

LA-UR-08-07182

*Approved for public release;
distribution is unlimited.*

Title: MCNPX 2.7.A Extensions

Authors: Denise B. Pelowitz
John S. Hendricks
Joe W. Durkee
Michael R. James
Michael L. Fensin
Gregg W. McKinney
Stepan G. Mashnik
Laurie S. Waters

Los Alamos National Laboratory

Intended for: <http://mcnpx.lanl.gov>



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

MCNPX 2.7.A Extensions

by

Denise B. Pelowitz
John S. Hendricks
Joe W. Durkee
Michael L. Fensin
Michael R. James
Gregg W. McKinney
Stepan G. Mashnik
Laurie S. Waters

Los Alamos National Laboratory

November 6, 2008

Copyright 2008. Los Alamos National Security, LLC. All rights reserved.

This material was produced under U.S. Government contract DE-AC52-06NA25396 for Los Alamos National Laboratory, which is operated by Los Alamos National Security, LLC for the U.S. Department of Energy. The Government is granted for itself and others acting on its behalf a paid-up, nonexclusive, irrevocable worldwide license in this material to reproduce, prepare derivative works, and perform publicly and display publicly. Beginning five (5) years after (insert date permission to assert copyright was obtained), subject to additional five-year worldwide renewals, the Government is granted for itself and others acting on its behalf a paid-up, nonexclusive, irrevocable worldwide license in this material to reproduce, prepare derivative works, distribute copies to the public, perform publicly and display publicly, and to permit others to do so.

NEITHER THE UNITED STATES NOR THE UNITED STATES DEPARTMENT OF ENERGY, NOR LOS ALAMOS NATIONAL SECURITY, LLC, NOR ANY OF THEIR EMPLOYEES, MAKES ANY WARRANTY, EXPRESS OR IMPLIED, OR ASSUMES ANY LEGAL LIABILITY OR RESPONSIBILITY FOR THE ACCURACY, COMPLETENESS, OR USEFULNESS OF ANY INFORMATION, APPARATUS, PRODUCT, OR PROCESS DISCLOSED, OR REPRESENTS THAT ITS USE WOULD NOT INFRINGE PRIVATELY OWNED RIGHTS.

TABLE OF CONTENTS

1	INTRODUCTION	1
1.1	Summary of New MCNPX Features	1
1.2	Availability	1
1.3	Documentation	2
1.4	Cash Awards Quality Assurance Program	2
2	DESCRIPTION OF NEW MCNPX FEATURES	3
2.1	Tally Tagging	3
2.1.1	User Interface	3
2.1.2	Usage and Special Cases	4
2.1.3	Examples	5
2.2	CEM 3.02 Upgrade	6
2.3	MCNPLOT Graphics and "HELP" Command Upgrades	8
2.3.1	Graphics Upgrades	8
2.3.2	"HELP" Command Upgrade	10
2.4	Embedded Sources	12
2.4.1	User Interface	12
2.4.2	Examples	12
2.5	Stopping Power Energy Table Control	15
3	MCNPX 2.7.A FEATURE EXTENSIONS AND ENHANCEMENTS	15
3.1	Spontaneous Photon Automatic Weight Adjustment	16
3.2	Burnup Enhancements	18
3.3	Lattice FILL Error Trap	19
3.4	LINUX Static Build	19
3.5	Compilation without Graphics	19
3.6	Minor Enhancements:	20
4	MCNPX CORRECTIONS	20
4.1	Significant Problem Corrections	20
4.1.1	Minor Burnup Reaction Rates Off	20
4.1.2	Spontaneous Fission Normalization Error	20
4.1.3	DXTRAN Fail with Repeated Structures	20
4.1.4	Heavy Ion Errors	21
4.1.5	S(α,β) File Reading Failure	21
4.1.6	Burnup Continue Runs	21
4.2	Minor Problem Corrections	21
4.2.1	INTEL Compiler Workaround	21
4.2.2	LAQGSB Corrections	21
4.2.3	Compiler Workaround	21

- 4.2.4 Detector Tallies in Lattice Geometries.....22
- 4.2.5 Burnup Duplicate ZAIDs22
- 4.2.6 Source Particle Types Unrecognized in Dependent Distributions.....22
- 4.2.7 Incorrect Warning Error for Detector Tallies22
- 4.2.8 Choice of High-Energy Mode.....22
- 4.2.9 Lost Particles Cause Crash22
- 5 FUTURE WORK22**

MCNPX 2.7.A Extensions

ABSTRACT

MCNPX is a Fortran90 Monte Carlo radiation transport computer code that transports nearly all particles at nearly all energies for nearly all applications. The new capabilities of the latest version, MCNPX 2.7.A, beyond those in MCNPX 2.6.0, are described.

1 INTRODUCTION

MCNPX (MCNP eXtended) is a Fortran90 (F90) Monte Carlo radiation transport computer code that transports nearly all particles at nearly all energies. It is a superset of MCNP4C3 and has many capabilities beyond MCNP4C3. MCNPX is a production computer code that models the interaction of radiation with matter. Its quality is highly regarded and cash awards (described in Section 1.4) are offered for bugs.

1.1 Summary of New MCNPX Features

The new capabilities and enhancements of MCNPX 2.7.A beyond MCNPX 2.6.0 are listed below. (Where applicable, the initials of the principal developers are shown in parentheses.¹) Note that the delayed-gamma emission and criticality source convergence acceleration capabilities are developmental. They are not backed by the MCNPX cash awards quality assurance program and must be used cautiously. However, these capabilities are considered of such significance that they are included. Recall also that as of version 2.6.0 total v (not prompt v) is the default for fission neutrons. The new MCNPX 2.7.A capabilities include the following:

- Tally tagging (GWM)
- CEM 3.02 upgrade (MRJ/SGM)
- Plot appearance upgrades (JWD)
- Embedded sources (JSH)
- Stopping power energy table control (GWM)

1.2 Availability

The latest general-release version is MCNPX 2.6.0, which is available from the Radiation Safety Information Computational Center (RSICC)² and the Office of Economic Cooperation and Development/Nuclear Energy Agency (OECD/NEA) in Paris, France.³

¹ John S. Hendricks (JSH, LANL, X-3-MCC); Gregg W. McKinney (GWM, LANL, X-3-MCC); Joseph W. Durkee (JWD, LANL, D-5); Jay S. Elson (JSE, LANL, D-5); Michael L. Fensin (MLF, LANL, N-4); Michael R. James (MRJ, LANL, D-5); Russell C. Johns (RCJ, LANL, D-5); Stepan G. Mashnik (SGM, LANL, X-3-MCC); Denise B. Pelowitz (DBP, LANL, D-5); Laurie S. Waters (LSW, LANL, D-5); Franz X. Gallmeier (FXG, ORNL)

² <http://rsicc.ornl.gov/>

³ <http://www.nea.fr/>

For approved users, beta test program versions may be downloaded from the MCNPX website.¹ Approximately 2000 MCNPX beta test users are an essential part of the MCNPX software quality assurance program. In general, to become a new beta test user usually requires being an MCNPX sponsor, collaborator, or U.S. citizen participant in an MCNPX workshop training course. For more information, contact us at the following address: mcnpx@lanl.gov.

MCNPX is U.S. export controlled. Generally, MCNPX is easy to obtain for legitimate purposes, but restrictions protect the code, data, and documentation once it is acquired.

1.3 Documentation

Restricted Documentation:

Complete MCNPX documentation is available in the “MCNPX™ User’s Manual,”² which comes with MCNPX from RSICC or OECD/NEA and is available to MCNPX beta testers from the Worldwide Web (WWW).³

Unrestricted Documentation:

Documentation of MCNPX features in MCNPX 2.6.0—that is, all MCNPX features beyond MCNPX 2.3.0 and MCNP4C—is provided in “MCNPX Extensions, Version 2.6.0.”⁴

MCNPX capabilities developed since MCNPX 2.6.0 are documented in this report.

All of the extended MCNPX capabilities are summarized in the one-page MCNPX features list.⁵

1.4 Cash Awards Quality Assurance Program

We are so confident in the quality of MCNPX that John Hendricks offers cash awards to the first person who finds any bug in the code. He will pay \$20 to the first person finding anything that does not meet or exceed the capabilities of MCNPX 2.3.0 and MCNP4C3. European users will be awarded €20. He also will pay a brand new \$2 bill for any error in MCNPX that has been inherited from its constituent codes.⁶ A list of all winners since June 2001 can be found on the MCNPX WWW site.⁷

¹ <http://mcnpx.lanl.gov/beta/>

² Denise B. Pelowitz, ed., “MCNPX User’s Manual, Version 2.6.0,” Los Alamos National Laboratory report, LA-CP-07-1473 (April 2008), http://mcnpx.lanl.gov/docs/MCNPX_2.6.0_Manual.pdf/.

³ <http://mcnpx.lanl.gov/>.

⁴ <http://mcnpx.lanl.gov/opendocs/reports/Interface.pdf/>.

⁵ <http://mcnpx.lanl.gov/opendocs/misc/FeaturesList.pdf/>

⁶ Cash Award Fine Print: This offer is subject to cancellation or modification without notice. A bug is defined as an error we choose to correct in the source code. We make awards even for the most trivial or insignificant of problems, but not for proposed code enhancements or proposed extended capabilities. Awards are given only to the first MCNPX user reporting a problem. MCNPX development team members are ineligible. Reported problems must be reproducible, and awards are paid when the correction is integrated into a forthcoming MCNPX version. We endeavor to make MCNPX the most error-free and robust Monte Carlo radiation-transport code possible, and we back this code with a cash award.

⁷ <http://mcnpx.lanl.gov/opendocs/misc/Cashbugs.pdf/>.

2 DESCRIPTION OF NEW MCNPX FEATURES

Note that the delayed-gamma emission and criticality source convergence acceleration capabilities are developmental. These capabilities are not backed by the MCNPX cash awards program and must be used cautiously. Nevertheless, they are considered of such significance that they are included.

The principal new capabilities of MCNPX beyond MCNPX 2.6.0 described in detail in this section are

- Tally tagging
- CEM 3.02 upgrade
- Plot appearance upgrades
- Embedded Sources
- Stopping power energy table control

2.1 Tally Tagging

Tally tagging, a new feature in v.2.7.a, provides the ability to separate a tally into components based on how and where the scoring particle was produced. This feature is available for both standard and detector tallies; however, it is only implemented for neutron, photon, and electron tallies.

2.1.1 User Interface

Tally tagging is specified by using the new keyword, TAG, with its associated parameter, a , on the FT n tally special treatment card, which is associated with a standard F n tally. The format is

```
FT $n$  TAG  $a$ 
```

where n is the neutron or photon tally number, and a specifies how scatter is to be treated (i.e., whether the creation tag on a particle should be retained or a separate scatter tag be invoked):

- $a=1$ indicates that all collided particles will lose their tag and that bremsstrahlung and annihilation photons will be included in the bin of collided particles;
- $a=2$ indicates that all collided particles will lose their tag, but that bremsstrahlung and annihilation photons will be given special tags that allow them to be segregated; and
- $a=3$ indicates that all collided particles will retain their production tag.

Binning specifications for the tagged tally must be provided on the FU special tally card. Each bin_i (described below) requests three distinct pieces of tagging information:

- 1) a cell of interest where particles are produced;
- 2) a target nuclide from which the particle is emitted; and
- 3) a reaction or, in the case of spallation, a residual nuclide of interest.

The format on the FU card when used in association with the tagging treatment is

```
FUn  $bin_1$   $bin_2$  ...  $bin_N$ 
```

where each tagging bin_i has the form CCCCCZZAAA.RRRRR. A user cell number is designated by CCCC (leading zeros are not required); ZZAAA represents a five-digit isotope

identifier for a target nuclide, where *ZZ* represents the atomic number and *AAA* represents the atomic mass number and *RRRRR* specifies a reaction identifier for library interactions or a residual nuclide *ZZAAA* identifier for high-energy model interactions. Special cases for the *CCCCZZAAA.RRRRR* *FU* card entries are provided in Section 2.2.2. Note that the *ZZ* atomic number specification is limited to two characters; therefore nuclides with $Z > 99$ cannot be tagged. By default, a total over all specified bins is provided automatically for the *FU* special tally. To suppress the generation of a total, add the “*NT*” parameter after the last specified bin.

2.1.2 Usage and Special Cases

If cell tagging is not desired, the *CCCC* portion of the tag should be omitted or, alternatively, set to “00000”. In either case, tally contributions will be accumulated for all cells for that *FU* bin, provided the *ZZAAA.RRRRR* portion of the tag is satisfied. In the case of particle production from electrons, which are material based (not nuclide specific), the *CCCC* input should be used to identify the cell and the *ZZAAA* input should be set to “00000”. The suffix *RRRRR* refers to a standard ENDF reaction number for library interactions (e.g., “00102” stipulates [n,gamma]) or, in the case of high-energy model interactions, *RRRRR* refers to a residual nuclide *ZZAAA* identifier (e.g., “06012” for ¹²C). Additional library-related reactions numbers can be found in Appendix G of the MCNPX User’s Manual.

In general, a zero input for any portion of the tag results in the sum of all contributions related to the entry. For example, the tag “0000092000.00000” will collect all tally contributions where any isotope of uranium ($z=92$) had produced the particle making the tally. However, the tag “0000000000.00000” is reserved for scattered particles. Note that each tally contribution is made only to the first *FU* bin that satisfies the tag description (i.e., those that have not already been tallied). If no appropriate *FU* bin is found, the tally contribution is not made; however a special “everything else” *bin_i* (i.e., “1e10”) can be specified to collect any portion of the tally that falls into no other bin. When the “everything else” bin is used, then the user is assured that the “user-bin total” bin will reproduce the original tally as if the *FTn TAG* option has not been used.

Special designations for *CCCCZZAAA*:

-0000000001	or	-1	source particle tag for all cells
-CCCC00001			source (i.e., uncollided) particle tag for cell <i>CCCC</i>
0000000000	or	0	scattered particle tag
10000000000	or	1e10	everything else tag

Photon tally special designations for *ZZAAA.RRRRR*:

00000.00001	bremsstrahlung from electrons
ZZ000.00003	fluorescence from nuclide <i>ZZ</i>
00000.00003	K x-rays from electrons
00000.00004	annihilation photons from e^-
ZZ000.00005	Compton photons from nuclide <i>ZZ</i>
ZZAAA.00006	muonic x-rays from nuclide <i>ZZAAA</i>

Electron tally special designations for *ZZAAA.RRRRR*:

ZZ000.00001	photoelectric from nuclide <i>ZZ</i>
ZZ000.00003	Compton recoil from nuclide <i>ZZ</i>

ZZ000.00004	pair production from nuclide ZZ
ZZ000.00005	Auger electron from nuclide ZZ
00000.00005	Auger electron from electrons
00000.00006	knock-on electrons

Neutron and photon tally special designations for ZZAAA.RRRRR:

ZZAAA.99999	delayed particles from fission of ZZAAA
-------------	-----------------------------------------

2.1.3 Examples

Example 1:

```
F1:N 10
FT1 TAG 1
FU1 0000092235.00016 0000092235.0000 1e10
```

If an [n,2n] neutron that is produced from an interaction with ^{235}U contributes to the F1 tally, then its contribution will be included only in the first FU bin even though its tag also will satisfy the criteria for the 2nd FU bin. Thus, the order of the FU bin tags is important for segregating the tally. Note in this example that neutrons produced by some other reaction with ^{235}U will be placed in the 2nd FU bin and neutrons produced by reactions with other target nuclides will be placed in the last (“everything else”) bin. The sum of these three bins should preserve the value of the original F1:N tally.

Example 2:

```
F1:P 1
FT1 TAG 1
FU1 0.0 01001.00102 01001.00000
      26056.00102 26056.00051 26056.00052
      26056.24052 26056.26053 26056.26054 26056.26055
      26056.00000
```

In this case, all collided photons will be put into the FU “0.0” bin. All capture gammas from ^1H will be put into the 01001.00102 bin; all remaining gammas from ^1H interactions will be put into the 01001.00000 bin. All capture gammas from ^{56}Fe will go into the 26056.00102 bin; all [n,n'] 1st level gammas will go into the 26056.00051 bin; all [n,n'] 2nd level gammas will go into the 26056.00052 bin; all deexcitation gammas from the spallation of ^{56}Fe into ^{52}Cr will go into the 26056.24052 bin; etc. All remaining gammas produced from ^{56}Fe interactions will go into the 26056.00000 bin.

Example 3:

```
F5:P 0 0 0 1
FT5 TAG 3
FU5 -1.0 0000106012.00005 0000106012.00000
      0000026056.00102 0000026056.00000
      0000000000.00051
      10000000000.00000
```

In this case, all collided photons will retain their original creation tag. All source photons will go into the -1.0 bin. All Compton photons from ^{12}C in cell 1 will be put into the 2nd bin; all

remaining photons produced from interactions with ^{12}C in cell 1 will go into the 3rd bin. All capture gammas from ^{56}Fe will go into the 4th bin; all remaining photons/gammas produced from interactions with ^{56}Fe will go into the 5th bin. All $[n,n']$ 1st level gammas will be put into the 6th bin, and all remaining photons/gammas that were not included in any of the previous bins will be placed in the last bin.

2.2 CEM 3.02 Upgrade^{1,2}

In versions 03.02 of CEM and LAQGSM, the Fermi breakup model is used at all stages of a reaction for an excited nucleus with atomic mass $A < 13$. Previously, in versions 03.01 of these codes, the Fermi breakup model was used to calculate the disintegration of light nuclei but was not used at the preequilibrium, evaporation, and fission stages for an excited nucleus or a fission fragment with $A < 13$.

Also, the computer routines that describe the Fermi breakup model, written by the original authors in Russia twenty years ago, historically have had some occasional odd behavior. These problems have been corrected in version 03.02. First, no longer do these routines allow the rare production of some light unstable fragments like ^5He , ^5Li , ^8Be , and ^9B as a result the breakup of some light excited nuclei. Also no longer permitted is the very rare production of “neutron stars” or “proton stars,” that is, the production of residual nuclei produced via Fermi breakup that consist of only neutrons or only protons. Code crashes caused by an occasional division by zero within the Fermi breakup routines have been eliminated as have several other coding bugs.

CEM 3.02 was integrated into MCNP6 and MCNPX by H. Grady Hughes and Richard E. Prael. The 03.02 versions of CEM and LAQGSM more correctly describe nuclear reactions on intermediate and light nuclei and the production of fragments heavier than ^4He from heavy targets than did their predecessors. Additionally, these new versions produce very few unstable unphysical final products and are free of the corrected bugs.

The file mollinix.tbl is now in the MCNPX data files and is required for CEM 03.02.

A graphical comparison of Bertini, CEM 03.01, and CEM 03.02 appears in Figures 1 and 2. The plots are results of neutron flux near a proposed tungsten target at the LANL LANSCE facility. The driver for the plotted flux was an 800-MeV proton beam. Overall, the energy spectrum show very good agreement between the CEM versions, but is not as good with Bertini.

¹ S. G. Mashnik, R. E. Prael, and K. K. Gudima, "Implementation of CEM03.01 into MCNP6 and its Verification and Validation Running through MCNP6. CEM03.02 Upgrade," research note X-3-RN(U)-07-03, Los Alamos National Laboratory (2007).

² S. G. Mashnik, K. K. Gudima, R. E. Prael, A. J. Sierk, M. I. Baznat, and N. V. Mokhov, "CEM03.03 and LAQGSM03.03 Event Generators for the MCNP6, MCNPX, and MARS15 Transport Codes," invited lectures presented at the Joint ICTP-IAEA Advanced Workshop on Model Codes for Spallation Reactions, February 4–8, 2008, ICTP, Trieste, Italy, Los Alamos National Laboratory report LA-UR-08-2931 (2008); E-print: arXiv:0805.0751v2 [nucl-th]; IAEA Report INDC(NDS)-0530, Vienna, Austria, August 2008, p. 53.

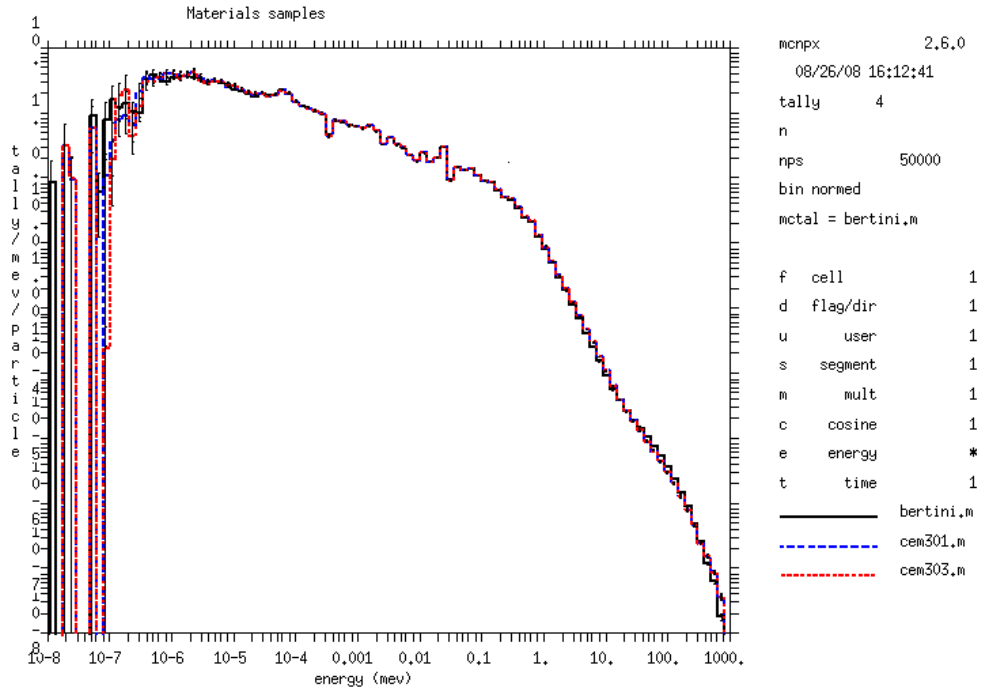


Figure 1. Neutron flux generated by an 800-MeV proton beam in LANSCE material samples. (Note that CEM 03.03 is mislabeled as “cem303” in the plot.)

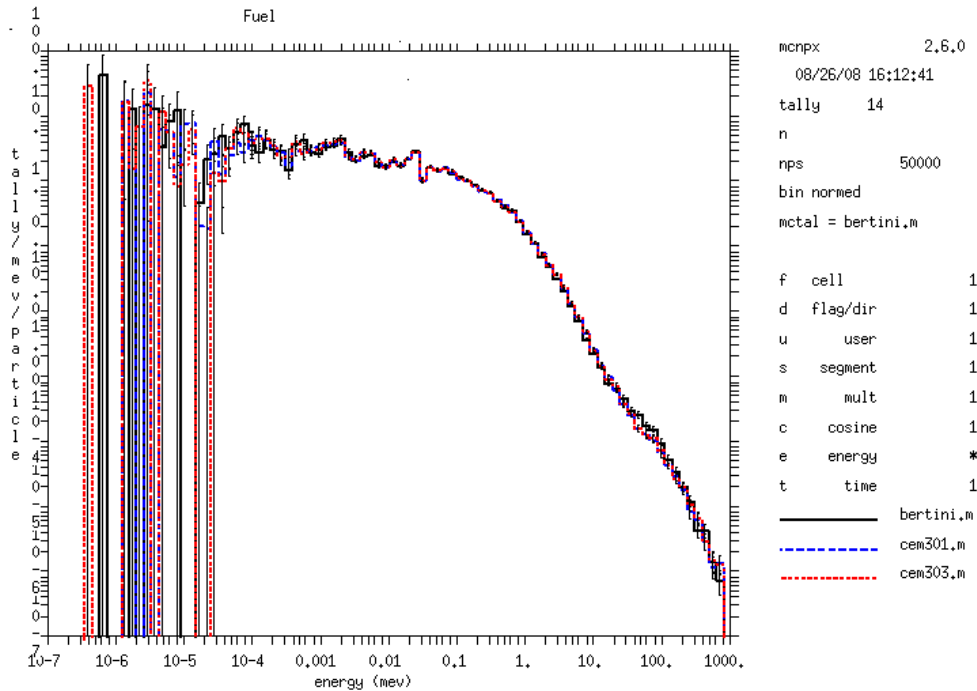


Figure 2. Neutron flux generated by an 800-MeV proton beam in LANSCE fuel sample. (Note that CEM 03.03 is mislabeled as “cem303” in the plot.)

2.3 MCPLLOT Graphics and “HELP” Command Upgrades

MCPLLOT has been upgraded, providing improved appearance of graphics and an enhanced help package. MCPLLOT plots table-range cross-section data, tallies, and other quantities such as convergence criteria and criticality parameters. Among these improvements are the following:

- Significantly improved help package with alphabetized commands and the capability of retrieving short descriptions of each command with its options
- Capitalized fonts and capitalization of code-supplied titles and quantities. Among other things, “mev” is now “MeV”
- Horizontally displayed y-axis labels
- Enable 70-character plot titles
- Increase the limit of orders of magnitude for tally and other data from 17 to 35. This allows time phenomena from nanoseconds to years can be placed on a log plot.
- Allow log10 axes for contour plots

2.3.1 Graphics Upgrades

Among the graphical improvements are the following:

Improved readability and flexibility of axis values: Ordinate values are now displayed horizontally and the default print density of the numbering on each axis was adjusted to eliminate overcrowding. This default density of the axes tick values can be overridden by the user using the *nsteps* parameter on the XLIMS and/or YLIMS command. For logarithmic plots, larger tick marks now signify major decades while smaller marks indicate values between decades. The number of decades that can be displayed on a logarithmic plot has been increased from 17 to 35. This increase allows time phenomena from nanoseconds to years to be placed on a logarithmic plot. Also, use of the “E” format to represent scientific notation decreases the tick label length and consequently enhances readability.

Axes and legend labels: For energy plots, the default units now read “MeV” instead of “mev” and the first letter of each word in default axis labels is now capitalized. Character limits for labels have been increased: up to 23 characters can now be displayed in the legend label and 70-character-long main titles are now permitted for both screen and PDF plots.

Information and legend blocks: The first letter of the left-aligned tally bin type is now capitalized and the *probid* is included in the information block to explicitly indicate the date and time stamp of the runtpe or mtal file instead of the build date of the executable.

Figure 3 illustrates several plot upgrades, including (1) longer titles, (2) legend problem identification (*probid*), (3) alignment of legend tally type labels, (4) increased length of legend curve identifiers, (5) capitalized letters for axes and legend labels, (6) horizontal display of ordinate numbers, and (7) wider major ticks for logarithmic axes.

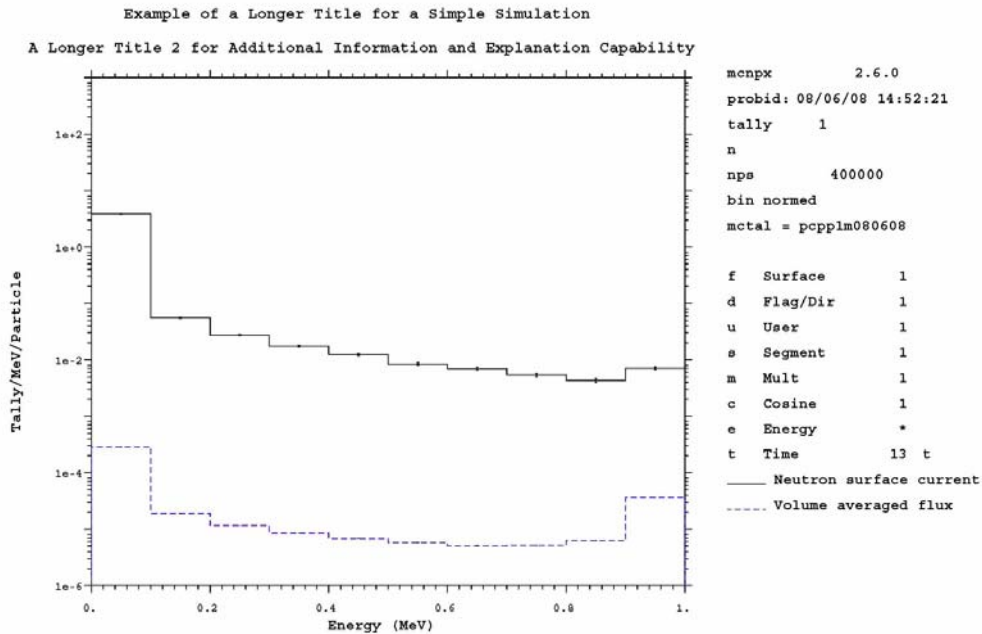


Figure 3. Upgraded MCPLLOT 2-D tally graphics.

Figure 4 illustrates (1) 200-decade logarithmic-axis plot capability, (2) wider ticks at major logarithmic-axis decades, (3) automated refinement of the number of minor ticks, (4) automate refinement of the number density on logarithmic axes.

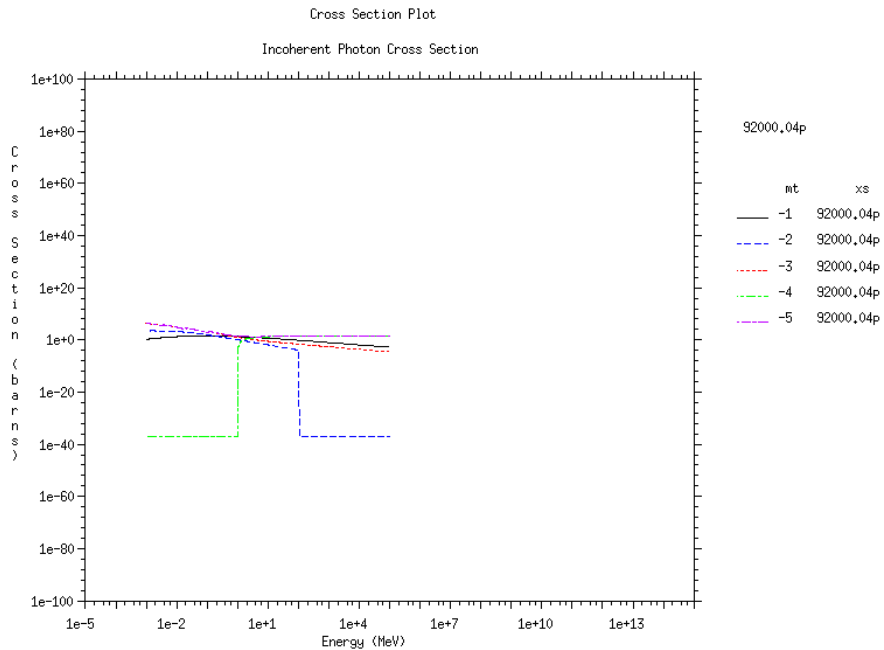


Figure 4. Upgraded MCPLLOT cross-section plot graphics.

Contour plots: Contour plots may now be created using logarithmic axes. In addition, contour-plot legend information now ranges from low to high as one moves up the page.

Figure 5 illustrates (1) the use of logarithmic-axis plotting for contour plots and (2) legend decade identifiers ranging from low to high, moving up the page.

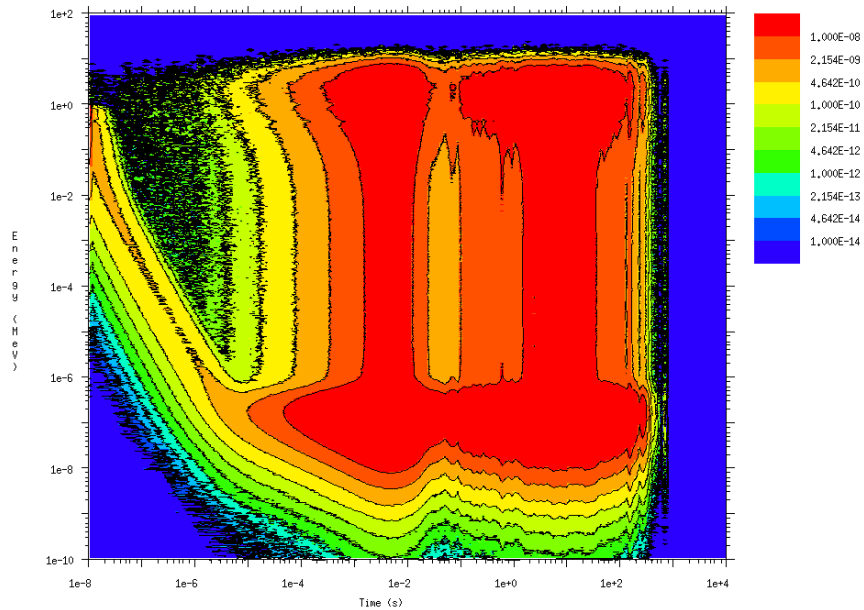


Figure 5. MCNPLOT log-scale contour plot.

2.3.2 “HELP” Command Upgrade

The MCNPX MCNPLOT “HELP” command has been upgraded to improve its functionality. The three-column listing of HELP commands is now alphabetized. In addition, tips appear at the bottom of the three-column list of HELP commands to indicate how to

- 1) invoke a listing of all HELP commands with an explanation of their function and use syntax (“HELP ALL”),
- 2) provide a listing of function and syntax for a single HELP command (“HELP <command>”),
- 3) request an overview of the MCNPLOT capability (“OVERVIEW”), and
- 4) summarize input and execution-line options (“EXECUTE”).

To further aid the user, some commands are now cross-referenced to guide the user toward related commands that may be of interest.

Example 1: After invoking the MCPLLOT utility, if you enter the command “HELP”, you will receive the following response:

```
mcplot>
help
Available commands:
      bar          loglog      subtitle
      below       mt          tally
      contour     noerrbar   term
      coplot      factor      tfc
      dump        options    thick
      end         par         thin
      factor      pause      title
      file        pert       view
      fixed       plinear     wmctal
      free        plot       xlims
      freq        printal    xs
      help        printpts   xtitle
      hist        reset      ylims
      iptal       return     ytitle
      kcode       rmctal     ztitle
      label       runtpe     &
      legend      scales     ?
      linlin      set        3d
      linlog      spline
      loglin      status

Type "help all" for a verbose list of all help commands
"help <command>" to list a specific help command,
"help overview" for an overview of MCPLLOT,
or "help execute" for MCPLLOT input & execution-line
options.
mcplot>
```

Example 2: After invoking the MCPLLOT utility, if you enter the command “HELP legend”, you will receive the following response:

```
mcplot>
help legend
  legend > Syntax: legend [x y]
           Include or omit the legend according to the values of
           optional parameters x and y.
           If no x and no y, put the legend in its
           normal place. (DEFAULT)
           If x=0 and no y, omit the legend.
           If both x and y are defined, for 2-D plots only,
           Put most of the legend in its usual place,
           but put the part that labels the plot lines at
           location x,y.
           Units for x and y are the units used for
           the x and y axes.
           The maximum legend curve label length is
           23 characters.
           See "label".
mcplot>
```

2.4 Embedded Sources

Source distributions now can be embedded within each other to describe accelerator micro-pulses and other sources that vary regularly.

2.4.1 User Interface

The format to specify an embedded source is

```
SDEF   TME = ( D11 < D12 < D13 )
```

or, for distributions of distributions, the following form may be used:

```
SDEF   TME = D41
SI41   S  51 ( D11 < D12 < D13 ) 52
```

In both cases, distributions 11, 12, 13 are all for the same variable, time. Distribution 11 covers a small time range that is repeated as often as needed to exactly fill the larger time range of distribution 12. Similarly, distribution 12 is repeated as often as needed to exactly fill the even larger time range of distribution 13.

Note that the parentheses are optional and that the designator “D” on the SI card with “S” option is also optional. Thus

```
SDEF   TME = ( D11 < D12 < D13 )   and
SDEF   TME =  D11 < D12 < D13     are equivalent.
```

Also,

```
SI41   S  51 ( D11 < D12 < D13 ) 52   ,
SI41   S  51   D11 < D12 < D13   52   ,
SI41   S  51 (  11 <  12 <  13 ) 52   , and
SI41   S  51   11 <  12 <  13   52   are all equivalent.
```

The embedded distributions must start at zero or a fatal error message is issued. For (D11<D12<D13) the lowest value on the SI11 and SI12 cards must be zero. The embedding distribution, D13, can have any range.

The embedded distributions should fit within each other (nearly) exactly. If they do not there is a fatal error message, “embedded distribution *nn* has improper range” and the distribution will spill into the next bin and have a strange normalization for values in its last bin.

Only continuous source distributions such as ERG, TME, X, Y, Z, DIR, RAD and EXT may use embedded distributions.

2.4.2 Examples

Example 1:

```
sdef   tme= d11<d12<d13
si11  0 1 2
sp11  0 1 0
```



```

si12 0 100 150
sp12 0 1 0
si13 -150 900
sp13 0 1
    
```

Figure 6 shows the MCNPLOT plot (MCNPX Z option) of the tally of this source in a vacuum during the time regime ranging from -200 to 1000 shakes. The seven repetitions of distribution 12 (which is described in SI12 as ranging from 0 to 150 shakes) are clearly seen embedded within distribution 13 (which describes the time period from -150 to 900 shakes.)

Figure 7 shows the MCNPLOT plot from -5 to 25 shakes. The 1st 13 of a total 50 repetitions of distribution 11 (which are each two shakes in duration) are seen in the first part of the 2nd repetition of distribution 12.

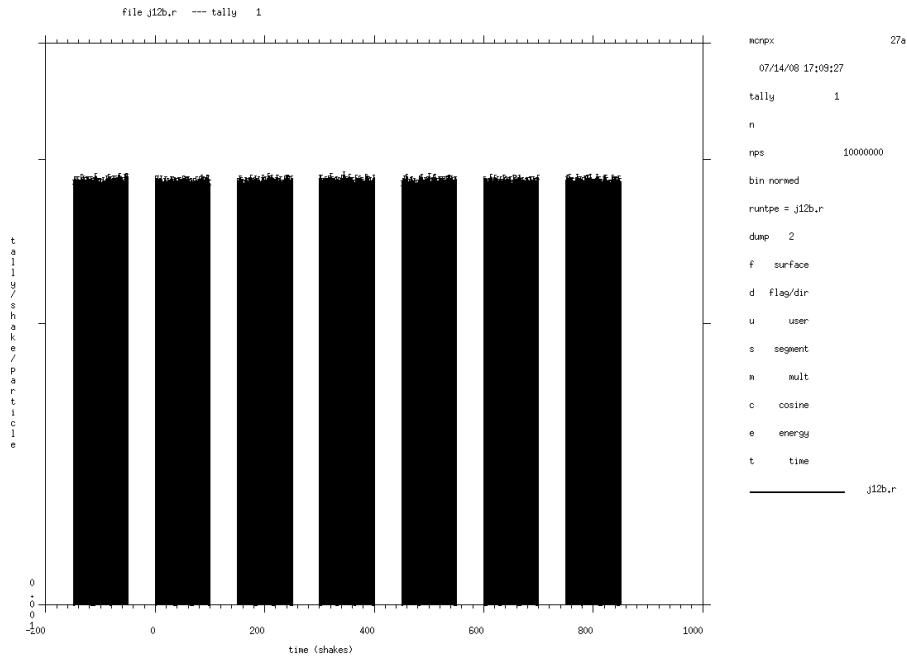


Figure 6. MCNPLOT plot of tally from -200 to 1000 shakes shows embedded pulses defined by distribution 12.

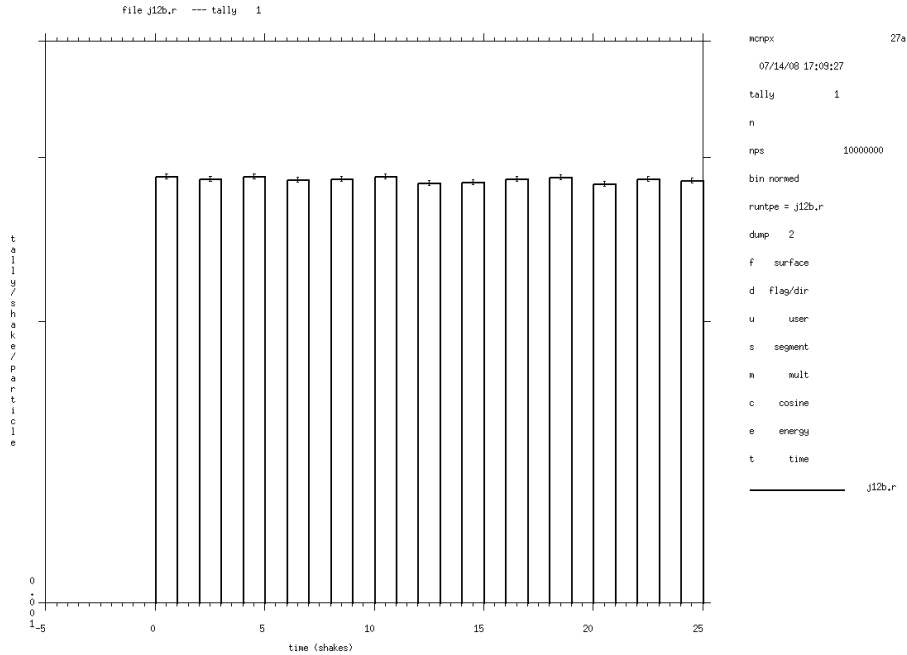


Figure 7. MCNPLOT plot of tally from -5 to 25 shakes, displaying embedded pulses defined by distribution 11.

Example 2 (embedding a distribution of distributions):

```
sdef  erg d21 cel=d11 tme=d41
si41  S 52<51 (D31<32<d33) 61
sp41  .1 .8 .1
si51  A -26 -16
sp51  0 1
si52  H 0 1 2
sp52  0 1 0
si61  A 32 40
sp61  1 0
si31  0 1 2
sp31  0 1 0
si32  0 16
sp32  -41 8 8
si33  -16 32
sp33  0 1
```

This more complex example illustrates how embedded distributions can reside within distributions of distributions (D41), and can use built-in functions (D32 uses a Gaussian centered at t=8 with FWHM=8) and interpolated distributions (D51 and D61 use the SI A option). Distribution D52 is embedded in distribution D51; distribution D31 is embedded in distribution D32, which is embedded in distribution D33. The tally plot is in Figure 8.

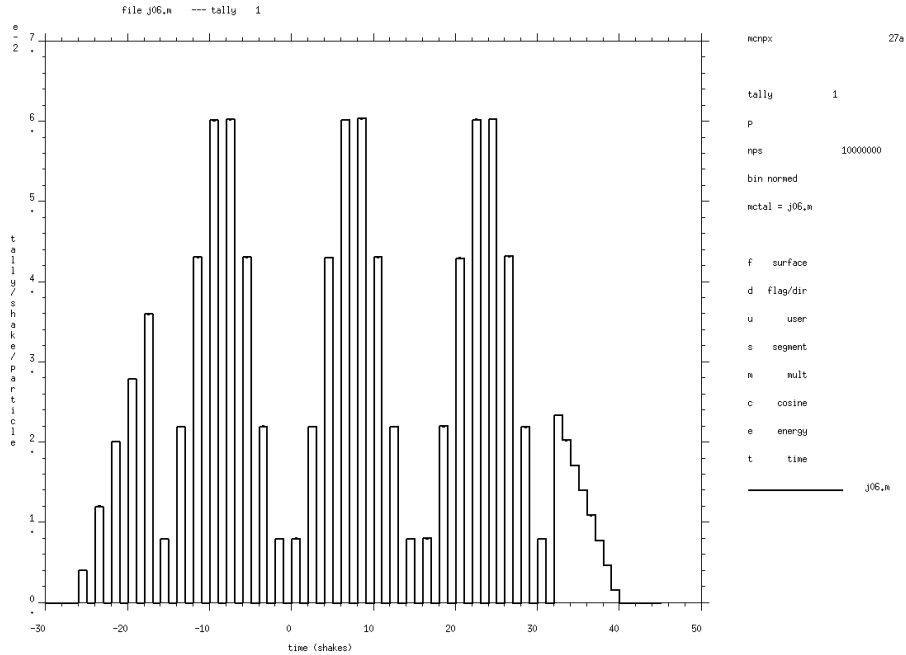


Figure 8. MCNPLOT plot of tally from -30 to 50 shakes.

2.5 Stopping Power Energy Table Control

A new 11th entry on the `PHYS:<pl>` card, where `<pl>` is any charged particle type, now controls the stopping power table energy spacing. This 11th entry is the parameter `efac`. When `efac` is specified, the energy fraction (`efac`) for stopping power tables is the following:

$$E_{n-1} = E_n * efac$$

This applies to charged particles only. The value of `efac` is limited to $0.8 \leq efac \leq 0.99$. A larger `efac` produces more points in the stopping power tables. Default is `efac` = 0.917.

3 MCNPX 2.7.A FEATURE EXTENSIONS AND ENHANCEMENTS

Some MCNPX features have been extended and have included changes or additions to the user interface. Where appropriate, initials of the principal developer(s) are provided in parentheses:¹

- spontaneous photon automatic weight adjustment (MRJ/LSW/GWM)
- burnup enhancements (MLF)
- lattice FILL error trap (JSH)
- LINUX static build (GWM)

¹ John S. Hendricks (JSH, LANL, X-3-MCC); Gregg W. McKinney (GWM, LANL, X-3-MCC), Joseph W. Durkee (JWD, LANL, D-5), Jay S. Elson (JSE, LANL, D-5), Michael L. Fensin (MLF, LANL, N-4), Michael R. James (MRJ, LANL, D-5), Russell C. Johns (RCJ, LANL, D-5), Stepan G. Mashnik (SGM, LANL, X-3-MCC), Denise B. Pelowitz (DBP, LANL, D-5), Laurie S. Waters (LSW, LANL, D-5), Franz X. Gallmeier (FXG, ORNL)

- Compilation without graphics (GWM/MLF)
- Minor enhancements

3.1 Spontaneous Photon Automatic Weight Adjustment

The decay (or spontaneous) gamma source is a feature that allows the user to easily create a gamma spectra source associated with the decay of specified isotopes. With the implementation of the CINDER database and a gamma-line database, MCNPX can now lookup the appropriate activity and decay lines associated with a large number of isotopes. For many other isotopes, a multi-group treatment is available.

Spontaneous material source: The user can build a material that includes unstable isotopes and specify their appropriate fractions. In this case—analogue to the “SF” particle identification feature—the “SP” particle type can be specified on the SDEF card. The decay gammas will then be created based the unstable isotopes found in the material card at the sampled source location. If multiple unstable isotopes are present, they will be sampled based on their relative activities *within* that material.

Example 1:

```
m1 1001 1
    8016 1
    7016 1e-4 $ Unstable isotope N-16
    25054 1e-2 $ Unstable isotope Mn-54

sdef par=sp pos= 0 0 0 $ Location of material 1

phys:p 5j -102
```

When the SF source is specified, the code will create an optional table 44 that summarizes the activities of the materials in the problem.

Spontaneous gamma particle specification: The other way to start decay gammas builds upon the heavy ion feature in MCNPX. The user can specify a Z*1000+A as a particle type on the SDEF card and set the energy of that particle to zero. This will signal the code not to transport the heavy ion but to start its decay gammas. Note that in this case, the ‘#’ (heavy ion) *must* appear on the mode card.

Example 2:

```
mode p #
sdef par=7016 erg=0 pos= 0 0 0
phys:p 5j -102
```

Features and defaults: The decay gammas that are emitted are assumed to arise from the instantaneous activity of a large pool of decaying isotopes. So the time behavior will assume either whatever is specified on the SDEF card or the default of TME=0.

The sampled heavy ion isotopes will appear in Table 110 as the source particle although the created gamma lines do not.

It is also possible to do distributed sources with either of these options, with both options together or in combination with other sources in MCNPX.

As is the case with other source particles, be aware that if an isotope emits multiple gamma lines, the emissions will not necessarily be correlated.

Also note that isotopes with half-lives longer than 1×10^{18} seconds ($\sim 3.17 \times 10^{10}$ years) are treated as stable.

Exponential decay function: A new source probability function has been added that allows users to create a source with an exponential decay shape. The special function is invoked on the associated source probability (*SPn*) card by a “-7” followed by the half-life specified in units of shakes. The default half-life is 1 shake.

Example 3:

```
sdef tme=d1
sp1 -7 2e8 $ 2e8 shakes = 2 seconds
```

SPn W source probability option: A new source probability option has been added. The “*SPn W*” is a distribution specification that can only be applied to particle distributions. This distribution allows the user to specify actual intensities for a mix of particle sources. The intensities will be normalized per standard operation of the source, however the factor used to renormalize the intensities will be applied to the source weight to give the tallies the correct magnitude.

Example 4:

```
sil      n      p      h
sp1 w  3e9  5e9  2e9
```

The source shown here will create three sources and sample them by the relative magnitudes (neutrons 30%, photons 50%, protons 20%). The weight assigned to each particle will be the sum of the non-normalized values, $3e9+5e9+2e9=1e10$.

The second aspect of the “*SPn W*” option is to allow *SP* (spontaneous photon) and *SF* (spontaneous fission) sources to be assigned magnitudes automatically in the code. A negative number corresponding to one of these two source types tells MCNPX to use the cell volume represented by the cell number that follows for the computation of the intensity.

Example 5:

```
sil      sp      sf      n
sp1 w  -10  -15  3e9
```

In the source described here, the spontaneous photon source will look to cell 10 and use the material and volume to calculate the overall activity that will be substituted into the *SP1* distribution. Correspondingly, the *SF* source will look to the material and volume in cell 15 for the intensity of the spontaneous fission source (note that a +*SF* normalizes per

spontaneous fission neutron, a `-SF` normalizes per spontaneous fission). The neutron source is unchanged.

Once the overall activity is computed, the renormalization will be done as before and the value passed into the weight parameter.

This option will also work with the `SB` (source biasing) option where sampling frequencies can be adjusted with the weights to bias the distributions.

If the cell specified in the `SPn W` option is a lattice cell, then the code may not know the correct volume for this cell. A volume can be supplied by a `VOL` card or with a `VOL` keyword on the cell card. If the user does not wish to correct the volume in these ways, a `WGT` keyword can be used with the source as a multiplicative factor.

Example 6:

```
sdef par=d1 wgt=264
sil L sf
spl w -35
```

In this example, the spontaneous fission source is weighted by the activity from cell 35 which has been duplicated 264 times in the geometry. The final source weight will be the activity from cell 35 * 264.

3.2 Burnup Enhancements

Parallelizing the CINDER90 Interface: The burnup capability in previous versions of the code required that MCNPX calculate steady-state reaction rates (when transport cross sections were available) and 63-group fluxes to generate depletion coefficients for the time-dependent CINDER90 burnup calculation. Because MCNPX was already parallelized, the steady-state reaction-rate calculation was also parallelized in MCNPX 2.6.B to execute in multiprocessor mode. However, at the end of each steady-state reaction-rate calculation each slave processor had to rendezvous with the master processor to perform the interface/CINDER90 subroutines in a serial execution.

Several independent serial looping structures within the burnup interface allow the assemblage of depletion coefficients and isotopic information into a format suitable for CINDER90 execution. The computational expense associated with this serial execution increases with the number of burn materials requested in the burnup calculation. For example, burning a single material may only require one minute of interface/CINDER90 execution time; however, burning 150 materials may require 300 minutes.

Because the depletion of each burn material is an independent calculation, the computational cost associated with the interface/CINDER90 execution was remedied by parallelizing the interface/CINDER90 routines for multi-processor execution. The interface/CINDER90 calculation is now distributed across all available processors including the master processor. Each processor executes the interface/CINDER90 routines for a subset of burn materials and then rendezvous at the end of the interface/CINDER90 routines to determine total system burnup as well as atom densities for the subsequent reaction-rate calculation. Because the

interface/CINDER90 routines are independent for each burn material and minimal communication across processors is necessary to execute the interface/CINDER90 subroutines, the computational enhancement scales almost linearly with the number of processors.

Variable Dimensioning for Burnup and Spontaneous Photons: The dimensions of spontaneous photon and burnup arrays was made variable so that these arrays only take up computer memory space when they are used for these features. Certain arrays for the burnup and spontaneous photon capability were automatically dimensioned by the mix variable for any type of calculation. This unnecessary dimensioning wasted memory, thus limiting the size of calculations that could be examined by the code. All the arrays for the burnup capability are now dimensioned by unity if a BURN card is not present; similarly, all arrays for the spontaneous photon capability are dimensioned by unity if the spontaneous-photon capability is not used (MLF).

Unlimited CONC Changes: Additional dynamic dimensioning now permits an unlimited number of user-specified concentration changes (CONC option on BURN card.) Users implementing the MATMOD keyword in MCNPX 2.6.A(0) received strange results when trying to manually modify more than three nuclides per time step. Users may now manually modify as many nuclides as exist in the burn material. Nuclides that make up the burn material are included from the material card, fission product tier, and isotope generator algorithm (MLF)

3.3 Lattice FILL Error Trap

Errors in a fully specified lattice FILL are now captured with a “Bad Trouble” error. This is not a bug fix because it only happens with user errors when setting up geometries. Previously, an erroneous lattice fill would go undetected and usually cause a crash or lost particles, but sometimes cause wrong answers.

Consider the following:

```
FILL  0:2  0:1  0:0   1  1  1   1  1  1  2
```

The fully specified FILL describes a 3 x 2 x 1 lattice, so it should be followed by six universes. Instead there are seven. The new lattice fill error trap now detects either too few or too many universes.

3.4 LINUX Static Build

The MCNPX executables provided for LINUX and built with the Intel compiler have been compiled so that users can use executables without having to install a compiler. To make LINUX Intel static builds simply add

```
--with-LDFLAGS=-static-libcxa
```

to the configure line when compiling.

3.5 Compilation without Graphics

MCNPX may now be built without graphics on Windows (using both Intel 9.1 and 10.1) for MPI parallel versions. The MCNPX C code was revised to omit the use of TIMEVAL and GETTIMEOFDAY() and to use instead the C++ intrinsic function FTIME.

3.6 Minor Enhancements:

- *E > 1 TeV / nucleon Warning*: The warning for source particles with $E > 1$ TeV/nucleon was improved. Suggested by Robert Singleterry (NASA) (MRJ/GWM)
- Removed DBCN(20) which caused tracking with MCNPX 2.5.0 (JSH)
- Removed DBCN(21) which caused tracking with MCNPX 2.5.E (JSH)
- Removed DBCN(25) which turned off the Eric Pitcher / Bob Little $S(\alpha,\beta)$ broadening enhancement in MCNPX2.6.0 (JSH)
- Deleted the screen print of contour levels during geometry plots unless contour levels were set by the user. (JSH)

4 MCNPX CORRECTIONS

4.1 Significant Problem Corrections

The following problems could cause incorrect answers. Fortunately, they occurred only in very special situations and affected only a few MCNPX users.

MCNPX v.2.7.A corrections to MCNPX v.2.6.0

4.1.1 Minor Burnup Reaction Rates Off

Data were loaded incorrectly (off by one energy bin) into the CINDER90 63-group flux bin structure for determining the contribution of a specific track traverse for a given burn material cell. The error only affected burnup/depletion for minor reactions that are treated with the 63-group fluxes – not fission, capture, and other major reactions that are treated continuously or non-burnup problems. The effect is generally so small as to be unnoticed in benchmark calculations. The error was a bug in the binary search algorithm in the burn_trk.F subroutine and has been in all MCNPX versions with burnup. A \$20 award was made to Charles Whitmer (Whitmer Consulting, North Bend, WA) (X-3:08-77). (MLF)

4.1.2 Spontaneous Fission Normalization Error

When there is a spontaneous fission source (PAR=SP) and there is more than one spontaneous fission nuclide in a cell, then the relative weighting of source strengths was done based upon spontaneous fission yield data. However, MCNPX uses yield data in gram units and did not convert to atom units for the relative weighting. This error only affects spontaneous fission when there is more than one spontaneous fission nuclide in the cell. Whereas spontaneous fission is only done with actinides whose mass and atom fractions are similar, the effect is small and generally within the uncertainty of the yield. The error has been in all MCNP and MCNPX 2.6.0 and earlier versions. (MRJ)

4.1.3 DXTRAN Fail with Repeated Structures

Contributions from repeated structure / lattice cells to DXTRAN spheres are incorrect because the lattice level is not saved. This applies to all previous code versions including MCNPX 2.6.0 and MCNP5. There is no problem with only one DXTRAN sphere, but with additional DXTRAN spheres or point detectors, the lattice level index is wrong and particles can get lost, cause a crash, or cause silent wrong answers. A \$2 award was made to Richard Olsher, LANL RP-2 (X-3:JSH-09-007). (GWM)

4.1.4 Heavy Ion Errors

The heavy ion ipt is set to the ZA in bankit for banking heavy ions, but it was not being set back to mipt. This could lead to out-of-bounds array calls and other errors if the banking routine expected to use the ipt value after the bankit call. In bankit, the ipt is now restored to mipt after a heavy ion is banked. (MRJ)

4.1.5 $S(\alpha,\beta)$ File Reading Failure

If $S(\alpha,\beta)$ data files are mixed in with other types of data files (like neutron or photon data tables) and if they are read before the other tables, some cross-section pointers can be incorrect causing wrong answers. The problem affects MCNPX 2.6.0 and all previous versions. In one case, k_{eff} was wrong by a factor of two. Note that none of the data libraries distributed with MCNP/MCNPX mixes $S(\alpha,\beta)$ and other data table types. Therefore, this is only a problem for users who mix data type when constructing their own libraries of nuclear data tables. (GWM)

4.1.6 Burnup Continue Runs

Continue runs of burnup problems that included the MATVOL or MATMOD keywords did not track and were wrong because arrays *bvol* and *comb* were not written to the RUNTPE file. These arrays are now written to the RUNTPE file and continue runs now work with all burnup capabilities. (MLF)

4.2 Minor Problem Corrections

The following problems do not cause wrong answers, but they may cause crashes when encountered. In any case, they occurred only in very special situations and affected only a few MCNPX users.

MCNPX v. 2.6.0 corrections to MCNPX v.2.6.F

4.2.1 INTEL Compiler Workaround

Large problems failed with the INTEL 10 compiler requiring a workaround. A \$2 award was made to Manuel Bardies (INSERM, Nantes, France) (X-3:JSH-2008-050). (GWM)

4.2.2 LAQGSM Corrections

LAQGSM Standalone: The LAQGSM module compiled as a standalone code wrote to the wrong unit number. (GWM)

LAQGSM Infinite Loops: Several instances of infinite loops in LAQGSM have been identified and are now resampled rather than having MCNPX hang. (MRJ)

Coulomb Barrier Physics: Coulomb barrier physics has been added for heavy ions. (MRJ)

Heavy Ion SPABI: Secondary particle production biasing, the SPABI input card option, failed for heavy ions and has been corrected. (MRJ)

4.2.3 Compiler Workaround

The floating point variable TDS was used in a ‘do loop’ in routine ITALLY. (This is a harmless error or compile error, depending upon the system.) An award of €20 was made to Frank Günsing, (CEA/Saclay, France) (X-3:JSH-2008-045). (JSH)

4.2.4 Detector Tallies in Lattice Geometries

If a detector tally contribution is from a cell filled with a lattice, then the mesh type 4 or point detector tallies did not score. A \$2 award was made to Róbert Kákonyi (University of Szeged, Hungary) (X-3:JSH-2008-049). (GWM)

MCNPX v.2.7.A corrections to MCNPX v.2.6.0

4.2.5 Burnup Duplicate ZAIDs

Duplicate ZAIDs on a materials card caused burnup to crash. \$20 to Gregory A. Johnson (Hamilton Sunstrand–Rocketdyne, Canoga Park, CA) (X-3:08-60). (MLF)

4.2.6 Source Particle Types Unrecognized in Dependent Distributions

Specification of different source particles in dependent distributions is unrecognized. For example, the combination

```
SDEF PAR=FPOS=D99
```

and

```
DS99 S D12 D13
```

fails and gives spurious warning messages. MCNPX thinks D12 and D13 are particle types rather than distributions (where the particle types would then be specified). An award of \$20 was made to Gregory Thoreson (LANL D-5) (X-3:08-78). (LSW/JSH)

4.2.7 Incorrect Warning Error for Detector Tallies

The error message “tally not scored beyond last time bin” for detector tallies printed the time at the collision rather than the time at the detector for point/ring detectors. The bug was in all previous MCNP, MCNP5, and MCNPX versions. A \$2 award was made to Bill Sailor (LANL, ISR-4) (X-3:08-76). (GWM)

4.2.8 Choice of High-Energy Mode

Neutrons, protons, or deuterium colliding with neutrons protons, or deuterium at high energies should go to FLUKA, not LAQGSM. A \$2 award was made to Sukesh Aghara (Prairie View A&M University, Texas) (X-3:JSH-2008-048). (MRJ)

4.2.9 Lost Particles Cause Crash

See Section 4.1.3 – DXTRAN fails with repeated structures.

5 FUTURE WORK

The following projects are at least partially funded and are actively being developed:

- Additional CINDER90 capabilities:
 - Burnup / depletion enhancements;
 - Fixed source depletion;
 - Automatic production of background activation; and
 - Automatic production of radioactive sources;
- Correlated secondary particle production;
- Improvements in photonuclear cross sections and yields.

The following projects are on our wish list. Some have been partially developed but await further funding.

- Improved convergence of eigenfunctions in criticality problems;
- Integration of HTAPE tallies directly into MCNPX, including continue runs;
- Electric and magnetic field tracking;
- Detectors and DXTRAN for all neutral particles at all energy ranges with anisotropic scattering (currently approximated as isotropic for models);
- CAD link;
- Secondary particle angle biasing for isotropic distributions;
- Neutral particle perturbation techniques extended to physics model region;
- Plotting of physics model total and absorption cross sections; and
- Forced collisions for neutral particles extended to physics models.