(K)    TYPE OF OUTGOING PARTICLE ASSOCIATED WITH             -
CD                   DATA TYPE K                                        -
CD    NSUBM(I)    NUMBER OF SUBMATERIALS FOR MATERIAL I                 -
CD    LOCM(I)     LOCATION OF MATERIAL I                                -
C                                                                       -
C-----------------------------------------------------------------------
C
C
C-----------------------------------------------------------------------
CR          GROUP STRUCTURE                                             -
C                                                                       -
CL    (GPB(I),I=1,NGR),EMIN                                             -
C                                                                       -
CC    NGR=NGRP(J)                                                       -
C                                                                       -
CW    NGRP(J)+1                                                         -
C                                                                       -
CB    FORMAT(4H 4D ,1P5E12.5/(6E12.5))                                  -
C                                                                       -
CD    GPB(I)      MAXIMUM ENERGY BOUND FOR GROUP I FOR PARTICLE J       -
CD    EMIN        MINIMUM ENERGY BOUND FOR PARTICLE J                   -
C                                                                       -
C-----------------------------------------------------------------------
C
C
C-----------------------------------------------------------------------
CR          MATERIAL CONTROL                                            -
C                                                                       -
CL    HMAT,AMASS,(TEMP(I),SIGZ(I),ITYPE(I),N1D(I),N2D(I),               -
CL   1LOCS(I),I=1,NSUBM)                                                -
C                                                                       -
CW    MULT+1+6*NSUBM                                                    -
C                                                                       -
CB    FORMAT(4H 6D ,A8,1H*,1P2E12.5/(2E12.5,5I6))                       -
C                                                                       -
CD    HMAT        HOLLERITH MATERIAL IDENTIFIER                         -
CD    AMASS       ATOMIC WEIGHT RATIO                                   -
CD    TEMP        AMBIENT TEMPERATURE OR OTHER PARAMETERS FOR           -
CD                    SUBMATERIAL I                                     -
CD    SIGZ        DILUTION FACTOR OR OTHER PARAMETERS FOR               -
CD                    SUBMATERIAL I                                     -
CD    ITYPE       DATA TYPE FOR SUBMATERIAL I                           -
```

```
CD    N1D         NUMBER OF VECTORS FOR SUBMATERIAL I                  -
CD    N2D         NUMBER OF MATRIX BLOCKS FOR SUBMATERIAL I            -
CD    LOCS        LOCATION OF SUBMATERIAL I                            -
C                                                                      -
C-----------------------------------------------------------------------
C
C
C-----------------------------------------------------------------------
CR          VECTOR CONTROL                                             -
C                                                                      -
CL    (HVPS(I),I=1,N1D),(NFG(I),I=1,N1D),(NLG(I),I=1,N1D)              -
C                                                                      -
CW    (MULT+2)*N1D                                                     -
C                                                                      -
CB    FORMAT(4H 7D ,8A8/(9A8))        HVPS                             -
CB    FORMAT(12I6)                    IBLK,NFG,NLG                     -
C                                                                      -
CD    HVPS(I)     HOLLERITH IDENTIFIER OF VECTOR                       -
CD                      NELAS     NEUTRON ELASTIC SCATTERING           -
CD                      N2N       (N,2N)                               -
CD                      NNF       SECOND CHANCE FISSION                -
CD                      GABS      GAMMA ABSORPTION                     -
CD                      P2N       PROTONS IN, 2 NEUTRONS OUT           -
CD                         .          .                                -
CD                         .          .                                -
CD                         .          .                                -
CD    NFG(I)      NUMBER OF FIRST GROUP IN BAND FOR VECTOR I           -
CD    NLG(I)      NUMBER OF LAST GROUP IN BAND FOR VECTOR I            -
C                                                                      -
C-----------------------------------------------------------------------
C
C
C-----------------------------------------------------------------------
CR          VECTOR BLOCK                                               -
C                                                                      -
CL    (VPS(I),I=1,KMAX)                                                -
C                                                                      -
CC    KMAX=SUM OVER GROUP BAND FOR EACH VECTOR IN BLOCK J              -
C                                                                      -
CW    KMAX                                                             -
C                                                                      -
CB    FORMAT(4H 8D ,1P5E12.5/(6E12.5))                                 -
C                                                                      -
CD    VPS(I)      DATA FOR GROUP BANDS FOR VECTORS IN BLOCK J.         -
CD                BLOCK SIZE IS DETERMINED BY TAKING ALL THE GROUP     -
CD                BANDS THAT HAVE A TOTAL LENGTH LESS THAN OR EQUAL    -
CD                TO MAXW.                                             -
C                                                                      -
C-----------------------------------------------------------------------
C
C
C-----------------------------------------------------------------------
CR        SCATTERING MATRIX CONTROL                                    -
```

```
C                                                                       -
CL    HMTX,LONE,LORD,JCONST,                                            -
CL   1(JBAND(L),L=1,NOUTG(K)),(IJJ(L),L=1,NOUTG(K))                     -
C                                                                       -
CW    MULT+3+2*NOUTG(K)                                                 -
C                                                                       -
CB    FORMAT(4H 9D ,A8/(12I6))        HMTX,LONE,LORD,JCONST,            -
CB                                     JBAND,IJJ                        -
C                                                                       -
CD    HMTX        HOLLERITH IDENTIFICATION OF BLOCK                     -
CD    LONE        LOWEST ORDER PRESENT                                  -
CD    LORD        NUMBER OF ORDERS PRESENT                              -
CD    JCONST      NUMBER OF GROUPS WITH CONSTANT SPECTRUM               -
CD    JBAND(L)    BANDWIDTH FOR GROUP L                                 -
CD    IJJ(L)      LOWEST GROUP IN BAND FOR GROUP L                      -
C                                                                       -
C-----------------------------------------------------------------------
C
C
C-----------------------------------------------------------------------
CR          SCATTERING SUB-BLOCK                                        -
C                                                                       -
CL    (SCAT(K),K=1,KMAX)                                                -
C                                                                       -
CC    KMAX=LORD TIMES THE SUM OVER ALL JBAND IN THE GROUP RANGE OF      -
CC            THIS SUB-BLOCK                                            -
C                                                                       -
CB    FORMAT(5H 10D ,1P5E12.5/(6E12.5))                                 -
C                                                                       -
CW    KMAX                                                              -
C                                                                       -
CD    SCAT(K)     MATRIX DATA GIVEN AS BANDS OF ELEMENTS FOR INITIAL    -
CD                GROUPS THAT LEAD TO EACH FINAL GROUP.  THE ORDER      -
CD                OF THE ELEMENTS IS AS FOLLOWS:  BAND FOR P0 OF        -
CD                GROUP I, BAND FOR P1 OF GROUP I, ... , BAND FOR P0    -
CD                OF GROUP I+1, BAND FOR P1 OF GROUP I+1, ETC.  THE     -
CD                GROUPS IN EACH BAND ARE GIVEN IN DESCENDING ORDER.    -
CD                THE SIZE OF EACH SUB-BLOCK IS DETERMINED BY THE       -
CD                TOTAL LENGTH OF A GROUP OF BANDS THAT IS LESS THAN    -
CD                OR EQUAL TO MAXW.                                     -
CD                                                                      -
CD                IF JCONST.GT.0, THE CONTRIBUTIONS FROM THE JCONST     -
CD                LOW-ENERGY GROUPS ARE GIVEN SEPARATELY.               -
C                                                                       -
C-----------------------------------------------------------------------
C                                                                       -
C                                                                       -
C-----------------------------------------------------------------------
CR          CONSTANT SUB-BLOCK                                          -
C                                                                       -
CL    (SPEC(L),L=1,NOUTG(K)),(PROD(L),L=L1,NING(K))                     -
C                                                                       -
CC    L1=NING(K)-JCONST+1                                               -
```

```
C                                                                         -
CW    NOUTG(K)+JCONST                                                     -
C                                                                         -
CB    FORMAT(4H11D ,1P5E12.5/(6E12.5))                                    -
C                                                                         -
CD    SPEC        NORMALIZED SPECTRUM OF FINAL PARTICLES FOR INITIAL      -
CD                PARTICLES IN GROUPS L1 TO NING(K)                       -
CD    PROD        PRODUCTION CROSS SECTION (E.G., NU*SIGF) FOR            -
CD                INITIAL GROUPS L1 THROUGH NING(K)                       -
CD                                                                        -
CD         THIS OPTION IS NORMALLY USED FOR THE ENERGY-INDEPENDENT        -
CD         NEUTRON AND PHOTON SPECTRA FROM FISSION AND RADIATIVE          -
CD         CAPTURE USUALLY SEEN AT LOW ENERGIES.                          -
C                                                                         -
C-------------------------------------------------------------------------
```

The MATXS format is intended to communicate multigroup cross sections and matrices for all reaction types, incident particles, and outgoing particles from a nuclear data processing code to applications. It also includes temperature and self-shielding effects, delayed-neutron data, and a limited format for decay heat and delayed photon or particle emission. As shown in the "File Structure" presentation above, the main loop is over material. Materials are subdivided into submaterials, which usually correspond to different data types, temperatures, and background cross section (σ_0) values. Each submaterial can contain a series of "Vector Blocks" giving cross section versus energy for one of the allowed group structures and incident particles, and it can contain a series of matrix blocks and subblocks giving the cross sections for group-to-group transfers.

The "File Identification" record is the same for all CCCC files. It gives the Hollerith name for the file (which is always `MATXS`), a version number `IVERS`, which can be used to distinguish between different libraries in this format, and a Hollerith identification string `HUSE`, which can be used for entries like "`T2 LANL NJOY`".

The "File Control" record contains parameters that are needed to compute the lengths of the following records. The meaning of the various names is well explained in the format specification, and the values of the parameters are obtained from the user's input. The purpose of `MAXW` is to tell application codes how much memory they will need to read through the records on these MATXS files. It is used for both vector blocks and matrix subblocks when deciding how to break them up. The MATXSR value is 5000 words. The code tries to make as many records as possible that have nearly this size in order to minimize the number of I/O operations. The parameter `LENGTH` is used to help find the end of the file when appending a new material to an existing file. Its units are left unspecified in the format. It is usually the length in records, in which case record skipping can be used to find the end. Or it could be the length in words on computer systems that allow direct word-addressed I/O operations (for example, CTSS on Cray computers).

The "Set Hollerith Identification" record comes next. It contains an arbitrary amount of Hollerith text to describe the contents of the library. The description comes from the user's input.

The "File Data" record contains a number of important arrays that define the structure of data types and the location of materials. The parameter `HPRT` contains the standard names for the `NPART` particles. The standard names for particles were discussed above. The standard names for the data types and materials (`HTYPE` and `HMATN`) were also discussed above. The next three parameters are used to complete the specification of the data types included in this MATXS library. `NGRP` just gives the number of groups used for each particle type; for example, the traditional Los Alamos 30×12 library (with 30 neutron groups and 12 photon groups) would have `NGRP(1)=30` and `NGRP(2)=12`. Using the same example, the `NSCAT` data type would have `JINP(1)=1` and `JOUTP(1)=1`; the `NG` data type would have `JINP(2)=1` and `JOUTP(2)=2`; and the `GSCAT` data type would have `JINP(3)=2` and `JOUTP(3)=2`. The information for these 6 arrays is given in the user's input.

The final two parameters in the "File Data" record are `NSUBM` and `LOCM`. The value of `NSUBM` depends on the number of data types, the number of temperatures, and the number of background cross sections found on the input GENDF tapes. The value for `LOCM(I)` is usually the record index for material `I`. A code can then jump directly to a desired material using record skipping (forward or backward). However, the units for `LOCM` have been left unspecified to allow direct random access for systems that use word-addressable random-access I/O operations (such as CTSS on Cray computers).

The "Group Structure" records give the energy bounds for `NPART` group structures. Following the normal convention for application codes, the energy bounds are given in the order of decreasing energy. The numbers are obtained from MF1,MT451 on the GENDF tape. The GROUPR module currently uses one group structure for all particles (n, p, α, etc.), and another for photons (γ).

Inside the material loop, there is a "Material Control" record for each material. The choice of names for `HMAT` was discussed above. The `AMASS` parameter is the same as the ENDF AWR parameter; that is, it is the ratio of the target mass to the neutron mass. Temperatures `TEMP` are given in degrees Kelvin, and background cross sections for self-shielding codes (`SIGZ`, or σ_0) are given in barns. The parameter `ITYPE` tells which data-type each submaterial belongs to using the data type codes defined by `HTYPE`. Although it is not specified in the format description, the order that MATXSR loops through submaterials is as follows: the outer loop is over data type, the next loop is over temperature, and the innermost loop is over background cross section. The number of cross section vector reactions and matrix reactions for each submaterial are given in `N1D` and `N2D`. Finally, `LOCS(I)` normally gives the record index for submaterial `I`. A code can search the arrays `TEMP`, `SIGZ`, and `ITYPE` for a desired submaterial, and then jump right to the desired submaterial using record skipping. Alternatively, a version that uses word addresses in `LOCS` could use random-access methods to jump to the desired submaterial.

Each submaterial that contains vectors (`N1D>0`) starts with a "Vector Control" record. This record gives a list of reaction names in `HVPS`. MATXSR constructs these reaction names automatically based on the MT number, ENDF "LR flag" (if any), and the incident particle type. Examples of these names were given above. The parameters `NFG` and `NLG` are used to remove unnecessary leading or trailing zeros in reaction cross section vectors. The zeros are usually due to thresholds. For example, an (n,2n) reaction might only have nonzero cross sections for groups 1 through 5 out of an 80-group structure. Storing only the 5-element band will save 75 words on the MATXS file. Even with the zeros removed, the number of words of vector cross section data for a submaterial can be quite large. Therefore, the MATXS format provides a way to break the vector data into a number of vector blocks. The idea is to sum up the bandwidths for each reaction (that is, `NLG(I)-NFG(I)+1`) in order to find the largest number of reactions that will fit within a block of length less than or equal to `MAXW`. The data for this block are written to the output file, and then the next group of reactions are found. This continues until all the vector data have been written out as a series of "Vector Block" records with lengths less than `MAXW` (usually 5000 words in MATXSR). This method minimizes the number of records on the file while allowing the application codes that read MATXS libraries to allocate space for reading the records economically.

Most submaterials will contain `N2B` "Scattering Matrix Control" records. The convention for the reaction names used in `HMTX` were discussed above. `LORD` gives the *number* of Legendre orders present for this reaction; that is, `LORD=4` for a P_3 matrix. Matrices are compacted for efficient storage and data transfer using two techniques. First, unnecessary leading and trailing zeros for group-to-group transfers into a particular final group are removed by banding. `JBAND(I)` gives the number of incident groups in the band for final group `I`, and `IJJ(I)` gives the group index for the lowest-energy group (highest group index) in the band for group `I`. For example, consider an isotropic (n,2n) reaction with a threshold in group 3 of a 30-group structure. The values for `JBAND` and `IJJ` for group 20 might well be 3 and 3, respectively. That is, groups 1 through 3 scatter into group 20. The order of storage for this matrix would be as follows:

Band	Element	JBAND	IJJ
1	1→1	1	1
2	2→2	2	2
	1→2		
3	3→3	3	3
	2→3		
	1→3		
4	3→4	3	3
	2→4		
	1→4		
...	...		

Note that this scheme is more efficient than the similar one used for the CCCC ISOTXS file, because that method required that the ingroup element had to be included in the band. For anisotropic matrices, the `LORD` Legendre components are stored with the source-group elements. Thus, there are `LORD(I)*JBAND(I)` elements in each band, and they are stored in the following order (assuming a P_1 matrix):

Band	Element	Order	JBAND	IJJ
1	1→1	0	1	1
	1→1	1		
2	2→2	0	2	2
	1→2	0		
	2→2	1		
	1→2	1		
3	3→3	0	3	3
	2→3	0		
	1→3	0		
	3→3	1		
...	...			

This kind of matrix block can be subdivided into subblocks using a method similar to the one described above for vector cross sections. The code starts summing the product of `JBAND` and `LORD` for the final energy groups until the data for the next band will cause the sum to exceed `MAXW` (normally 5000 words in MATXSR). These data are then written out as a matrix subblock. The code then repeats the process for the rest of the group ranges. The result is a minimal number of "Scattering Sub-Blocks," none of which has a size larger than the file limit.

The second method used to compact scattering matrices is based on the observation that the shape of the outgoing neutron or photon spectrum from fission and radiative capture reactions tends to be independent of energy at low neutron energies. GROUPR determines the group where significant energy dependence begins. Below this point, it computes a single spectrum to describe the outgoing neutron or photon distribution and a production cross section to go with it. At high energies, it produces a group-to-group matrix. This method can lead to appreciable reductions in storage requirements. For example, consider the 187-group structure, which has many low-energy groups. The fission spectrum for ^{235}U doesn't begin to show significant energy dependence until an energy of about 9 keV. This means that there are 118 constant groups. A 187×187 matrix is thereby reduced to a 187×69 matrix, and 187-group vector, and a 118-group vector—a 62 % reduction in storage requirements. Even larger reductions in size are obtained in more favorable cases. The parameter `JCONST` gives the number of low-energy groups having a constant spectrum. If `JCONST>0`, a single "Constant Sub-Block" record will be given after the regular "Scattering Sub-Block" records. This record will contain the spectrum `SPEC` and the production cross section `PROD` needed to reconstruct the low-energy part of the full matrix. In mathematical form,

$$\sigma_{x,g\to g'} = \chi^{LE}_{g'}\,\sigma^{LE}_{Px,g} + \sigma^{HE}_{x,g\to g'}\ , \tag{2}$$

where LE stands for low energy, HE stands for high energy, χ^{LE} is the constant spectrum, σ^{LE}_{Px} is the production cross section for reaction x, and σ^{HE} is the normal group-to-group matrix.

C. Historical Notes

As noted at the top of the MATXS format description, the original version of the file was constructed in September of 1977. The "Matrix Control" record in this original version contained the names and banding parameters for every group of every reaction. As a result, the record could become very large for libraries with many groups. The MATXS format was modified in September of 1980 to have a different "Matrix Control" record for each reaction. For several years following this date, NJOY contained both MATXSR and NMATXS modules, and two different versions of TRANSX were in use. All traces of the original version of the format have now disappeared.

The last few changes are all fairly recent. The changes of June 1988 were introduced to make MATXSR able to handle data types with either incoming or outgoing charged particles. Actually, the format wasn't changed. The particle, data type, and reaction name conventions described in Tables 1–11 were chosen, and these new names required some corresponding changes in the code. The

TRANSX code also had to be modified to recognize the new names and to work with multiparticle coupled sets.

The concept of using constant subblocks to reduce the size of GENDF and MATXS files is actually quite old. Various versions of updates to install the scheme were written over about a 5-year period. It was finally decided to permanently install the scheme in June of 1990. The changes required to GROUPR were the easier part of the job; corresponding changes were required in DTFR, CCCCR, MATXSR, POWR, and WIMSR. In addition, TRANSX had to be changed to accept the new constant subblocks. Unfortunately, this change is large enough to impact all MATXS users.

If many users were to be impacted, other nagging problems with the code could also be corrected at this time. With the original format, it was always very difficult to add, replace, or extract a material. The data type loop was outside the material loop, and the pieces for a given material were scattered throughout the file. Therefore, the ordering of the file was changed to put the data type loop inside the material loop along with the other submaterials. With this change, it was easy to rewrite the BBC code to be able to insert a new material at any point in the library. It was also easy to give BBC the capability of extracting a short library containing only selected materials from a large MATXS library. Using the short library can significantly speed up TRANSX runs to prepare data for reactor design problems. The only disadvantage of the new ordering is that identical photoatomic data blocks have to be given for each isotope of an element. However, this data is not too bulky, and the additional overhead is manageable. This arrangement might make it easier to add photonuclear data in the future.

Finally, the current scheme of dividing the vector and matrix data into subblocks was added. The goal was to keep the record size below some fairly large maximum value (5000 words is currently being used). This kind of limit makes it easier to design application codes that make efficient use of memory. In addition, using as few records as possible reduces the number of I/O operations needed for a trip through the library, thereby improving execution time. The new subblocking scheme is also very simple.

D. MATXS Libraries

The normal process for preparing a MATXS library using NJOY starts with a series of runs to prepare PENDF tapes for each of the materials of interest. (For incident neutrons, these PENDF runs normally involve running the modules RECONR, BROADR, UNRESR, HEATR, and THERMR. For incident photons and charged particles, only RECONR is needed.) The next step is to run GROUPR for each material and incident particle. Each GROUPR run can produce data for all outgoing particles and for photons. As an example, consider the problem of producing data for a coupled neutron-photon-proton library. GROUPR would be run using the ENDF-6 incident-neutron data and requesting MFD values of 3 (cross sections), 6 (neutron matrices, nn), 16 (photon production matrices, $n\gamma$), and 21 (proton production matrices, np). GAMINR would be run to produce the photoatomic cross sections and $\gamma\gamma$ matrices (no photonuclear data are currently available, so γn and γp matrices cannot be generated). GROUPR would be run again for the ENDF-6 proton sublibrary, requesting MFD values of 3 (cross sections), 6 (neutron production, pn), 16 (photon production, $p\gamma$), and 21 (proton scattering and production, pp).

Once all the GENDF fragments from all these GROUPR and GAMINR runs are available, they can be merged into multimaterial GENDF tapes using MODER. For the example above, three multimaterial GENDF tapes should be prepared: one for incident neutrons, one for incident photons, and one for incident protons. The input description in the following section shows how the unit numbers for these three GENDF files are given to MATXSR.

When the MATXSR run is made, the code scans through all the input GENDF tapes searching for the specified materials. For each material, it extracts all the data types requested in the MATXSR input and appearing on the input tapes. In addition, it extracts every temperature, σ_0 value, reaction, and Legendre order found on the input tapes. The following paragraphs describe various special features of the formatting process.

1. Normal Neutron and Photon Data. Normal, infinitely dilute cross sections and group-to-group matrices for neutron reactions, photon production, and photonuclear reactions are read from the input tapes and stored in the MATXS file using the formats described above. Almost all partial reactions are kept. Reaction names are constructed automatically; a number of examples were given above in Tables 3–11. The total cross section and weighting flux from GROUPR are given in the section MF3,MT1 with both P_0 and P_1 components. All four vectors are written to the MATXS file. A similar treatment is used for MT501 in the photoatomic case, except that only P_0 terms are saved.

The treatment of fission is also special. ENDF libraries sometimes use only MT18, the total fission reaction, for both cross section and emission spectrum data; sometimes the partial fission cross sections (n,f), (n,n′f), (n,2nf), and (n,3nf) are also given in File 3 (MT19, 20, 21, and 38); and sometimes the partial reactions are also used to describe the neutron emission matrix. If the last case is found, the MT18 matrix may not be complete above the threshold for second-chance fission, and it should be ignored. Normally, it would not be processed in GROUPR, but just in case, MATXSR will ignore it. Delayed-neutron spectra processed in GROUPR are written into the section labeled MF5,MT455 on the GENDF tape, and this section includes data for all six time groups. MATXSR adds up these separate time-group spectra to produce a single delayed-neutron spectrum for the MATXS vector data blocks. Application codes can use the fission data in the MATXS library to construct steady-state fission vectors as follows:

$$\bar{\nu}_g^{SS} = \frac{\displaystyle\sum_{g'} \sigma_{fg\to g'}^{HE} + \sigma_{Pfg}^{LE} + \bar{\nu}_g^D \sigma_{fg}}{\sigma_{fg}} \;, \tag{3}$$

where σ^{HE} is the matrix part of the prompt fission reaction, σ_{Pfg}^{LE} is the low-energy production cross section for fission neutrons, $\bar{\nu}^D$ is the total delayed-neutron yield, and σ_{fg} is the fission cross section. If partial fission matrices are available, the first two terms in the denominator would also have to be summed over the reactions present. Continuing,

$$\chi_{g'}^{SS} = \frac{\displaystyle\sum_g \sigma_{fg\to g'}^{HE}\phi_g + \chi_{g'}^{LE}\sum_g \sigma_{Pfg}^{LE}\phi_g + \chi_{g'}^D \sum_g \bar{\nu}_g^D \sigma_{fg}\phi_g}{\mathrm{NORM}} \;, \tag{4}$$

where χ^{LE} is the constant-spectrum part of the fission reaction, χ^D is the total delayed-neutron spectrum, and NORM is the quantity that normalizes χ^{SS}, namely, the sum of the numerator over all g'. The weighting flux ϕ_g would normally be problem- and region-dependent in the application code.

2. Self-Shielding Data. If higher temperatures and σ_0 values are found on the input GENDF file, MATXSR automatically prepares submaterials for them. This self-shielding information can be used by application codes to prepare effective cross sections for mixtures of materials in various geometrical arrangements using equivalence theory. This approach is often called the Bondarenko method.[8]

As discussed in more detail in the GROUPR section of this report, this system is based on using a model flux for isotope i of the form

$$\phi_\ell^i(E,T) = \frac{C(E)}{[\sigma_0^i + \sigma_t^i(E,T)]^{\ell+1}} \;, \tag{5}$$

where $C(E)$ is a smooth weighting flux, $\sigma_t^i(E,T)$ is the total cross section for material i at temperature T, ℓ is the Legendre order, and σ_0^i is a parameter that can be used to account for the presence of other materials and the possibility of escape from the absorbing region (heterogeneity). GROUPR uses this model flux to calculate effective multigroup cross sections for the resonance-region reactions (total, elastic, fission, capture) for several values of σ_0 and several values of T.

When σ_0 is large with respect to the highest peaks in σ_t, the flux is essentially proportional to $C(E)$. This is called infinite dilution, and the corresponding cross sections are appropriate for an absorber in a dilute mixture or for a very thin sample of the absorber. As σ_0 decreases, the flux $\phi(E)$ develops dips where σ_t has peaks. These dips cancel out part of the effect of the corresponding peaks in the resonance cross sections, thereby reducing, or self-shielding the reaction rate. In the MATXS format, these effects are represented by differences instead of the f-factors used in earlier formats (see the description of the BRKOXS file in the CCCCR section of this report). That is, the submaterial corresponding to temperature T and background cross section σ_0 contains the differences between the cross sections found on the GENDF tape for those parameters and the infinitely dilute values found in the first submaterial (normally, T=300 K and σ_0=1×10^{10}barns).

This approach has two advantages. First, data for groups with no self-shielding are automatically removed from MATXS vectors and matrices by the banding process without having to violate the principle of *generalization* and define a special format for neglecting values of "1.0" instead of values of "0.0." Second, effective cross sections can be accumulated by simply adding the self-shielding effects multiplied by appropriate interpolation weights to an accumulating sum that starts out equal to the infinitely dilute cross section. With f-factors, it is necessary to save the infinitely dilute cross sections during each step so that they can be multiplied by the f-factors. Thus, using differences is more economical in coding and in storage space requirements. The TRANSX code makes good use of this feature.

3. Thermal Data. ENDF-6 thermal data comes from a thermal sublibrary. In this sublibrary, the various material configurations are handled as separate materials. However, after the ENDF data have been processed by the THERMR module, the different thermal materials are handled as reactions. Table 10 gives the correspondence between the special NJOY thermal MT numbers and the actual thermal materials.

Each of these thermal reactions corresponds to a particular dominant material. For example, MT222 gives the thermal cross section for hydrogen bound in water.

It will only appear in the material H1 in a MATXS library. Similarly, MT223–227 will only appear in H1, MT228 will only appear in H2, MT229–230 will only appear in CNAT, MT231–234 will only appear in BE9, and MT235–236 will only appear in ZRNAT. Free-gas scattering (MT221) will appear in all materials.

Many of these materials describe the scattering from one atom of a compound bound in that compound; for example, H in H_2O, D in D_2O, or Zr in ZrH. The application code using this data is expected to add on the effects of scattering from the other atoms of the compound. For water, the scattering from free oxygen is added to the "H in H_2O" scattering. For zirconium, the scattering for "H in ZrH" is added to the scattering from "Zr in ZrH". There are two exceptions in the existing ENDF/B evaluations. The benzene data set contains the entire scattering from the C_6H_6 molecule normalized to the hydrogen cross section. Therefore, if an application code specifies H1 with the benzene scattering option and the correct density for H1 in the system, the result will contain all of the benzene scattering effect. No additional scattering is to be added for the carbon atoms. Similarly, BeO contains all the scattering from the compound normalized to the beryllium cross section. Check the THERMR section of this report for more information.

Four of these materials have names ending with $. These reactions represent coherent elastic scattering from crystalline powdered materials (C, Be, BeO) or incoherent elastic scattering from solids containing hydrogen (polyethylene). These reactions contain ingroup elements only in the scattering matrices; that is, they cause angular redistribution without energy loss in scattering. Each $ reaction should be added to the corresponding inelastic reaction by the application code. This part of the scattering can lead to difficulties with transport corrections in discrete-ordinates transport codes.

The final unique aspects of the thermal data are that the matrices show up-scatter, and they are only defined below some maximum energy. The energy range aspect is easily handled by the banding method used to reduce the size of the vector blocks. The upscatter aspect is handled by `JBAND` and `IJJ`. The order of storage for a simple thermal case with only 2 upscatter groups would be as follows:

Band	Element	JBAND	IJJ
...	...		
27	27→27	2	27
	26→27		
28	29→28	3	29
	28→28		
	27→28		
29	30→29	3	30
	29→29		
	28→29		
30	30→30	2	30
	29→30		

4. Charged-Particle Data. The treatment of data for incident charged particles is very similar to that used for neutron data, except for charged-particle elastic scattering. The elastic channel has contributions from Coulomb scattering that become infinite as the scattering angle goes to zero. In nature, this singularity is removed by electronic shielding, but some other approach is needed for a data set that concentrates on isolated nuclear reactions. The approach selected is used in some existing applications. The elastic scattering distribution is broken up into two parts: (1) a normal angular distribution for angles from some low cutoff, say 20°, back to 180°, and (2) a straight-ahead continuous slowing-down contribution to represent the effects of angles below the cutoff. The continuous slowing-down part is closely related to the normal "stopping power" for charged particles. The large-angle part can be converted into a normal scattering matrix (see the discussion of charged-particle elastic scattering in the GROUPR chapter of this manual for more details).

Discrete-ordinates transport codes can often be modified to handle charged-particle data in this form, but they require an effective total cross section. The ENDF-6 format for charged-particle data does not define a total cross section because of the singularity in the elastic contribution. However, for application purposes, a reasonable definition of an effective total cross section is that it is the sum of all the partial reaction cross sections including the cross section obtained for the truncated elastic scattering reaction. This is the method that MATXSR uses to compute the quantities PTOTO, DTOTO, TTOTO, etc.

5. Delayed-Neutron Data. Delayed-neutron yields, spectra, and decay constants by time group are required by reactor kinetics codes. The previous CCCC format for these data was DLAYXS (see the CCCCR chapter of this manual). Because of the generalization inherent in the MATXS format, these data can be added without any changes in the structure of the file. This option has not yet been added to MATXSR, and the procedures used will be described in a future version of this report.

6. Decay-Photon and Decay-Heat Data. Many nuclear reactions leave radioactive products. These products may be simple daughter isotopes that decay in a few steps to a stable final state with the emission of a few photons and electrons (or heat), or they may be a complex array of fission product isotopes that emit a complex spectrum of photons and electrons (or heat) showing many time constants. Procedures to store these data in MATXS libraries will be included in a future version of this report.

E. User Input

The user input specifications below were copied from the comment cards at the beginning of the MATXSR module. It is always a good idea to check the comment cards in the current version of the code for possible changes.

```
*---INPUT SPECIFICATIONS (FREE FORMAT)--------------------------*
*                                                               *
* CARD 1 UNITS                                                  *
*   NGEN1     INPUT UNIT FOR DATA FROM GROUPR                   *
*   NGEN2     INPUT UNIT FOR DATA FROM GAMINR                   *
*   NMATX     OUTPUT UNIT FOR MATXS                             *
*             (OUTPUT UNIT IS ALWAYS BINARY)                    *
*   NGEN3     INCIDENT PROTON DATA FROM GROUPR (DEFAULT=0)      *
*   NGEN4     INCIDENT DEUTERON DATA FROM GROUPR (DEFAULT=0)    *
*   NGEN5     INCIDENT TRITON DATA FROM GROUPR (DEFAULT=0)      *
*   NGEN6     INCIDENT HE3 DATA FROM GROUPR (DEFAULT=0)         *
*   NGEN7     INCIDENT ALPHA DATA FROM GROUPR (DEFAULT=0)       *
* CARD 2 USER IDENTIFICATION                                    *
*   LPRINT    0/1 MEANS NO PRINT/PRINT                          *
*   IVERS     FILE VERSION NUMBER (DEFAULT=0)                   *
*   HUSE      USER ID (UP TO 16 CHARACTERS, DELIMITED BY *,     *
*             ENDED BY /) (DEFAULT=BLANK)                       *
* CARD 3 FILE CONTROL                                           *
*   NPART     NUMBER OF PARTICLES FOR WHICH GROUP               *
*                STRUCTURES ARE GIVEN                           *
*   NTYPE     NUMBER OF DATA TYPES IN SET                       *
*   NHOLL     NUMBER OF CARDS TO BE READ FOR HOLLERITH          *
*             IDENTIFICATION RECORD.                            *
*   NMAT      NUMBER OF MATERIALS DESIRED                       *
* CARD 4 SET HOLLERITH IDENTIFICATION                           *
*   HSETID    HOLLERITH IDENTIFICATION OF SET                   *
*             (EACH LINE CAN BE UP TO 72 CHARACTERS,            *
*             DELIMITED WITH *, ENDED BY /)                     *
* CARD 5 PARTICLE IDENTIFIERS                                   *
*   HPART     HOLLERITH IDENTIFIERS FOR PARTICLES               *
*             (UP TO 8 CHARACTERS EACH)                         *
```

```
* CARD 6 ENERGY GROUPS                                          *
*   NGRP      NUMBER OF GROUPS FOR EACH PARTICLE                *
* CARD 7 DATA TYPE IDENTIFIERS                                  *
*   HTYPE     HOLLERITH IDENTIFIERS FOR DATA TYPES              *
*             (UP TO 8 CHARACTERS EACH)                         *
* CARD 8 INPUT PARTICLE IDS                                     *
*   JINP     INPUT PARTICLE ID FOR EACH DATA TYPE               *
* CARD 9 OUTPUT PARTICLE IDS                                    *
*   JOUTP    OUTPUT PARTICLE ID FOR EACH DATA TYPE              *
* CARD 10 MATERIAL DATA (ONE CARD PER MATERIAL)                 *
*   HMAT      HOLLERITH MATERIAL IDENTIFIER                     *
*             (UP TO 8 CHARACTERS EACH)                         *
*   MATNO     INTEGER MATERIAL IDENTIFIER                       *
*             (ENDF/B MAT NUMBER)                               *
*   MATGG     MAT NUMBER FOR PHOTOATOMIC DATA                   *
*             (DEFAULT=100*(MATNO/100) AS IN ENDF-6)            *
*                                                               *
*****************************************************************
```

Card 1 is used to specify the units for the input GENDF tapes and the output MATXS library. As usual, the sign of a GENDF unit number is used to determine its mode; negative numbers mean binary, and positive numbers mean coded (that is, ASCII, EBCDIC, etc.). The output file is always binary, and its sign is ignored. The most common MATXSR runs are for neutron and photon data only. In these cases, card 1 can be truncated after the `NMATX`. If incident charged-particle GENDF files are available, any of the units `NGEN3` through `NGEN7` can be assigned.

Card 2 is used to control the MATXSR print option `IPRINT` and to provide the information for the MATXS "User Identification" record. An example for Card 2 follows:

```
0  17  *T2 LANL NJOY*/  .
```

The delimiting * characters and the terminating / are required (see the description of free-form input in the NJOY section of this report). Card 3 is used to input the counts for the "File Control" record and to tell the user input routine how many quantities to read in subsequent input cards. Card 4 is repeated `NHOLL` times to give the Hollerith description for the library. Each line must be delimited by * characters and terminated by /. For example,

```
* MATXS17                                27 FEB 91 */
* 69-GROUP THERMAL LIBRARY FROM ENDF/B-VI          */
*   THIS LIBRARY INCLUDES NEUTRON, PHOTON, AND     */
*   THERMAL SCATTERING DATA FOR 133 MATERIALS.     */
```

The lines are written out on the MATXS file as given, except that the maximum length for each line is 72 characters.

Card 5 is used to read in the Hollerith names for the NPART particles for this library. The standard names for the particles were given above in Table 1. The number of groups desired for each particle are given on card 6. The data-type names (see Table 2) are given on Card 7. Cards 8 and 9 are used to specify the input and output particles for each data type. The user should take care that the number of groups given for each particle is consistent with the input data, and that the particle assignments to data type names (JINP and JOUTP) are consistent with the names. The code does not check.

Card 10 is repeated NMAT times to specify the materials to be written out on the library. The rules for constructing the material names HMAT were given in Section B. The MATNO parameter is the ENDF MAT number for this material as used on the input GENDF files. For ENDF/B-VI, the MAT number is the same for all sublibraries (that is, for incident neutrons, photons, protons, etc.), and only one value is needed to specify the desired material. However, photoatomic data are atomic in character, and the MAT numbers always refer to the element. For example, MAT=2600 for the photoatomic data of iron. MATXSR reads a second MAT number field, MATGG, for the photoatomic data. Its default value is given by

```
100*(MATNO/100)  ,
```

and it is, therefore, not usually needed. The normal form for card 10 would be as follows:

```
FE56  2631/  .
```

However, the photoatomic libraries for earlier versions of the ENDF/B files used MAT numbers like 26 for elements. The MATGG parameter can be used to process data from these older libraries. Note that delimiting * characters are not required for Hollerith names that are single words and start with a letter.

MATXSR always processes all submaterials found on all the nonzero GENDF units NGEN1 through NGEN7. It processes every temperature and σ_0 value found. It processes (almost) every reaction cross section and matrix found, and it processes every Legendre order given. The only way to control the contents of the MATXS library is through the input to GROUPR or GAMINR and by the materials included when building the input GENDF files. A convenient way to handle this task is to assemble the results of a number of single-material GROUPR runs into composite GENDF tapes using MODER.

F. Coding Details

The main container array for dynamic storage allocation is defined by the statement "COMMON/MSTORE/A(35000)," and its length is specified in the statement "ISIZA=35000." It may be necessary to increase this allocation if very large group structures are used. The code will issue error messages when more storage is needed. The dynamic array DUM is allocated with two words of storage. If the resulting pointer L0 is even, one of these two words is released. This assures that the first real array reserved will have an odd pointer, as required for proper word alignment on short-word machines.

The values of MULT and NCW change between short-word machines (32 bits) and long-word machines (60–64 bits). Also note that all Hollerith variables (which have "h" as their initial letter) are implicitly REAL*8 double-precision variables on short-word machines. The role of MULT is discussed in the CCCCR section of this report. It is used for calculating pointers and the lengths of arrays for dynamic arrays and common blocks that contain mixtures of Hollerith, real, and integer variables. The parameter NCW contains the number of Hollerith characters per word for the target machine (either 4 or 8). It is used by FREE (see the NJOY section of this report).

The next step in subroutine MATXSR is to read in the unit numbers for the input and output files and to open the files. Note that the sign for NMATX is ignored. It will always be binary. Several binary scratch files are also opened. Next, the user input for the "File Identification" record is read (see card 2 in the user input description), and the CMATXS routine is called to read the input GENDF files and construct the output MATXS file. When CMATXS returns, the main routine checks whether a listing of the new library is desired (LPRINT>1). Then, it calls INDEX to prepare an index listing summarizing all the materials, submaterials, data types, and reactions given in the new library. Finally, MATXSR closes the files, writes a summary, and returns to the main loop over modules in the NJOY program.

Subroutine CMATXS starts by defining three constants:

NSUBMX=100 is the maximum number of submaterials allowed for a material, including data types, temperatures, and background cross sections;

MAXORD=5 is the maximum Legendre order (that is, P_5 allowed for group-to-group matrices; and

MAXW=5000 is the maximum size for vector block and matrix subblock records on the MATXS file.

It continues by calling RUINM to read the user's input (see the "User Input" section above), and then it calls `MTXDAT` to read the input GENDF data and write it to the scratch tapes used later in this subroutine.

The first step in `MTXDAT` is to reserve dynamic storage space for the `NPART` group structures. Note that the number of words reserved is forced to be even in order to avoid possible word-alignment problems on short-word machines. With the current version of GROUPR, all particles are assumed to use the same group structure. It is only necessary to look through the unit numbers from input card 1 and find the first one that contains data. The group structures are then read in from MF1,MT451, reversed to have the conventional decreasing-energy order, and stored into the `GRUP` array. The subroutine now reserves space for the "Material Control" record (see `MATC` and `IMATC`) and begins the material loop (see `DO 700 IM=1,NMAT`). Note how the Hollerith material name is loaded into the "Material Control" record storage area by making use of the `MULT` parameter and the equivalence between the arrays `A` and `HA`.

The first loop inside the material loop is the data-type loop (see `DO 600 I=1,NTYPE`). The same MAT number is used for all particles, but a different one is used for photoatomic data. The input unit to be used depends on the identity of the incident particle. Since these comparisons use constants like `HPROT` with values like "`6HP    `," it is important to use the standard names in the user input (see Table 1). The identity of the outgoing particle for the data type determines the MF number for the matrices on the GENDF tape, `MFM`. As described in the GROUPR section of this report, the MF assignments are as follows:

MF value	Outgoing Particle
6	neutrons
16	photons
21	protons
22	deuterons
23	tritons
24	^{3}He particles
25	alphas
26	photoatomic data

The MF number for the vector data (`MFV`) is 3, except for the photoatomic case, where it is 23.

The next step is to search for the desired material `IMAT` on the current input tape. When the first occurrence of `IMAT` has been found, `CMATXS` proceeds to copy the data for the first temperature to the two scratch files `NSCR` and `IREF`. File `IREF` will be used for the higher temperature and σ_0 values. At the same time, MTXDAT counts the number of one-dimensional reactions, NONED, and the number of two-dimensional reactions, NTWOD, for each value of the background cross section, σ_0. When the scratch tapes are complete, the subroutine starts a loop over σ_0 for this material and temperature (this loop goes through statement number 440), and it calls VECIN, VECOUT, and MATRIX to process the vector and matrix data for this σ_0. As each σ_0 is finished, it loads the parameters for the current submaterial into the "Material Control" record and jumps back to statement number 440 to continue the loop. When all the σ_0 values for this temperature have been processed, it jumps back to statement number 300 to get the next temperature. Finally, it reaches statement number 600 and goes back through the entire process again for the next data type. When the "DO 600" loop exits, the entire "Material Control' block has been filled in, and the subroutine writes the record to scratch file NSCRT6.

At this point, the processing for material IMAT is complete. The "DO 700" loop continues until all the requested materials have been found and processed.

Subroutine FINDG is used to search GENDF file NTAPE for a specified section (MAT, MF, MT). If it does not find the requested section, a fatal error message is issued. Materials (MAT) do not have to be in order on a GENDF tape. Therefore, this routine simply reads forward until it comes to the first record matching the requested MAT, MF, and MT values. If the section is not found, it rewinds once and searches again. When the section is found, it backspaces by one record so that the next record read will be the desired record.

Subroutine HNAME is used to construct a Hollerith reaction name HREACT from an ENDF MT number (MT), an ENDF "LR flag" (LR), and an incident particle name (HP). The definitions of the names were discussed above in connection with Tables 3–11, and the actual MT numbers and strings used are given in data statements in this subroutine. If no preset name is found for an MT number, a default name of the form MTnnn is constructed.

Subroutine VECIN reads the cross section vectors for one submaterial (that is, one data type, one temperature, and one σ_0). It starts by clearing some flags that are used to detect the occurrence of particular reactions. For example, K107 is set if the (n,α) reaction (MT107) is found. Dynamic storage arrays are reserved for the vector control and vector data arrays (see VCON, IVCON, VDAT, and IVDAT). Note that there must be enough container memory available to hold all the vector data (NING*N1D words). The routine now starts up a loop over sections on the input scratch tape (the loop goes through statement number 115). The sections that it processes are determined by the value of MFD, which will be 3, except for photoatomic data, when it will be 23. Delayed fission χ is a special case; it will be found in the section with an MT value of `MFD+2`. As each interesting section is found, the Hollerith name for the reaction is constructed, and the location of the reaction in `VDAT` is computed. When the total cross section is found (MT=1), names and locations are set up for the P_ℓ components of the weight function (for example, `NWT0` and `NWT1`) and for the P_ℓ components of the total cross section (`NTOT0` and `NTOT1`). If appropriate, the first letter of these names might be `P`, `D`, etc., depending on the incident particle. When the total photoatomic cross section is found (MT=501), names and locations are defined for the γ weight and the P_0 cross section (`GWT0` and `GTOT0`). When the charged-particle elastic cross section is found for particle `x` (MT2), names and locations are set up for the weight function `xWT0` and the effective charged-particle total cross section `xTOT0`. Once the name and location have been selected, the code reads the data for all groups from file `NSCR` and stores them at the selected location. If this submaterial corresponds to a higher temperature or σ_0, the corresponding data is also read from file `IREF` and subtracted from the `NSCR` data. As a result, self-shielding information in the MATXS file is given as differences rather than f-factors as in the previous CCCC self-shielding file, BRKOXS. The delayed-neutron spectra are read from MF5,MT455; the spectra from the six time groups on the GENDF tape are added to obtain a single spectrum χ_d.

There is a special feature in `VECIN` for incident charged-particle reactions. Although discrete-ordinates transport codes require a total cross section, it is not actually possible to compute a total cross section for charged-particle reactions because Coulomb scattering is singular for straight-ahead scattering (at least in the absence of electronic screening). The solution to this problem used by GROUPR is to exclude a range of forward angles from the charged-particle elastic matrix (this angular range is handled by continuous-slowing-down theory in the application codes). Therefore, MATXSR can construct an effective total cross section by adding all the reaction cross sections found to this truncated elastic cross section. However, it must be careful to watch for possibly redundant cross sections that must be omitted from the sum; for example, if the total (x,p) cross section (MT103) is present for incident particle x, the discrete-level (x,p_n) cross sections and the (x,p) continuum reactions (MT600–649) must be omitted from the sum. This is the function of flags like `K103`.

Subroutine `VECOUT` thins the vector data loaded by `VECIN` by removing unneeded leading or trailing zero from each reaction. These zeros can arise from thresholds and upper limits for thermal-range reactions. In addition, since data for the higher temperatures and σ_0 values are stored as differences, zero (or very small) elements can imply that there is no significant temperature or σ_0 dependence. The subroutine simply steps through the VDAT data block to find the band limits NFG and NLG for each reaction, to squeeze out the zeros, and to record the band limits in the vector control block VCON. When all of the N1D reactions have been processed, the resulting vector data is written out as a series of "Vector Block" records on scratch tape NSCRT3 (see the block of coding starting at statement number 290). The subblocking is controlled by MAXW (normally 500 words). The code just steps through the cross section bands until the next reaction will cause the sum to exceed MAXW. It then writes these reactions to the scratch file, and it repeats the process for the next group of reactions. Each record will have fewer than MAXW words, and the minimum number of records will be produced within the restriction that no reaction is broken up between records. The last step in the subroutine is to write the "Vector Control" record for this submaterial to scratch file NSRT2.

Subroutine MATRIX reads the group-to-group matrix data for this submaterial, converts it to MATXS format, and writes the results to a scratch file. The dynamic storage arrays used include MCON to accumulate matrix control information, CDAT for the low-energy constant spectrum data (if any), JGLL for banding information, and MDAT for the actual matrix data. The main loop is over matrix reaction types (see "DO 280 I=1,N2D"). The head card for the reaction is read in from NSCR, and for higher temperatures and σ_0 values, from IREF also. Subroutine HNAME is called to construct the Hollerith reaction name, subroutine BAND is called to read through the data and determine the banding parameters used to remove excess zero elements, and subroutine SHUFL is called to read through the input scratch file again and load the data into memory in its final compact form.

Subroutine BAND is fairly simple. It loops over all the groups for this reaction on NSCR and/or IREF (the loop goes through statement number 110). When the loop is finished, JGLLO(I) and JGLHI(I) contain lowest and highest initial-group indices found for each final group I. Note that the actual cross section for a matrix element (or the cross section difference when IREF is being used) is not checked; it is assumed that GROUPR has done a good job removing excess zeros. This should be improved in a future version. Before returning, these limiting group indices are used to compute the MATXS parameters JBAND and IJJ, and to insert them into the accumulating matrix control block (see IMCON).

Subroutine SHUFL starts by initializing a loop through statement number 100 that will be used to produce "Matrix Sub-Block" records. Inside this loop, the code sums the bandwidths found by BAND to find out how many final-energy group bands can fit in a subblock with length less than or equal to `MAXW` words (see the `DO 101` loop). The result is a pair of group indices `IO` and `IMAX` that define the range of groups to be included in the subblock. `SHUFL` now makes a pass through scratch file `NSCR`, and also through scratch file `IREF` for higher temperatures and σ_0 values. It stores away the constant subblock data (`IG=0` for the constant spectrum and `IG2LO=0` for the production cross section), if found. For each group-to-group element, it computes the final group index `JG2`. If the group is in the allowed range for this subblock, it computes the output location `NOLOC` and stores the element in memory. For higher temperatures and σ_0 values, it subtracts the base value found on file `IREF`. (This step is normally needed for neutron elastic scattering only.) When the group loops for `NSCR` and `IREF` are complete, the subroutine writes out the "Matrix Sub-Block" record on a scratch file, resets the `NG2Z` parameter to the top group of the subblock, and continues the loop through statement number 100 to produce the rest of the subblocks for this reaction. The resulting series of subblock records on the scratch file are all less than `MAXW` words in length, and the number of records is minimal within the restriction that bands are not split between records.

Returning to `MATRIX`, data already in memory are used to complete the "Matrix Control" record, which is written out onto `NSCRT2`. If no matrix data were found by `BAND` and `SHUFL`, a dummy matrix control record is written on `NSCRT2` and a dummy matrix data record is written on `NSCRT3`. Finally, the constant subblock data saved in `SHUFL` (if any) is written to `NSCRT3`.

Subroutine `PMATXS` is a simple routine that reads through an input file `NIN` in MATXS format and prepares an interpreted listing on `NOUT`. This subroutine is also available in the auxiliary code BBC.

Subroutine `INDEX` reads an input file `NIN` in MATXS format and prepares an index listing that gives the materials, data types, temperatures, background cross sections, vector reaction types, and matrix reaction types found on the library. This subroutine is also available in BBC.

G. Error Messages

The fatal-error and warning messages generated by MATXSR are given below, along with suggested actions to alleviate the problem.

`ERROR IN MTXDAT***INPUT ERROR (NIN=0)`

No input GENDF tape has been given. Check Card 1 of the user input.

`MESSAGE FROM MTXDAT---NO DATA FOR ITYPE=nn MAT=nnnn ITEMP n`

The input GENDF tapes do not have to contain data for all the materials requested by the user with the `HMAT` values. This diagnostic message warns the user that some potential materials were not found. In some cases, the user will want to modify the input data if the warning is a surprise!

`ERROR IN MTXDAT***TOO MANY SUBMATERIALS`

The code is currently limited to 100 submaterials per material. See `NSUBMX=100` in `CMATXS`.

`ERROR IN FINDG***MAT OR MF OR MT LE 0 NOT ALLOWED`

Subroutine `FINDG` has been asked to search for an illegal section on the GENDF tape.

`ERROR IN FINDG***MAT=nnnn MF=nn MT=nnn NOT ON TAPE`

The requested section was not found on the input GENDF tape. Check that the correct tape was mounted.

`ERROR IN BAND***INPUT TOO LARGE`

There is not enough room in memory to read the matrix record for one group from the GENDF tape. The only solution is to increase the size of the main container array. See `COMMON/MSTORE/A(35000)` and `ISIZA=35000` at the beginning of the `MATXSR` routine.

`ERROR IN SHUFL***INPUT TOO LARGE`

See comments above.

`ERROR IN INDEX***STORAGE EXCEEDED`

This error should not occur for the version of `INDEX` that is incorporated into MATXSR because the code was able to write the records in the first place!

`ERROR IN LST1IO***STORAGE EXCEEDED`

See comments above.

H. References

1. B. M. Carmichael, “Standard Interface Files and Procedures for Reactor Physics Codes, Version III,” Los Alamos Scientific Laboratory report LA-5486-MS (February 1974).

2. R. Douglas O’Dell, “Standard Interface Files and Procedures for Reactor Physics Codes, Version IV,” Los Alamos Scientific Laboratory report LA-6941-MS (September 1977).

3. R. E. MacFarlane, “TRANSX-CTR: A Code for Interfacing MATXS Cross-Section Libraries to Nuclear Transport Codes for Fusion Systems Analysis,” Los Alamos National Laboratory report LA-9863-MS (February 1984).

4. R. E. MacFarlane, “TRANSX 2: A Code for Interfacing MATXS Cross-Section Libraries to Nuclear Transport Codes,” Los Alamos National Laboratory report LA-12312-MS (July 1992).

5. K. D. Lathrop, "DTF-IV, A FORTRAN Program for Solving the Multigroup Transport Equation with Anisotropic Scattering," Los Alamos Scientific Laboratory report LA-3373 (November 1965).

6. R. E. MacFarlane, "Energy Balance of ENDF/B-V," *Trans. Am. Nucl. Soc.* 33, 681 (1979). See also R. E. MacFarlane, "Energy Balance of ENDF/B-V.2," Minutes of the Cross Section Evaluation Working Group (available from the National Nuclear Data Center, Brookhaven National Laboratory, Upton, NY) (May 1984).

7. Y. Farawila, Y. Gohar, and C. Maynard, "KAOS/LIB-V: A Library of Nuclear Response Functions Generated by KAOS-V Code From ENDF/B-V and Other Data Files," Argonne National Laboratory report ANL/FPP/TM-241 (April 1989).

8. I. I. Bondarenko, Ed., *Group Constants for Nuclear Reactor Calculations* (Consultants Bureau, New York, 1964).

CONTENTS