LA-UR-13-24019
Approved for public release; distribution is unlimited.

Title: Verification of MCNP5-1.60 and MCNP6.1 for Criticality Safety Applications

Author(s): Brown, Forrest B.
Kiedrowski, Brian C.
Bull, Jeffrey S.

Intended for: ANS NCSD 2013 - Criticality Safety, 2013-09-29/2013-10-03 (Wilmington, North Carolina, United States)
MCNP documentation
Web

Issued: 2013-06-03

*ANS NCSD 2013 - Criticality Safety in the Modern Era: Raising the Bar*
Wilmington, NC, September 29 –October 3, 2013, on CD-ROM, American Nuclear Society, LaGrange Park, IL (2013)

# Verification of MCNP5-1.60 and MCNP6.1 for Criticality Safety Applications

**Forrest B. Brown, Brian C. Kiedrowski, Jeffrey S. Bull**
Monte Carlo Codes Group (XCP-3), Los Alamos National Laboratory
PO Box 1663, MS A143, Los Alamos NM 87545
fbrown@lanl.gov; bckiedro@lanl.gov; jsbull@lanl.gov

## ABSTRACT

To verify that both MCNP5-1.60 and MCNP6.1 are performing correctly for criticality safety applications, several suites of verification/validation benchmark problems were run in early 2013. Results from these benchmark suites were compared with results from previously verified versions of MCNP5. Testing results for 354 problems on Mac OS X, Linux, and Windows systems indicate that: (1) Both MCNP5-1.60 and MCNP6.1 perform correctly for criticality safety applications. (2) Using the latest compilers, small differences were noted for a few cases compared to using older compilers, but these are strictly due to computer roundoff and are not a concern for verification. (3) MCNP5-1.60 and MCNP6.1 yield the same results on different computer platforms – Mac OS X, Linux, and Windows – for criticality safety applications. (4) MCNP5-1.60 and MCNP6.1 yield the same results using OpenMP threading and/or MPI message-passing parallelism. Criticality safety analysts should consider testing MCNP6.1 on their particular problems and validation suites, to prepare for the migration from MCNP5 to MCNP6. It is expected that this migration should be accomplished within the next 1-3 years.

*Key Words*: MCNP, Monte Carlo, k-effective, verification, validation

## 1 INTRODUCTION

To verify that both MCNP5-1.60 and MCNP6.1 are performing correctly for criticality safety applications, several suites of verification/validation benchmark problems were run in early 2013. Results from these benchmark suites were compared with results from previously verified versions of MCNP5. The goals of this verification testing were:

- Verify that MCNP5-1.60 continues to work correctly for nuclear criticality safety applications, producing the same results as for the previous verifications performed in 2010 and 2012.
- Determine the sensitivity to computer roundoff using different Fortran compilers for building MCNP5 and MCNP6, to support using current versions of the compilers.
- Verify that MCNP6.1 works correctly for nuclear criticality safety applications, producing the same results as for MCNP5-1.60. This provides support for eventual migration of users and applications to MCNP6.

The current production version of MCNP5 included in the RSICC release package is MCNP5-1.60 [1-7]. This version was first distributed by RSICC in October 2010. While there were subsequent RSICC distributions of the MCNP package in July 2011, February 2012, and January 2013, no changes were made to MCNP5-1.60. The RSICC release package in February

2013 included both MCNP5-1.60 and the current beta version of MCNP6, MCNP6-Beta-3. MCNP5-1.60 will also be included in the upcoming production release of MCNP6.1.

MCNP6 [6-8] is the merger of MCNP5 and MCNPX capabilities. MCNP6 includes all of the features for criticality safety calculations that are available in MCNP5-1.60, and many new features largely unrelated to nuclear criticality safety calculations. The production release of MCNP6.1 is targeted for mid-2013. The RSICC release package will include MCNP6.1, MCNP5-1.60, MCNPX-2.70, the new ENDF/B-VII.1 data libraries, and an updated MCNP Reference Collection.

The benchmark suites used for the MCNP5 and MCNP6 verification are standard criticality suites in the MCNP code repository:

- **VERIFICATION_KEFF** – A suite of 75 criticality problems for which exact analytical results are available [9]. A representative set of 10 problems was chosen from this suite. While the problems use 1-group or few-group energy treatments and simple 1-D geometry, they verify that MCNP reproduces the exact analytical results. That is, they serve to verify the fundamental power iteration scheme used in MCNP that underlies all criticality calculations.
- **VALIDATION_CRITICALITY** - The "Criticality Validation Suite" [10-12] consisting of 31 problems from the *International Handbook of Evaluated Criticality Benchmark Experiments* [10], using the ENDF/B-VI.6, ENDF/B-VII.0, and ENDF/B-VII.1 nuclear data libraries,
- **VALIDATION_CRIT_EXPANDED** - The "Expanded Criticality Validation Suite" [14,15] consisting of 119 problems from the *International Handbook of Evaluated Criticality Benchmark Experiments*, using the ENDF/B-VII.0 and ENDF/B-VII.1 nuclear data libraries,
- **CRIT_LANL_SBCS** – A suite of 194 ICSBEP criticality benchmark problems from the LANL SB-CS Group used to verify and validate MCNP for their work, using ENDF/B-VI nuclear data libraries.

For the **VALIDATION_CRIT_EXPANDED** Suite, all problems were also run with both ENDF/B-VII.0 [16-18] and ENDF/B-VII.1 [19-25] nuclear data libraries. The ENDF/B-VII.1 libraries were recently released and will be the default nuclear data for MCNP5 and MCNP6 in the upcoming RSICC production release of MCNP6.1.

An important part of the recent testing was a comparison of results obtained from MCNP5-1.60 and MCNP6.1 after they were recompiled using different versions of the Intel Fortran compiler. It should be noted that Fortran compilers are complex software programs, and all such programs have bugs. Testing MCNP using different versions of the Fortran compiler helps to verify that both MCNP and the Fortran compilers are performing correctly for criticality safety applications. It is also important to perform the MCNP and Fortran compiler testing on different computer operating systems, e.g., Windows, Linux, and Mac OS X, since codes and compilers sometimes perform differently on different systems.

In Reference [6], it was demonstrated that results from MCNP5 and MCNP6 compiled with different versions of the Intel Fortran compilers agree exactly for nearly all problems, and differ but agree within statistics for a few problems. While there will always be some roundoff differences due to the noncommutative and nonassociative nature of computer arithmetic, and the rearrangement of the order of operations by optimizing compilers, Reference [6] demonstrated that the roundoff differences are entirely acceptable, not errors, and recommended that all future MCNP development be carried out with the Intel 12 (current) Fortran compiler.

All of the testing performed recently was done in a parallel mode, using OpenMP threading with 8-16 cpu-cores.

For all systems, we have used the "*-O1*" optimization level. Past testing typically showed only small gains in performance with higher optimization levels, at the expense of tremendous complications in verification due to small roundoff differences. We discourage users from invoking higher optimization levels, unless they are willing to also perform the necessary additional verification of code correctness.

In general, we try to choose options for different Fortran compilers and computer platforms that are as consistent as possible for building MCNP5. Nevertheless, computer roundoff differences will occur with different compilers/hardware. Roundoff differences are not considered errors. Careful examination of these differences is necessary in the verification process to ensure that these differences are due solely to roundoff, and not to errors in coding or compilers.

**2**

## 3 VERIFICATION RESULTS FOR THE VERIFICATION_KEFF PROBLEMS USING MCNP6.1

**Table I** shows the $K_{eff}$ results for 10 benchmark problems from the **VERIFICATION_KEFF** suite run using MCNP6.1 compiled with the Intel-12 Fortran compiler. The problems were run on a Mac Pro computer using a 64-bit executable, 2 quad-core Xeon processors, OS X 10.6.8, and 8 threads. These analytic problems use 1-group cross-sections. The MCNP6.1 results are compared with the exact analytic results for $K_{eff}$. No significant differences are observed in Table I.

**Table I. MCNP6.1 Results for Analytic Keff Benchmarks**

| Case | Name | Analytic Exact keff | MCNP_Results keff | MCNP_Results std |
|---|---|---|---|---|
| prob11 | Ua-1-0-IN | 2.25000 | 2.25000 | 0.00000 |
| prob14 | Ua-1-0-SP | 1.00000 | 1.00006 | 0.00010 |
| prob18 | Uc-H2O(2)-1-0-SP | 1.00000 | 1.00005 | 0.00011 |
| prob23 | UD2O-1-0-CY | 1.00000 | 1.00000 | 0.00006 |
| prob32 | PUa-1-1-SL | 1.00000 | 0.99995 | 0.00011 |
| prob41 | UD2Ob-1-1-SP | 1.00000 | 1.00003 | 0.00007 |
| prob44 | PU-2-0-IN | 2.68377 | 2.68377 | 0.00003 |
| prob54 | URRa-2-0-SL | 1.00000 | 1.00007 | 0.00013 |
| prob63 | URRd-H2Ob(1)-2-0-ISLC | 1.00000 | 0.99993 | 0.00006 |
| prob75 | URR-6-0-IN | 1.60000 | 1.59999 | 0.00001 |

## 4 VERIFICATION RESULTS FOR THE VALIDATION_CRITICALITY SUITE – MCNP5-1.60 VS MCNP6.1 USING ENDF/B-VII.0 NUCLEAR DATA

**Table II** shows the $K_{eff}$ results for 31 benchmark problems for MCNP5-1.60 compiled with the Intel-10 and Intel-12 Fortran compilers, and MCNP6.1 compiled with the same Intel-12 compiler. The Intel-10 compiler generates only 32-bit executables; the Intel-12 compiler generates 64-bit executables. The problems were run on a Mac Pro computer using 2 quad-core Xeon processors, OS X 10.6.8, and 8 threads.

To simplify the comparisons, **Table II** shows the MCNP5-1.60 Intel-12 results and differences that arise for MCNP5-1.60 Intel-10. Cases that show differences are highlighted in green in both tables.

For the 4 MCNP5-1.60 cases that show differences between the Intel-10 and Intel-12 versions, the differences are within statistics and indicate computer roundoff (most likely from reordering of arithmetic due to compiler optimization), not errors in either MCNP or the Intel compilers. These differences are exactly the same as those seen in 2012 for the previous verification.

Using the Intel-12 compiler, results for MCNP5-1.60 and MCNP6.1 match exactly for all 31 cases. No differences in results were seen, when the same Intel Fortran compiler was used for each code. MCNP6.1 compiled with the Intel-12 compiler in 64-bit addressing mode is roughly 30% slower than MCNP5-1.60.

The **VALIDATION_CRITICALITY** Suite was also run with MCNP6.1 on a **Windows** computer system. MCNP6.1 was compiled using the Intel 12.1.5 Fortran compiler (with 64-bit addressing) on a system running Windows 7 Professional (Service Pack 1). All of the results for this suite of problems exactly matched the results obtained on Mac OS X and Linux systems.

**Table II. MCNP5 & MCNP6 for VALIDATION_CRITICALITY Suite, with Different Fortran Compilers – Diffs, Mac**

```
mcnp5_10_70  = mcnp5-1.60   + Intel 10 Fortran90 + endf/b-vii.0
mcnp5_12_70  = mcnp5-1.60   + Intel 12 Fortran90 + endf/b-vii.0
mcnp6_12_70  = mcnp6.1      + Intel 12 Fortran90 + endf/b-vii.0

Differences are relative to reference case:    mcnp5_12_70
*'s indicate differences > 1, 2, or 3 std
```

| | mcnp5_10_70 | | mcnp5_12_70 | | mcnp6_12_70 | |
|---|---|---|---|---|---|---|
| | **deltak** | **std** | **keff** | **std** | **deltak** | **std** |
| **U233 Benchmarks** | | | | | | |
| JEZ233 | 0.0000 | ( 8) | 0.9989 | ( 5) | 0.0000 | ( 8) |
| FLAT23 | 0.0000 | ( 9) | 0.9990 | ( 7) | 0.0000 | ( 9) |
| UMF5C2 | 0.0000 | ( 8) | 0.9931 | ( 5) | 0.0000 | ( 8) |
| FLSTF1 | 0.0000 | (15) | 0.9830 | (11) | 0.0000 | (15) |
| SB25 | 0.0000 | (14) | 1.0053 | (10) | 0.0000 | (14) |
| ORNL11 | 0.0000 | ( 5) | 1.0018 | ( 4) | 0.0000 | ( 5) |
| **HEU Benchmarks** | | | | | | |
| GODIVA | 0.0000 | ( 8) | 0.9995 | ( 5) | 0.0000 | ( 8) |
| TT2C11 | **0.0010** | **(10)** | 1.0008 | ( 7) | 0.0000 | ( 9) |
| FLAT25 | 0.0000 | ( 9) | 1.0034 | ( 7) | 0.0000 | ( 9) |
| GODIVR | 0.0000 | ( 9) | 0.9990 | ( 7) | 0.0000 | ( 9) |
| UH3C6 | 0.0000 | (11) | 0.9950 | ( 8) | 0.0000 | (11) |
| ZEUS2 | **0.0002** | **( 9)** | 0.9972 | ( 7) | 0.0000 | ( 9) |
| SB5RN3 | 0.0000 | (18) | 0.9985 | (13) | 0.0000 | (18) |
| ORNL10 | 0.0000 | ( 5) | 0.9993 | ( 4) | 0.0000 | ( 5) |
| **IEU Benchmarks** | | | | | | |
| IMF03 | 0.0000 | ( 8) | 1.0029 | ( 5) | 0.0000 | ( 8) |
| BIGTEN | 0.0000 | ( 7) | 0.9945 | ( 5) | 0.0000 | ( 7) |
| IMF04 | 0.0000 | ( 8) | 1.0067 | ( 5) | 0.0000 | ( 8) |
| ZEBR8H | **-0.0001** | **( 7)** | 1.0196 | ( 5) | 0.0000 | ( 7) |
| ICT2C3 | 0.0000 | ( 9) | 1.0037 | ( 7) | 0.0000 | ( 9) |
| STACY36 | 0.0000 | ( 8) | 0.9994 | ( 5) | 0.0000 | ( 8) |
| **LEU Benchmarks** | | | | | | |
| BAWXI2 | 0.0000 | ( 9) | 1.0013 | ( 7) | 0.0000 | ( 9) |
| LST2C2 | 0.0000 | ( 8) | 0.9940 | ( 5) | 0.0000 | ( 8) |
| **Pu Benchmarks** | | | | | | |
| JEZPU | 0.0000 | ( 8) | 1.0002 | ( 5) | 0.0000 | ( 8) |
| JEZ240 | 0.0000 | ( 8) | 1.0002 | ( 5) | 0.0000 | ( 8) |
| PUBTNS | 0.0000 | ( 8) | 0.9996 | ( 5) | 0.0000 | ( 8) |
| FLATPU | 0.0000 | ( 9) | 1.0005 | ( 7) | 0.0000 | ( 9) |
| THOR | 0.0000 | ( 9) | 0.9980 | ( 7) | 0.0000 | ( 9) |
| PUSH2O | 0.0000 | ( 9) | 1.0012 | ( 7) | 0.0000 | ( 9) |
| HISHPG | **0.0004** | **( 7)** | 1.0118 | ( 5) | 0.0000 | ( 8) |
| PNL2 | 0.0000 | (12) | 1.0046 | ( 9) | 0.0000 | (12) |
| PNL33 | 0.0000 | ( 9) | 1.0065 | ( 7) | 0.0000 | ( 9) |

# 5 VERIFICATION RESULTS FOR THE VALIDATION_CRIT_EXPANDED SUITE – MAC, LINUX

## 5.1 MCNP5 & MCNP6 – Fortran Compiler Checking, Mac

This testing involved shortened versions of the 119 problems in the Expanded Criticality Validation Suite (using "**`kcode 1000 1.0 10 50`**", rather than "**`kcode 10000 1.0 100 600`**"). The purpose was simply to look for any apparent differences in using the Intel-10 and Intel-12 Fortran compilers. Any absolute results should be discounted, since the problems were just run mechanically without regard to proper convergence.

**Tables 3a and 3b in Reference [26]** show the full set of $K_{eff}$ results from MCNP5-1.60 using the Intel-10 and Intel-12 compilers, and the $K_{eff}$ differences for MCNP6.1 for the shortened version of the Expanded Criticality Validation Suite. The problems were run on a Mac Pro computer using a 32-bit executable from Intel-10, a 64-bit executable from Intel-12, 2 quad-core Xeon processors, OS X 10.6.8, and 8 threads. One of the 119 cases showed minor roundoff differences between MCNP5-1.60 compiled with Intel-10 vs Intel-12. All of the Intel-12 MCNP5-1.60 and MCNP6.1 results agreed exactly in these shortened tests. The Intel-10 vs Intel-12 differences for MCNP5-1.60 are judged to be insignificant, and simply the normal roundoff differences between the two codes that are expected when running very many calculations.

## 5.2 MCNP5 & MCNP6 – Using ENDF/B-VII.0 Nuclear Data, Linux

**Tables 4a and 4b in Reference [26]** show the full set of $K_{eff}$ results from MCNP5-1.60 and the $K_{eff}$ differences for MCNP6.1 for the 119 problems in the Expanded Criticality Validation Suite (run in the standard way; not shortened). Both sets of calculations were run on a Linux cluster using 16 OpenMP threads and the same Intel-12 compiler with 64-bit executables. Four of the 119 cases showed minor roundoff differences between MCNP5 and MCNP6 results. Three of the cases showed roundoff differences less than 1σ, and the other case showed roundoff of just over 1σ. These differences are judged to be insignificant, and simply the normal roundoff differences between the two codes that are expected when running very many calculations.

## 5.3 MCNP6 – Using ENDF/B-VII.0 and ENDF/B-VII.1 Nuclear Data, Linux

**Tables 5a and 5b in Reference [26]** show the full set of $K_{eff}$ results for MCNP6.1 using ENDF/B-VII.0 and ENDF/B-VII.1 nuclear data for the 119 problems in the Expanded Criticality Validation Suite (run in the standard way; not shortened). Both sets of calculations were run on a Linux cluster using 16 OpenMP threads and the same Intel-12 compiler and a 64-bit executable.

Overall ENDF/B-VII.1 performs (on average) slightly better than ENDF/B-VII.0. The new dataset kept the ENDF/B-VII.0 evaluations for the major actinides, with the exception of the inelastic scattering cross section in $^{233}$U and the delayed neutron decay constant data for all major actinides, and focused on minor actinides, structural materials, and light elements. Most importantly, there is no particular set of cases where ENDF/B-VII.1 performs worse than ENDF/B-VII.0, so most users should be able to switch data versions. References [20] and [24] provide extensive results from testing the ENDF/B-VII.1 nuclear data libraries on a wide range of problems.

## 6 VERIFICATION RESULTS FOR THE LANL SBCS CRITICALITY VALIDATION SUITE

**Tables 6a – 6d in Reference [26]** show the full set of $K_{eff}$ results from 194 ICSBEP problems in the LANL SBCS Group criticality validation suite, using ENDF/B-VI cross-sections and a very old version of MCNP5, MCNP5-1.25 from 2003 compiled with the Intel-9 Fortran compiler. The tables also show the $K_{eff}$ differences (relative to the MCNP5-1.60 results) for older SBCS MCNP5-1.25 results and for results from MCNP6.1 compiled with the Intel-12 compiler.

The old MCNP5 results differ from the MCNP5-1.60 and MCNP6.1 results for 52 cases, with 42 of those cases within statistics and 10 with differences between 1σ and 2σ. The other 142 cases all matched within the precision shown. This agreement is excellent, considering that 10 years of MCNP5 development and bug-fixes, and 3 generations of Fortran compilers, separate the old MCNP5 results from the new ones. All of the 194 cases show differences of less than 2σ. (It should also be noted that 3 of the 10 problems showing differences from old results actually had problem input errors that were corrected for the MCNP5-1.60 and MCNP6.1 calculations.)

Comparing MCNP5-1.60 and MCNP6.1 results for the 194 cases (i.e., inspecting Tables 6a-6d in [26] for differences in the differences), 187 cases match, 4 show differences less than 1σ, and 3 show differences between 1σ and 2σ. These differences are judged to be insignificant, and simply the normal roundoff differences between the two codes that are expected when running very many calculations.

## 7 SUMMARY AND CONCLUSIONS

**Table III** provides a summary of the verification results for the recent testing of MCNP5-1.60 and MCNP6.1 for criticality safety applications. The general conclusions from this testing are:

- Both MCNP5-1.60 and MCNP6.1 perform correctly for criticality safety applications.
- While small differences were noted for a few cases, these are strictly due to computer roundoff and are not a concern for verification/validation.
- MCNP5-1.60 and MCNP6.1 yield the same results on different computer platforms – Mac OS X, Linux, and Windows – for criticality safety applications.
- MCNP5-1.60 and MCNP6.1 yield the same results using OpenMP threading and/or MPI message-passing parallelism.
- Using the Intel-12 compiler and 64-bit addressing produces roughly a 20% speedup in the MCNP executables compared to using older compilers.
- MCNP6.1 runs roughly 30% slower than MCNP5-1.60. Causes for the MCNP6.1 performance reduction are under investigation.

**Table III. Summary of Verification Results**

**VERIFICATION_KEFF Suite** – 10 analytical problems with exact $K_{eff}$ results

- MCNP6.1, Intel-12 F90 — **All results match**

**VALIDATION_CRITICALITY Suite** – 31 ICSBEP Cases, ENDF/B-VII.0 Data

- MCNP5-1.60 vs MCNP6.1
  - MCNP5 Intel-10 vs Intel-12: — **4 diffs, within statistics**
  - MCNP5 & MCNP6, Intel-12: — **All results match**

**VALIDATION_CRIT_EXPANDED Suite** – 119 ICSBEP Cases, ENDF/B-VII.0 Data

- MCNP5-1.60 vs MCNP6.1, SHORTENED PROBLEMS
  - MCNP5 Intel-10 vs Intel-12: — **1 diff, within statistics**
  - MCNP5 & MCNP6, Intel-12: — **All results match**
- MCNP5-1.60 vs MCNP6.1
  - MCNP5 & MCNP6, Intel-12: — **4 diffs, within statistics**

**CRIT_LANL_SBCS Suite** – 194 ICSBEP Cases, ENDF/B-VI data

- MCNP5-1.60 vs MCNP6.1, Intel-10.1 F90 — **187 results match**; **4 diffs, within 1σ statistics**; **3 diffs, within 2σ statistics**
- MCNP5-1.60 (2010, Intel-12) vs MCNP5-1.25 (~2003, Intel-9) — **142 results match**; **42 diffs, within 1σ statistics**; **10 diffs, within 2σ statistics**

As a result of this testing, it is recommended that all future development for MCNP be accomplished using the latest Fortran compiler, Intel-12, rather than older versions of the compiler. Using the Intel-12 Fortran compiler with 64-bit addressing permits the solution of very large problems that could not be run with older compilers and 32-bit addressing (where array sizes were limited to less than 2 GB), and also provides a speedup of roughly 30% in code execution.

Criticality safety analysts should consider testing MCNP6.1 on their particular problems and validation suites, to prepare for the migration from MCNP5 to MCNP6. It is expected that this migration should be accomplished within the next 1-3 years. Currently, no further development of MCNP5 is planned; all future MCNP improvements, bug fixes, and new capabilities are targeted only to MCNP6.

## 8 ACKNOWLEDGMENTS

This work was supported by the US DOE/NNSA Nuclear Criticality Safety Program.

## 9 REFERENCES

1. X-5 Monte Carlo Team, "MCNP – A General N-Particle Transport Code, Version 5 – Volume I: Overview and Theory", LA-UR-03-1987, Los Alamos National Laboratory (April, 2003).
2. Forrest Brown, Brian Kiedrowski, Jeffrey Bull, "MCNP5-1.60 Release Notes", LA-UR-10-06235 (2010).
3. Brian Kiedrowski, Forrest Brown, Jeffrey Bull, "MCNP5-1.60 Feature Enhancements & Manual Clarifications", LA-UR-10-06217 (2010).
4. Forrest Brown, Brian Kiedrowski, Jeffrey Bull, Matthew Gonzales, Nathan Gibson, "Verification of MCNP5-1.60", LA-UR-10-05611 (2010).
5. F.B. Brown, B.C. Kiedrowski, J.S. Bull, M.A. Gonzales, N.A. Gibson, "MCNP5-1.60 Release & Verification", Trans Am Nuc Soc 104, June 2011, LA-UR-11-00230 (2011).
6. B.C. Kiedrowski, F.B. Brown, J.S. Bull, "Verification of MCNP5-1.60 and MCNP6-Beta2 for Criticality Safety Applications", Trans. ANS 107 [also LA-UR-12-21729, presentation LA-UR-12-25950] (2012).
7. F.B. Brown, B.C. Kiedrowski, J.S. Bull, "Verification of MCNP5-1.60 and MCNP6-Beta-2 for Criticality Safety Applications", LA-UR-12-21041 (2012).
8. MCNP6 Development Team, "Initial MCNP6 Release Overview MCNP6 Beta 2", LA-UR-11-07082 (2012).
9. Sood, R.A. Forster, D.K. Parsons, "Analytic Benchmark Test Set for Criticality Code Verification", LA-13511 and LA-UR-01-3082 (2001)
10. Russell D. Mosteller, "Validation Suites for MCNP™," Proceedings of the American Nuclear Society Radiation Protection and Shielding Division 12th Biennial Topical Meeting, Santa Fe, New Mexico (April 2002). (LA-UR-02-0878)
11. Russell D. Mosteller, "An Assessment of ENDF/B-VI Releases Using the MCNP Criticality Validation Suite," LA-UR-03-7072 (2003)
12. Russell Mosteller, "ENDF/B-VII.0,ENDF/B-VI,JEFF-3.1,and JENDL-3.3 Results for the MCNP Criticality Validation Suite and Other Criticality Benchmarks", ANS PHYSOR-2008, Interlaken, Switzerland [also LA_UR-07-6284] (2008).
13. International Handbook of Evaluated Criticality Safety Benchmark Experiments, NEA/NSC/DOC(95)03, OECD Nuclear Energy Agency (2007).
14. R.D. Mosteller, "An Expanded Criticality Validation Suite for MCNP", LA-UR-10-06230 Rev3 (2010).

15. Russell D. Mosteller, Forrest B. Brown, Brian C. Kiedrowski, "An Expanded Criticality Validation Suite for MCNP", International Conference on Nuclear Criticality, Edinburgh, Scotland, 19-22 September 2011, paper LA-UR-11-04170, presentation LA-UR-11-05076 (2011).
16. M.B. Chadwick, et al., "ENDF/B-VII.0: Next Generation Evaluated Nuclear Data Library for Nuclear Science and Technology", Nuclear Data Sheets, Vol. 107, Number 12, Dec 2006, UCRL-JRNL-225066 (2006)
17. H. Trellue, R. Little, M.B. Lee, "New ACE-Formatted Neutron and Proton Libraries Based on ENDF/B-VII.0", LA-UR-08-1999 (2008).
18. H. Trellue, R. Little, "Release of New MCNP S(α,β) Library ENDF70SAB Based on ENDF/B-VII.0", LA-UR-08-3628 (2008).
19. M.B. Chadwick, et al., "ENDF/B-VII.1 Nuclear Data for Science & Technology: Cross Sectiond, Covariances, Fission Product Yields and Decay Data", Nuclear Data Sheets, Vol 112, No. 12, 2887-2996 (2011).
20. A.C. Kahler, et al., ENDF/B-VII.1 Neutron Cross Section Data Testing with Critical Assembly Benchmarks and Reactor Experiments", Nuclear Data Sheets, Vol 112, No. 12, 2997-3036 [also LA-UR-11-11271 ](2011).
21. J.L. Conlin, et al., "Listing of Available ACE Data Tables", LA-UR-13-21822 (2013)
22. J.L. Conlin, et al., "Continuous Energy Neutron Cross Section Data Tables Based upon ENDF/B-VII.1", LA-UR-13-20137 (2013).
23. J.L. Conlin, et al., "Release of ENDF/B-VII.1-based Continuous Energy Neutron Cross Section Data Tables for MCNP", 2013 ANS Annual Meeting, Atlanta, June 2013, LA-UR-13-20240 (2013).
24. A.C. Kahler, R. E. MacFarlane, R. D. Mosteller, B. C. Kiedrowski, M. B. Chadwick, "LANL Data Testing Support for ENDF/B-VII.1", presentation at US DOE Nuclear Criticality Safety Program Technical Seminar, Knoxville, TN, March 13-14, LA-UR-12-20002 (2012).
25. M. B. Chadwick, P. Talou, T. Kawano, G. Hale, J. Lestone, M. MacInnes, D. K. Parsons, J. L. Conlin, A. C. Kahler, "LANL Data Evaluation Support for ENDF/B-VII.1", presentation at US DOE Nuclear Criticality Safety Program Technical Seminar, Knoxville, TN, March 13-14, LA-UR-12-20003 (1012).
26. Forrest B. Brown, Brian C. Kiedrowski, Jeffrey S. Bull, "Verification of MCNP5-1.60 and MCNP6.1 for Criticality Safety Applications", LA-UR-13-22196 (2013).