LA-UR-17-23822

Approved for public release; distribution is unlimited.

Title:	Verification of MCNP6.2 for Nuclear Criticality Safety Applications
Author(s):	Brown, Forrest B. Rising, Michael Evan Alwin, Jennifer Louise
Intended for:	MCNP documentation Report
Issued:	2017-05-10

Verification of MCNP6.2 for Nuclear Criticality Safety Applications

Forrest B. Brown, Michael E. Rising, Jennifer L. Alwin

Monte Carlo Methods, Codes. & Applications Group (XCP-3)
Los Alamos National Laboratory, PO Box 1663, MS A143, Los Alamos NM 87545

1.0 INTRODUCTION

Several suites of verification/validation benchmark problems were run in early 2017 to verify that the new production release of MCNP6.2 performs correctly for nuclear criticality safety applications (NCS). MCNP6.2 results for several NCS validation suites were compared to the results from MCNP6.1 [1] and MCNP6.1.1 [2]. MCNP6.1 is the production version of MCNP® released in 2013, and MCNP6.1.1 is the update released in 2014. MCNP6.2 includes all of the standard features for NCS calculations that have been available for the past 15 years, along with new features for sensitivity-uncertainty based methods for NCS validation [3]. Results from the benchmark suites were compared with results from previous verification testing [4-8].

Several standard criticality benchmark suites were used for the verification calculations:

- **VERIFICATION_KEFF** [9-11] – A suite of criticality problems for which exact analytical results are available,
- **VALIDATION_CRITICALITY** [12] – 31 *ICSBEP* [13] problems, using ENDF/B-VII.1 [14],
- **VALIDATION_CRIT_EXPANDED** [15] - 119 *ICSBEP* problems, using ENDF/B-VII.1.
- **VALIDATION_ROSSI_ALPHA** [16] – 13 *ICSBEP* problems, using ENDF/B-VII.0 and ENDF/B-VII.1.

Over 1.5 x 10^9 active neutrons were run in the course of those calculations. The principal conclusion from the extensive NCS testing is that MCNP6.2 performs correctly, in that results for nearly all problems match results from MCNP6.1 and MCNP6.1.1. In a very few cases, results for MCNP6.2 differ by about 1 standard-deviation or less due to known bug fixes ($S(\alpha,\beta)$ numerics; coincident surface treatment; k-adjoint first k-effective estimate) or compiler differences (Intel-12 for previous versions vs. Intel 17 for MCNP6.2). No unusual or unexplained differences were found. In addition, MCNP6.2 was found to run about twice as fast as MCNP6.1 for NCS applications. MCNP6.2 is as correct, robust, and reliable for NCS applications as MCNP5, MCNP6.1, and MCNP6.1.1.

2.0 METHODOLOGY AND BACKGROUND

2.1 Analytic Criticality Suite

The VERIFICATION_KEFF verification suite has traditionally included 75 problems from [9-11] that were run as multigroup problems with MCNP. For the current testing, the verification suite has been completely revised and reconfigured [11,17]. It should be noted that previous usage of the

VERIFICATION_KEFF suite made use of different coding in MCNP6, the multigroup coding, that is never used in realistic NCS calculations. With the modifications to the suite, the problems can now exercise the continuous-energy coding portions of MCNP6, the same coding that is used in realistic NCS calculations. (Of course, the continuous-energy physics in this suite is limited to 1-speed problems with elastic scattering, but at least the overall flow of the calculation stays involves the standard continuous-energy portions of MCNP6.)

2.2 Criticality Validation Suites

All of the testing for VALIDATION_CRITICALITY and VALIDATION_CRIT_EXPANDED was previously performed using ENDF/B-VII.0 nuclear data, so that comparisons could be made with the older MCNP5-1.60 code. (All versions of MCNP5 used the discrete $S(\alpha,\beta)$ thermal scattering model and data, and could not make use of the continuous-energy $S(\alpha,\beta)$ data released with ENDF/B-VII.1.) For the current testing, only the ENDF/B-VII.1 data was used, with continuous-energy $S(\alpha,\beta)$ thermal scattering.

2.3 Rossi Alpha Validation Suite

Since the initial work on the VALIDATION_ROSSI_ALPHA test suite was performed in 2011 using MCNP5 1.60 with ENDF/B-VI and ENDF/B-VII.0, these test problems have not been updated to run with MCNP6 and ENDF/B-VII.1 until now. Because some time has elapsed since the previous effort, various combinations of MCNP5 and MCNP6 with ENDF/B-VII.0 and ENDF/B-VII.1 are used in the present work to show the evolution of the results in these test suites when compilers/hardware change, nuclear data is updated and when minor bug fixes are introduced into the code.

2.4 MCNP6 Coding Changes

2.4.1 Continuous $S(\alpha,\beta)$ numerics

Regarding bug fixes, MCNP6.1 had a small, infrequent error in dealing with the continuous-energy $S(\alpha,\beta)$ data: For some $S(\alpha,\beta)$ datasets at the very lowest energies (typically 10^{-5}-10^{-4} eV), NJOY lumps together scattering probabilities smaller than 10^{-6}. MCNP6.1 did not handle that properly. This problem was fixed in MCNP6.1.1 (MCNP TeamForge Artifact 25705). While the effect of this problem has insignificant impact on results, there should be some very minor differences in a few results for problems with thermal scattering using MCNP6.1 and MCNP6.1.1. MCNP6.1.1 and MCNP6.2 use the same (corrected) coding for $S(\alpha,\beta)$ thermal scattering, so should give the same results.

After the release of MCNP6.1.1 with the $S(\alpha,\beta)$ fix, additional problems were found with roundoff errors for certain $S(\alpha,\beta)$ datasets. In particular, the zr-h.20t and zr-h.30t data. In the continuous-energy sampling scheme for the exit energy, randomly sampling is performed for a linear probability density, requiring a square root. For some sequences of random numbers, roundoff problems led to improper cancellation and the square root of a negative number. For MCNP6.2, additional checks on this roundoff were introduced, and if needed the sampling is performed by a different, robust method that avoids the negative square roots. Since this roundoff problem was extremely rare, the different robust method is only used if needed. In nearly all cases, the previous method works correctly. This hybrid approach was taken to avoid changing the random number usage for all MCNP problems. Only the rare problems affected by the roundoff error use different random number sequences, hence verification-validation testing is unchanged except for a very few cases.

2.4.2 Coincident surface treatment

The *universe* and *fill* concepts were introduced into MCNP in the late 1980s. That is, when defining a cell in MCNP input, the cell can be filled with a universe rather than a single homogeneous material. We will refer to the cell being defined and filled as a *container* cell. A universe is a collection of cells (tagged with the same *u=n* universe number *n*). The problem encountered with the original universe/fill treatment occurred when a bounding surface of one or more cells in a universe was coincident with one of the container bounding surfaces. When this occurred, MCNP sometimes made a wrong decision on which surface a particle had hit (i.e., in a universe cell or the container cell), and lost particles or silent errors were the result.

In the early 1990s, a "fix" for the coincident-surface problem was introduced, first appearing in the release of MCNP4C in 2000. Unfortunately, that fix was flawed. It relied on preprocessing the bounding surface data for all cells and only considered coincident planes, but did not account for possible rotations that can be specified for filling a container with a (rotated) universe. Thus, if a universe was rotated on-the-fly during tracking when filling a container cell, then lost particles or silent errors could be produced. By accident, the coincident-surface fix worked correctly for 0^o and 180^o rotations, but was incorrect for all other rotations. There was also an absolute tolerance of 0.0001 cm used in the scheme for selecting the surface that was hit. (The tolerance could be changed by the *dbcn(9)* input entry.)

For MCNP6.2, the coincident surface treatment was revised. Preprocessing to search for possibly coincident surfaces was eliminated. Instead, all planar surfaces are flagged as possibly coincident. During tracking in a cell contained in a universe, the distances to the bounding surfaces at all universe levels are examined, and the minimum distance is retained. Each distance has an associated level or depth, with level=0 the "real world," level=1 the next deeper universe in the geometry hierarchy, level=2 next deeper, etc. Then, to allow for roundoff in the distance calculations, starting at the smallest depth or level (closest to 0), distances are examined in order of depth to see if they are within a <u>relative</u> tolerance of $\pm10^{-6}$ from the minimum distance. If so, that distance is the one selected, and the remaining distances are ignored. A relative tolerance of $\pm10^{-6}$ is entirely plausible and consistent as an estimate of possible roundoff in the distance calculations that are performed using 53-bit-precision IEEE standard arithmetic. Retaining the smallest distance (within the roundoff tolerance) at the least-deep level is what is desired. Note that this distance may actually be larger than the distance at a different (deeper) level, but is the correct logical choice given arithmetic roundoff. This choice prevents the selection of an incorrect surface distance.

The newly revised coincident-surface treatment is the default for MCNP6.2, with a default relative tolerance for distance roundoff checking of $\pm10^{-6}$. The older, flawed treatment can be used instead if desired, by setting *dbcn(100)* to a nonzero value. If the older treatment is used, the default for roundoff checking is an absolute distance of 0.0001 cm. For either the new or old treatment, the default for checking distance roundoff can be overridden by setting *dbcn(9)*, to a relative value for the new treatment or an absolute value in cm for the old treatment.

It is unavoidable that some, but not all, problems that use the universe/fill capabilities will show different results with the new coincident-surface treatment versus the old one. This is due to the different approaches to dealing with arithmetic roundoff in the distance calculations. The new coincident surface logic prevents errors when rotated fills are used and is the preferred treatment. In our testing experience, both new and old treatments gave the same results within statistics for all problems that did not involve rotated fills. For problems with rotated fills and coincident-surfaces, the new approach was correct, and the old approach was incorrect.

The use of the *dbcn(100)* option to choose between old and new coincident-surface treatments is provided for a limited time, to permit users to run a problem either way for verification purposes. It is likely that this option will be removed in the next future release (after MCNP6.2).

2.4.3 k-adjoint first k-effective estimate

During the calculation of the adjoint-weighted reactor kinetics parameters, MCNP computes an estimate of K_{eff} based upon the K_{eff} in the iterated fission probably method block. The way this was originally implemented, the block K_{eff} estimate was initialized at the end of the block after the first adjoint-weighted tally scores were made. Consequently, the first estimate of these tallies utilized K_{eff} information from the inactive cycles introducing a small bias. Shortly after the MCNP6.1.1 official release, the coding for this was fixed, with the block-estimate of K_{eff} now initialized at the beginning of the block.

This bug fix does change the results of the adjoint-weighted calculation of the reactor kinetics parameters. However, this change is very small, generally much smaller than the statistics of the tallies computed. In the case where a user is generally conservative when setting the number of inactive cycles, by discarding more cycles than necessary (even just a few), this bug fix has no impact on the quality of the results.

2.5 Fortran Compiler Issues

An important part of the recent testing was a comparison of results obtained from MCNP6.1 and MCNP6.1.1 compiled with the Intel-12 Fortran compiler versus MCNP6.2 compiled with the Intel-17 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 NCS applications. However, when switching to a newer, different compiler, it is generally not possible to avoid some minor differences in results caused by different arithmetic roundoff between the compilers. 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. 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. Such roundoff differences are normally less than the statistical error of the results. In rare cases where that is not true, serious focused investigation into any differences must be performed and documented.

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. Performance testing 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 MCNP. Nevertheless, computer roundoff differences will occur with different compilers/hardware.

3.0 TESTING RESULTS

The criticality verification/validation suites were run on both Mac OS X and Linux systems with MCNP6.1, MCNP6.1.1, and MCNP6.2. For Mac OS X, the suites were run on a Mac Pro computer using 64-bit executables, 12-core Xeon processor with 2 hyperthreads/core, OS X 10.11.6 & 10.12.4, and 12 MCNP threads. For Linux, the suites were run on a single node of a LANL cluster, with 64-bit executables, 8 dual-core Xeon processors, Chaos Linux, and 16 MCNP threads. For Windows, the suites were run on a Windows laptop using 64-bit executable, a quad-core I7-4930MX with hyperthreading, Windows 7, and 8 MCNP threads. (Only a few Windows results are presented in this work. Regular testing is performed to ensure that Mac, Linux, and Windows results match.)

3.1 VERIFICATION_KEFF Suite

For the VERIFICATION_KEFF suite, MCNP results can be compared to exact results from analytic benchmark problems. For MCNP6.1 and MCNP6.1.1, results were reported in [4-6] for the analytic test problems run using the multigroup mode in MCNP6. In the current testing, MCNP6.2 was run using both multigroup and continuous-energy treatments for 38 analytic benchmark problems. The results from this testing are detailed in **Figure 1** and summarized here. The conclusions are:

- MCNP6.2 gives correct results for the analytic problems when run in multigroup mode. The absolute accuracy of the results is within 3 pcm ± 3 pcm. (1 pcm = 0.00001)
- MCNP6.2 gives correct results for the analytic problems when run in continuous-energy mode. The absolute accuracy of the results is within 3 pcm ± 3 pcm.

3.2 VALIDATION_CRITICALITY Suite

Table 2 shows the K_{eff} results for 31 *ICSBEP* benchmark problems for MCNP6.1, MCNP6.1.1, and MCNP6.2 for Mac OS X, and **Table 3** shows the results for a Linux system. For MCNP6.2, results are presented for both the old and new coincident-surface treatments. To simplify the comparisons, the table shows the MCNP6.1 results and differences that arise for MCNP6.1.1 and MCNP6.2. Cases that show differences are highlighted in red in the tables, and the reasons for the differences are noted for each case. **Table 4** shows a comparison of MCNP6.2 results for Mac, Linux, and Windows systems.

To summarize the results of the present testing with the criticality validation suites and ENDF/B-VII.1 data, for 31 separate *ICSBEP* problems tested:

- On Mac OS X, 4 MCNP6.2 problems showed differences from MCNP6.1 or MCNP6.1.1. The differences were less than 2 combined standard deviations. ICT2C3 differences are due to the S(α,β)

fixes; ZEBR8H to compiler roundoff differences; SB25 and BAWXI2 to roundoff from the new coincident-surface treatment. MCNP6.2 was 1.7 times faster than MCNP6.1 for this suite.

- Linux, 3 MCNP6.2 problems showed differences from MCNP6.1 or MCNP6.1.1. The differences were less than 2 combined standard deviations. The differences were the same as for Mac OS X, except that the ZEBR8H compiler differences did not occur. This is not unexpected, since Mac and Linux compilers sometimes differ in arithmetic roundoff. MCNP6.2 was 2.0 times faster than MCNP6.1 for this suite.
- Comparing Mac OS X, Linux, and Windows results for MCNP6.2 shows agreement in 30 of 31 cases. The one difference for Windows is due to roundoff from compiler differences and possibly the slight differences in cpu hardware (I7 on Windows, Xeon on Mac and Linux). The difference on Windows is less than 1 standard-deviation.

3.3 VALIDATION_CRIT_EXPANDED Suite

For this benchmark suite, 119 *ICSBEP* benchmark problems were run on both Mac OS X and Linux using MCNP6.1, MCNP6.1.1, and MCNP6.2. Results are shown in **Table 5 (a-d)** for Mac OS X and **Table 6 (a-d)** for Linux. For both Mac and Linux, all results for MCNP6.1, MCNP6.1.1, and MCNP6.2 (with old coincident-surface) are identical, and MCNP6.2 shows 11 cases where there are differences of about 1 standard-deviation or less due to the different roundoff for the new coincident-surface treatment. MCNP6.2 is 1.9 times faster than MCNP6.1 on Mac OS X, and 2.2 times faster than MCNP6.1 on Linux.

3.4 VALIDATION_ROSSI_ALPHA Suite

This benchmark suite consists of 13 *ICSBEP* benchmark problems run on Linux using MCNP5 1.60 and MCNP6.1 using ENDF/B-VII.0 cross sections and using MCNP6.1, MCNP6.1.1 and MCNP6.2 using ENDF/B-VII.1 cross sections. Along with the experimental benchmark results, **Table 7** shows all of the computed results. The highlighted values indicate a difference compared with the adjacent column to the left and the asterisks indicate the magnitude of the difference. There are two differences between MCNP5 1.60 and MCNP6.1 using ENDF/B-VII.0 data that are likely due to compiler/hardware roundoff differences. Changing the nuclear data from ENDF/B-VII.0 to ENDF/B-VII.1 resulted in all except one benchmark changed with no consistent trend in the new values with respect to the experimental benchmark values. Finally, MCNP6.1 and MCNP6.1.1 gave identical results, and MCNP6.2 gave different results due to the k-adjoint first k-effective estimate bug fix described previously. While all of the test problems did give very small differences due to this bug fix, most were less than 1 standard-deviation with only two differences observed in the final decimal place shown in the quoted values in **Table 7**.

SUMMARY AND CONCLUSIONS

Table 8 provides a summary of the criticality validation suite results for the recent testing of MCNP6.1, MCNP6.1.1, and MCNP6.2 for NCS applications. The general conclusions from this testing are:

- MCNP6.1, MCNP6.1.1, and MCNP6.2 perform correctly for NCS applications.
- The VERIFICATION_KEFF results indicate that all versions of MCNP6 are accurate to within 3±3 pcm when exact simple cross-sections are used for analytic benchmarks.
- While small differences were noted for 15 out of 150 *ICSBEP* criticality-only problems and 2 out of 13 *ICSBEP* Rossi-α problems, these are strictly due to arithmetic roundoff from different compilers, a minor S(α,β) bug-fix, different arithmetic roundoff from the new coincident-surface treatment, or the fixed k-adjoint first k-effective estimate bug, and are not a concern for verification/validation.
- MCNP6.1, MCNP6.1.1, and MCNP6.2 yield the same results on different computer platforms – Mac OS X, Linux, and Windows – for NCS applications.

Criticality safety analysts should consider testing MCNP6.2 on their particular problems and validation suites. No further development of MCNP5 is planned. MCNP6.1 is now 4 years old, and MCNP6.1.1 is now 3 years old. In general, released versions of MCNP are supported only for about 5 years, due to resource limitations. All future MCNP improvements, bug fixes, user support, and new capabilities are targeted only to MCNP6.2 and beyond.

ACKNOWLEDGMENTS

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

REFERENCES

1. J.T. Goorley, et al., "Initial MCNP6 Release Overview," *Nuclear Technology*, **180**, 298 (2012).
2. J.T. Goorley, "MCNP6.1.1-Beta Release Notes," LA-UR-14-24680 (2014).
3. B.C. Kiedrowski, F.B. Brown, et al., "Whisper: Sensitivity/Uncertainty-Based Computational Methods and Software for Determining Upper Subcritical Limits", *Nucl. Sci. Eng.* **181**, 17-47 [also LA-UR-14-26558] (2015).
4. F.B. Brown, "MCNP Verification & Validation", DOE-NCSP Analytical Methods Working Group meeting, Charlotte NC, Sept 2015, LA-UR-15-27015 (2015).
5. F.B. Brown, "MCNP6 Optimization and Testing for Criticality Safety Calculations", *Trans. ANS* **112**, San Antonio, TX, June 2015, LA-UR-15-20422 (2015).
6. F.B. Brown, B.C. Kiedrowski, J.S. Bull, "Verification of MCNP6.1 and MCNP6.1.1 for Criticality Safety Applications", LA-UR-14-22480 (2014).
7. F.B. Brown, B.C. Kiedrowski, J.S. Bull, "Verification of MCNP5-1.60 and MCNP6.1 for Criticality Safety Applications", LA-UR-13-22196 (2013).
8. F.B. Brown, M.E. Rising, "Verification of MCNP6.1, MCNP6.1.1, and MCNP6.2-pre for Criticality Safety Applications", LA-UR-16-24308 (2016).
9. A. Sood, R.A. Forster, D.K. Parsons, "Analytic Benchmark Test Set for Criticality Code Verification," *Prog. Nucl. Energy*, **42**, 55-106 (2003).

10. A. Sood, R.A. Forster, D.K. Parsons, "Analytic Benchmark Test Set for Criticality Code Verification", LA-13511 and LA-UR-01-3082 (2001)

11. F.B. Brown, “The MCNP6 Analytic Criticality Benchmark Suite”, LA-UR-16-24255 (2016).

12. 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)

13. International Handbook of Evaluated Criticality Safety Benchmark Experiments, NEA No. 7281, OECD Nuclear Energy Agency (2015).

14. 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).

15. R.D. MOSTELLER, "An Expanded Criticality Validation Suite for MCNP", LA-UR-10-06230 Rev3 (2010).

16. R.D. Mosteller, B.C. Kiedrowski, “The Rossi Alpha Validation Suite for MCNP”, International Conference on Criticality, Edinburgh, Scotland, 19-22 Sept. 2011, LA-UR-11-04409 (2011).

17. F.B. Brown, “New Tools to Prepare ACE Cross-section Files for MCNP Analytic Test Problems”, LA-UR-16-24290(2016).

Table 1. MCNP6 Criticality Results vs Exact Results

Case	Name	Analytic keff	MCNP_Multigroup C/E-1	MCNP_Multigroup std		MCNP Continuous Energy C/E-1	MCNP Continuous Energy std	
01	PUa-1-0-IN	2.61290	-0 pcm	0		-0 pcm	0	
02	PUa-1-0-SL	1.00000	0	5		6	5	
03	PUa-H2O(1)-1-0-SL	1.00000	8	5	*	1	5	
04	PUa-H2O(0.5)-1-0-SL	1.00000	2	5		3	5	
05	PUb-1-0-IN	2.29032	-0	0		-0	0	
06	PUb-1-0-SL	1.00000	4	4		0	4	
07	PUb-1-0-CY	1.00000	-4	4	*	3	4	
08	PUb-1-0-SP	1.00000	6	4	*	6	4	*
09	PUb-H2O(1)-1-0-CY	1.00000	-3	4		5	4	
10	PUb-H2O(10)-1-0-CY	1.00000	5	4		5	5	
11	Ua-1-0-IN	2.25000	0	0		0	0	
12	Ua-1-0-SL	1.00000	6	4	*	-3	4	
13	Ua-1-0-CY	1.00000	4	4		3	4	
14	Ua-1-0-SP	1.00000	1	4		-5	4	*
15	Ub-1-0-IN	2.33092	0	0		0	0	
16	Ub-H2O(1)-1-0-SP	1.00000	-2	4		-1	4	
17	Uc-1-0-IN	2.25608	0	0		0	0	
18	Uc-H2O(2)-1-0-SP	1.00000	-1	4		0	4	
19	Ud-1-0-IN	2.23267	-0	0		-0	0	
20	Ud-H2O(3)-1-0-SP	1.00000	4	4		7	4	*
21	UD2O-1-0-IN	1.13333	-0	0		-0	0	
22	UD2O-1-0-SL	1.00000	3	2		0	2	
23	UD2O-1-0-CY	1.00000	-1	2		-5	2	**
24	UD2O-1-0-SP	1.00000	1	3		-4	2	**
25	UD2O-H2O(1)-1-0-SL	1.00000	2	2		-2	2	*
26	UD2O-H2O(10)-1-0-SL	1.00000	-5	2	**	1	2	
27	UD2O-H2O(1)-1-0-CY	1.00000	4	2	*	-1	2	
28	UD2O-H2O(10)-1-0-CY	1.00000	0	2		3	2	
29	Ue-1-0-IN	2.18067	0	0		0	0	
30	Ue-Fe-Na-1-0-SL	1.00000	-1	5		7	4	*
31	PU-1-1-IN	2.50000	0	0		0	0	
32	PUa-1-1-SL	1.00000	8	5	*	7	5	*
36	Ua-1-1-CY	1.00000	2	4		-3	4	
38	UD2Oa-1-1-IN	1.20559	0	0		0	0	
39	UD2Oa-1-1-SP	1.00000	-2	3		2	3	
40	UD2Ob-1-1-IN	1.22739	-0	0		-0	0	
41	UD2Ob-1-1-SP	1.00000	8	3	**	6	3	*
		RMS Differences	3 pcm	±3 pcm		3 pcm	±3 pcm	

1 pcm = 0.00001

Table 2. VALIDATION_CRITICALITY – Mac OS X - 2017-03-28

610_12_71 – 2013, mcnp6.1, Intel-12, endf/b-vii.1
611_12_71 – 2014, mcnp6.1.1, Intel-12, endf/b-vii.1
621_17_71 – 2017, mcnp6.2, Intel-17, endf/b-vii.1, with old coincident-surface treatment
620_17_71 – 2017, mcnp6.2, Intel-17, endf/b-vii.1, with new coincident-surface treatment

	610_12_71_mac	611_12_71_mac	621_17_71_mac	620_17_71_mac	
	keff std	deltak std	deltak std	deltak std	Reason for diffs
U233 Benchmarks			OLD coinc-sur	NEW coinc-sur	
JEZ233	1.0000 (5)	0.0000 (8)	0.0000 (8)	0.0000 (8)	
FLAT23	0.9974 (7)	0.0000 (9)	0.0000 (9)	0.0000 (9)	
UMF5C2	0.9960 (7)	0.0000 (9)	0.0000 (9)	0.0000 (9)	
FLSTF1	0.9845 (11)	0.0000 (15)	0.0000 (15)	0.0000 (15)	
SB25	0.9997 (10)	0.0000 (14)	0.0000 (14)	0.0009 (14)	roundoff, coinc-sur
ORNL11	1.0018 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)	
HEU Benchmarks					
GODIVA	0.9988 (5)	0.0000 (8)	0.0000 (8)	0.0000 (8)	
TT2C11	1.0009 (8)	0.0000 (11)	0.0000 (11)	0.0000 (11)	
FLAT25	1.0034 (5)	0.0000 (8)	0.0000 (8)	0.0000 (8)	
GODIVR	0.9989 (7)	0.0000 (9)	0.0000 (9)	0.0000 (9)	
UH3C6	0.9957 (8)	0.0000 (11)	0.0000 (11)	0.0000 (11)	
ZEUS2	0.9976 (7)	0.0000 (9)	0.0000 (9)	0.0000 (9)	
SB5RN3	0.9945 (13)	0.0000 (18)	0.0000 (18)	0.0000 (18)	
ORNL10	1.0001 (4)	0.0000 (5)	0.0000 (5)	0.0000 (5)	
IEU Benchmarks					
IMF03	1.0019 (5)	0.0000 (8)	0.0000 (8)	0.0000 (8)	
BIGTEN	0.9952 (5)	0.0000 (7)	0.0000 (7)	0.0000 (7)	
IMF04	1.0082 (5)	0.0000 (8)	0.0000 (8)	0.0000 (8)	
ZEBR8H	1.0193 (5)	0.0000 (8)	-0.0011 (8)*	-0.0011 (8)*	roundoff, compiler
ICT2C3	1.0023 (7)	0.0012 (9)*	0.0012 (9)*	0.0012 (9)*	Sab-fix
STACY36	0.9981 (5)	0.0000 (8)	0.0000 (8)	0.0000 (8)	
LEU Benchmarks					
BAWXI2	1.0025 (5)	0.0000 (8)	0.0000 (8)	-0.0004 (8)	roundoff, coinc-sur
LST2C2	0.9960 (5)	0.0000 (8)	0.0000 (8)	0.0000 (8)	
Pu Benchmarks					
JEZPU	0.9990 (5)	0.0000 (8)	0.0000 (8)	0.0000 (8)	
JEZ240	0.9999 (5)	0.0000 (8)	0.0000 (8)	0.0000 (8)	
PUBTNS	0.9980 (7)	0.0000 (9)	0.0000 (9)	0.0000 (9)	
FLATPU	1.0004 (7)	0.0000 (9)	0.0000 (9)	0.0000 (9)	
THOR	0.9976 (5)	0.0000 (8)	0.0000 (8)	0.0000 (8)	
PUSH2O	1.0013 (8)	0.0000 (11)	0.0000 (11)	0.0000 (11)	
HISHPG	1.0121 (5)	0.0000 (8)	0.0000 (8)	0.0000 (8)	
PNL2	1.0050 (10)	0.0000 (14)	0.0000 (14)	0.0000 (14)	
PNL33	1.0068 (7)	0.0000 (9)	0.0000 (9)	0.0000 (9)	
Wall-clock:	19.6 min	11.8 min	11.8 min	11.7 min	
Threads:	12	12	12	12	
Rel. Speed:	1.00	1.66	1.66	1.67	

Table 3. VALIDATION_CRITICALITY – Linux - 2017-04-07

610_12_71 – 2013, mcnp6.1, Intel-12, endf/b-vii.1
611_12_71 – 2014, mcnp6.1.1, Intel-12, endf/b-vii.1
621_17_71 – 2017, mcnp6.2, Intel-17, endf/b-vii.1, with old coincident-surface treatment
620_17_71 – 2017, mcnp6.2, Intel-17, endf/b-vii.1, with new coincident-surface treatment

	610_12_71_lin	611_12_71_lin	621_17_71_lin	620_17_71_lin	
	keff std	deltak std	deltak std	deltak std	Reason for diffs
U233 Benchmarks					
JEZ233	1.0000 (5)	0.0000 (8)	0.0000 (8)	0.0000 (8)	
FLAT23	0.9974 (7)	0.0000 (9)	0.0000 (9)	0.0000 (9)	
UMF5C2	0.9960 (7)	0.0000 (9)	0.0000 (9)	0.0000 (9)	
FLSTF1	0.9845 (11)	0.0000 (15)	0.0000 (15)	0.0000 (15)	
SB25	0.9997 (10)	0.0000 (14)	0.0000 (14)	0.0009 (14)	roundoff, coinc-sur
ORNL11	1.0018 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)	
HEU Benchmarks					
GODIVA	0.9988 (5)	0.0000 (8)	0.0000 (8)	0.0000 (8)	
TT2C11	1.0009 (8)	0.0000 (11)	0.0000 (11)	0.0000 (11)	
FLAT25	1.0034 (5)	0.0000 (8)	0.0000 (8)	0.0000 (8)	
GODIVR	0.9989 (7)	0.0000 (9)	0.0000 (9)	0.0000 (9)	
UH3C6	0.9957 (8)	0.0000 (11)	0.0000 (11)	0.0000 (11)	
ZEUS2	0.9976 (7)	0.0000 (9)	0.0000 (9)	0.0000 (9)	
SB5RN3	0.9945 (13)	0.0000 (18)	0.0000 (18)	0.0000 (18)	
ORNL10	1.0001 (4)	0.0000 (5)	0.0000 (5)	0.0000 (5)	
IEU Benchmarks					
IMF03	1.0019 (5)	0.0000 (8)	0.0000 (8)	0.0000 (8)	
BIGTEN	0.9952 (5)	0.0000 (7)	0.0000 (7)	0.0000 (7)	
IMF04	1.0082 (5)	0.0000 (8)	0.0000 (8)	0.0000 (8)	
ZEBR8H	1.0182 (5)	0.0000 (8)	0.0000 (8)	0.0000 (8)	
ICT2C3	1.0023 (7)	0.0012 (9)*	0.0012 (9)*	0.0012 (9)*	Sab-fix
STACY36	0.9981 (5)	0.0000 (8)	0.0000 (8)	0.0000 (8)	
LEU Benchmarks					
BAWXI2	1.0025 (5)	0.0000 (8)	0.0000 (8)	-0.0004 (8)	roundoff, coinc-sur
LST2C2	0.9960 (5)	0.0000 (8)	0.0000 (8)	0.0000 (8)	
Pu Benchmarks					
JEZPU	0.9990 (5)	0.0000 (8)	0.0000 (8)	0.0000 (8)	
JEZ240	0.9999 (5)	0.0000 (8)	0.0000 (8)	0.0000 (8)	
PUBTNS	0.9980 (7)	0.0000 (9)	0.0000 (9)	0.0000 (9)	
FLATPU	1.0004 (7)	0.0000 (9)	0.0000 (9)	0.0000 (9)	
THOR	0.9976 (5)	0.0000 (8)	0.0000 (8)	0.0000 (8)	
PUSH2O	1.0013 (8)	0.0000 (11)	0.0000 (11)	0.0000 (11)	
HISHPG	1.0121 (5)	0.0000 (8)	0.0000 (8)	0.0000 (8)	
PNL2	1.0050 (10)	0.0000 (14)	0.0000 (14)	0.0000 (14)	
PNL33	1.0068 (7)	0.0000 (9)	0.0000 (9)	0.0000 (9)	
Wall-clock:	18.9 min	10.2 min	9.6 min	9.6 min	
Threads:	16	16	16	16	
Rel. Speed:	1.00	1.86	1.97	1.97	

Table 4. VALIDATION_CRITICALITY – Mac, Linux, WIndows

```
620_17_71_mac    = mcnp6.2.0 + Intel 17  + endf/b-vii.1 + macosx
620_17_71_lin    = mcnp6.2.0 + Intel 17  + endf/b-vii.1 + linux
620_17_71_win    = mcnp6.2.0 + Intel 17  + endf/b-vii.1 + windows
```

	620_17_71_mac		620_17_71_lin		620_17_71_win		
	keff	std	deltak	std	deltak	std	Reason for diffs
U233 Benchmarks							
JEZ233	1.0000	(5)	0.0000	(8)	0.0000	(8)	
FLAT23	0.9974	(7)	0.0000	(9)	0.0000	(9)	
UMF5C2	0.9960	(7)	0.0000	(9)	0.0000	(9)	
FLSTF1	0.9845	(11)	0.0000	(15)	0.0000	(15)	
SB25	1.0006	(10)	0.0000	(14)	0.0000	(14)	
ORNL11	1.0018	(2)	0.0000	(4)	0.0000	(4)	
HEU Benchmarks							
GODIVA	0.9988	(5)	0.0000	(8)	0.0000	(8)	
TT2C11	1.0009	(8)	0.0000	(11)	0.0000	(11)	
FLAT25	1.0034	(5)	0.0000	(8)	0.0000	(8)	
GODIVR	0.9989	(7)	0.0000	(9)	0.0000	(9)	
UH3C6	0.9957	(8)	0.0000	(11)	0.0000	(11)	
ZEUS2	0.9976	(7)	0.0000	(9)	0.0000	(9)	
SB5RN3	0.9945	(13)	0.0000	(18)	0.0000	(18)	
ORNL10	1.0001	(4)	0.0000	(5)	0.0000	(5)	
IEU Benchmarks							
IMF03	1.0019	(5)	0.0000	(8)	0.0000	(8)	
BIGTEN	0.9952	(5)	0.0000	(7)	0.0000	(7)	
IMF04	1.0082	(5)	0.0000	(8)	0.0000	(8)	
ZEBR8H	1.0182	(5)	0.0000	(8)	0.0004	(7)	compiler roundoff diffs
ICT2C3	1.0035	(7)	0.0000	(9)	0.0000	(9)	
STACY36	0.9981	(5)	0.0000	(8)	0.0000	(8)	
LEU Benchmarks							
BAWXI2	1.0021	(5)	0.0000	(8)	0.0000	(8)	
LST2C2	0.9960	(5)	0.0000	(8)	0.0000	(8)	
Pu Benchmarks							
JEZPU	0.9990	(5)	0.0000	(8)	0.0000	(8)	
JEZ240	0.9999	(5)	0.0000	(8)	0.0000	(8)	
PUBTNS	0.9980	(7)	0.0000	(9)	0.0000	(9)	
FLATPU	1.0004	(7)	0.0000	(9)	0.0000	(9)	
THOR	0.9976	(5)	0.0000	(8)	0.0000	(8)	
PUSH2O	1.0013	(8)	0.0000	(11)	0.0000	(11)	
HISHPG	1.0121	(5)	0.0000	(8)	0.0000	(8)	
PNL2	1.0050	(10)	0.0000	(14)	0.0000	(14)	
PNL33	1.0068	(7)	0.0000	(9)	0.0000	(9)	
Wall-clock:	11.7 min		9.6 min		20.6 min		
Threads:	12		16		8		
Rel. Speed:	1.00		0.91		0.85		

Table 5a. VALIDATION_CRIT_EXPANDED – Mac OS X – 2017-04-11

```
610_12_71_mac     = mcnp6.1   + Intel 12 + endf/b-vii.1 + macosx
611_12_71_mac     = mcnp6.1.1 + Intel 12 + endf/b-vii.1 + macosx
621_17_71_mac     = mcnp6.2.0 + Intel 17 + endf/b-vii.1 + macosx, with old coincident-surface treatment
620_17_71_mac     = mcnp6.2.0 + Intel 17 + endf/b-vii.1 + macosx, with new coincident-surface treatment
```

	610_12_71_mac	611_12_71_mac	621_17_71_mac	620_17_71_mac	
	keff std	deltak std	deltak std	deltak std	
U233 Benchmarks					
u233-met-fast-001	1.0000 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)	
u233-met-fast-002-case-1	0.9983 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)	
u233-met-fast-002-case-2	1.0003 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)	
u233-met-fast-003-case-1	0.9995 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)	
u233-met-fast-003-case-2	0.9995 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)	
u233-met-fast-006	0.9984 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)	
u233-met-fast-004-case-1	0.9988 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)	
u233-met-fast-004-case-2	0.9956 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)	
u233-met-fast-005-case-1	0.9959 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)	
u233-met-fast-005-case-2	0.9952 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)	
u233-sol-inter-001-case-1	0.9845 (5)	0.0000 (7)	0.0000 (7)	0.0000 (7)	
u233-comp-therm-001-case-3	1.0034 (4)	0.0000 (5)	0.0000 (5)	-0.0006 (5)*	coinc r/o
u233-sol-therm-001-case-1	1.0010 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)	
u233-sol-therm-001-case-2	1.0010 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)	
u233-sol-therm-001-case-3	1.0007 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)	
u233-sol-therm-001-case-4	1.0007 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)	
u233-sol-therm-001-case-5	0.9996 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)	
u233-sol-therm-008	1.0016 (2)	0.0000 (2)	0.0000 (2)	0.0000 (2)	
LEU Benchmarks					
leu-comp-therm-008-case-1	1.0006 (2)	0.0000 (4)	0.0000 (4)	-0.0005 (4)*	coinc r/o
leu-comp-therm-008-case-2	1.0005 (2)	0.0000 (4)	0.0000 (4)	0.0002 (4)	coinc r/o
leu-comp-therm-008-case-5	1.0006 (2)	0.0000 (4)	0.0000 (4)	0.0004 (4)	coinc r/o
leu-comp-therm-008-case-7	1.0004 (2)	0.0000 (4)	0.0000 (4)	-0.0004 (4)	coinc r/o
leu-comp-therm-008-case-8	0.9997 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)	
leu-comp-therm-008-case-11	1.0007 (2)	0.0000 (4)	0.0000 (4)	0.0003 (4)	coinc r/o
leu-sol-therm-002-case-1	0.9994 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)	
leu-sol-therm-002-case-2	0.9964 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)	

Table 5b. VALIDATION_CRIT_EXPANDED – Mac OS X – 2017-04-11

	610_12_71_mac	611_12_71_mac	621_17_71_mac	620_17_71_mac
	keff std	deltak std	deltak std	deltak std
HEU Benchmarks				
heu-met-fast-001	0.9994 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)
heu-met-fast-008	0.9962 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)
heu-met-fast-018-case-2	0.9995 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)
heu-met-fast-003-case-1	0.9949 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)
heu-met-fast-003-case-2	0.9945 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)
heu-met-fast-003-case-3	0.9989 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)
heu-met-fast-003-case-4	0.9974 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)
heu-met-fast-003-case-5	1.0012 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)
heu-met-fast-003-case-6	1.0020 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)
heu-met-fast-003-case-7	1.0019 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)
heu-met-fast-028	1.0027 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)
heu-met-fast-014	0.9977 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)
heu-met-fast-003-case-8	1.0023 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)
heu-met-fast-003-case-9	1.0023 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)
heu-met-fast-003-case-10	1.0052 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)
heu-met-fast-003-case-11	1.0094 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)
heu-met-fast-003-case-12	1.0087 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)
heu-met-fast-013	0.9975 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)
heu-met-fast-021-case-2	0.9979 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)
heu-met-fast-022-case-2	0.9976 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)
heu-met-fast-012	0.9984 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)
heu-met-fast-019-case-2	1.0069 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)
heu-met-fast-009-case-2	0.9966 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)
heu-met-fast-009-case-1	0.9977 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)
heu-met-fast-011	0.9985 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)
heu-met-fast-020-case-2	1.0006 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)
heu-met-fast-004-case-1	1.0034 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)
heu-met-fast-015	0.9947 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)
heu-met-fast-026-case-c-11	1.0032 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)
heu-comp-inter-003-case-6	0.9948 (4)	0.0000 (5)	0.0000 (5)	0.0000 (5)
heu-met-inter-006-case-1	0.9929 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)
heu-met-inter-006-case-2	0.9968 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)
heu-met-inter-006-case-3	1.0008 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)
heu-met-inter-006-case-4	1.0072 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)
u233-comp-therm-001-case-6	0.9988 (4)	0.0000 (5)	0.0000 (5)	0.0000 (5)
heu-sol-therm-013-case-1	0.9985 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)
heu-sol-therm-013-case-2	0.9969 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)
heu-sol-therm-013-case-3	0.9939 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)
heu-sol-therm-013-case-4	0.9953 (2)	0.0000 (4)	0.0000 (4)	0.0000 (4)
heu-sol-therm-032	0.9992 (2)	0.0000 (2)	0.0000 (2)	0.0000 (2)

Table 5c. VALIDATION_CRIT_EXPANDED – Mac OS X – 2017-04-11

	610_12_71_mac		611_12_71_mac		621_17_71_mac		620_17_71_mac		
	keff	std	deltak	std	deltak	std	deltak	std	
Pu Benchmarks									
pu-met-fast-001	0.9993	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-met-fast-002	1.0003	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-met-fast-022-case-2	0.9984	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
mix-met-fast-001	0.9998	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
mix-met-fast-003	1.0004	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-met-fast-006	1.0001	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-met-fast-010	0.9996	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-met-fast-020	0.9983	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-met-fast-008-case-2	0.9977	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-met-fast-005	1.0019	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-met-fast-025-case-2	0.9991	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-met-fast-026-case-2	0.9987	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-met-fast-009	1.0048	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-met-fast-023-case-2	0.9994	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-met-fast-018	0.9993	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-met-fast-019	1.0004	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-met-fast-024-case-2	1.0025	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-met-fast-011	1.0000	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-met-fast-021-case-2	0.9935	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-met-fast-021-case-1	1.0047	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-met-fast-003-case-103	0.9990	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-comp-inter-001	1.0116	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
mix-comp-therm-002-case-pnl30	1.0002	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
mix-comp-therm-002-case-pnl31	1.0018	(4)	0.0000	(5)	0.0000	(5)	-0.0006	(5)*	coinc r/o
mix-comp-therm-002-case-pnl32	1.0020	(2)	0.0000	(4)	0.0000	(4)	-0.0003	(4)	coinc r/o
mix-comp-therm-002-case-pnl33	1.0063	(2)	0.0000	(4)	0.0000	(4)	-0.0001	(4)	coinc r/o
mix-comp-therm-002-case-pnl34	1.0045	(2)	0.0000	(4)	0.0000	(4)	0.0001	(4)	coinc r/o
mix-comp-therm-002-case-pnl35	1.0063	(2)	0.0000	(4)	0.0000	(4)	-0.0004	(4)	coinc r/o
pu-sol-therm-009-case-3a	1.0191	(2)	0.0000	(2)	0.0000	(2)	0.0000	(2)	
pu-sol-therm-011-case-16-5	1.0054	(4)	0.0000	(5)	0.0000	(5)	0.0000	(5)	
pu-sol-therm-011-case-18-1	0.9941	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-sol-therm-011-case-18-6	1.0005	(4)	0.0000	(5)	0.0000	(5)	0.0000	(5)	
pu-sol-therm-021-case-1	1.0053	(4)	0.0000	(5)	0.0000	(5)	0.0000	(5)	
pu-sol-therm-021-case-3	1.0043	(4)	0.0000	(5)	0.0000	(5)	0.0000	(5)	
pu-sol-therm-018-case-9	1.0026	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-sol-therm-034-case-1	1.0007	(4)	0.0000	(5)	0.0000	(5)	0.0000	(5)	

Table 5d. VALIDATION_CRIT_EXPANDED – Mac OS X – 2017-04-11

	610_12_71_mac		611_12_71_mac		621_17_71_mac		620_17_71_mac	
	keff	std	deltak	std	deltak	std	deltak	std
IEU Benchmarks								
ieu-met-fast-003-case-2	1.0028	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
ieu-met-fast-005-case-2	1.0024	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
ieu-met-fast-006-case-2	0.9958	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
ieu-met-fast-004-case-2	1.0075	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
ieu-met-fast-001-case-1	1.0009	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
ieu-met-fast-001-case-2	0.9999	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
ieu-met-fast-001-case-3	1.0011	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
ieu-met-fast-001-case-4	1.0015	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
ieu-met-fast-002	0.9991	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
ieu-met-fast-007-case-4	1.0045	(2)	0.0000	(2)	0.0000	(2)	0.0000	(2)
mix-met-fast-008-case-7	1.0192	(2)	0.0000	(2)	0.0000	(2)	0.0000	(2)
ieu-comp-therm-002-case-3	1.0038	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
leu-sol-therm-007-case-14	0.9947	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
leu-sol-therm-007-case-30	0.9971	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
leu-sol-therm-007-case-32	0.9959	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
leu-sol-therm-007-case-36	0.9990	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
leu-sol-therm-007-case-49	0.9972	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
Wall-clock:	301.2 min		158.7 min		156.0 min		156.4 min	
Threads:	12		12		12		12	
Rel. Speed:	1.00		1.90		1.93		1.93	

Table 6a. VALIDATION_CRIT_EXPANDED – Linux – 2017-04-07

610_12_71_lin = mcnp6.1 + Intel 12 + endf/b-vii.1 + linux
611_12_71_lin = mcnp6.1.1 + Intel 12 + endf/b-vii.1 + linux
621_17_71_lin = mcnp6.2.0 + Intel 17 + endf/b-vii.1 + linux, with old coincident-surface treatment
620_17_71_lin = mcnp6.2.0 + Intel 17 + endf/b-vii.1 + linux, with new coincident-surface treatment

	610_12_71_lin		611_12_71_lin		621_17_71_lin		620_17_71_lin		
	keff	std	deltak	std	deltak	std	deltak	std	
U233 Benchmarks									
u233-met-fast-001	1.0000	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
u233-met-fast-002-case-1	0.9983	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
u233-met-fast-002-case-2	1.0003	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
u233-met-fast-003-case-1	0.9995	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
u233-met-fast-003-case-2	0.9995	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
u233-met-fast-006	0.9984	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
u233-met-fast-004-case-1	0.9988	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
u233-met-fast-004-case-2	0.9956	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
u233-met-fast-005-case-1	0.9959	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
u233-met-fast-005-case-2	0.9952	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
u233-sol-inter-001-case-1	0.9845	(5)	0.0000	(7)	0.0000	(7)	0.0000	(7)	
u233-comp-therm-001-case-3	1.0034	(4)	0.0000	(5)	0.0000	(5)	-0.0006	(5)*	coinc r/o
u233-sol-therm-001-case-1	1.0010	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
u233-sol-therm-001-case-2	1.0010	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
u233-sol-therm-001-case-3	1.0007	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
u233-sol-therm-001-case-4	1.0007	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
u233-sol-therm-001-case-5	0.9996	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
u233-sol-therm-008	1.0016	(2)	0.0000	(2)	0.0000	(2)	0.0000	(2)	
LEU Benchmarks									
leu-comp-therm-008-case-1	1.0006	(2)	0.0000	(4)	0.0000	(4)	-0.0005	(4)*	coinc r/o
leu-comp-therm-008-case-2	1.0005	(2)	0.0000	(4)	0.0000	(4)	0.0002	(4)	coinc r/o
leu-comp-therm-008-case-5	1.0006	(2)	0.0000	(4)	0.0000	(4)	0.0004	(4)	coinc r/o
leu-comp-therm-008-case-7	1.0004	(2)	0.0000	(4)	0.0000	(4)	-0.0004	(4)	coinc r/o
leu-comp-therm-008-case-8	0.9997	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
leu-comp-therm-008-case-11	1.0007	(2)	0.0000	(4)	0.0000	(4)	0.0003	(4)	coinc r/o
leu-sol-therm-002-case-1	0.9994	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
leu-sol-therm-002-case-2	0.9964	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	

Table 6b. VALIDATION_CRIT_EXPANDED – Linux – 2017-04-07

	610_12_71_lin		611_12_71_lin		621_17_71_lin		620_17_71_lin	
	keff	std	deltak	std	deltak	std	deltak	std
HEU Benchmarks								
heu-met-fast-001	0.9994	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
heu-met-fast-008	0.9962	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
heu-met-fast-018-case-2	0.9995	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
heu-met-fast-003-case-1	0.9949	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
heu-met-fast-003-case-2	0.9945	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
heu-met-fast-003-case-3	0.9989	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
heu-met-fast-003-case-4	0.9974	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
heu-met-fast-003-case-5	1.0012	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
heu-met-fast-003-case-6	1.0020	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
heu-met-fast-003-case-7	1.0019	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
heu-met-fast-028	1.0027	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
heu-met-fast-014	0.9977	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
heu-met-fast-003-case-8	1.0023	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
heu-met-fast-003-case-9	1.0023	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
heu-met-fast-003-case-10	1.0052	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
heu-met-fast-003-case-11	1.0094	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
heu-met-fast-003-case-12	1.0087	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
heu-met-fast-013	0.9975	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
heu-met-fast-021-case-2	0.9979	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
heu-met-fast-022-case-2	0.9976	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
heu-met-fast-012	0.9984	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
heu-met-fast-019-case-2	1.0069	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
heu-met-fast-009-case-2	0.9966	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
heu-met-fast-009-case-1	0.9977	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
heu-met-fast-011	0.9985	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
heu-met-fast-020-case-2	1.0006	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
heu-met-fast-004-case-1	1.0034	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
heu-met-fast-015	0.9947	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
heu-met-fast-026-case-c-11	1.0032	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
heu-comp-inter-003-case-6	0.9948	(4)	0.0000	(5)	0.0000	(5)	0.0000	(5)
heu-met-inter-006-case-1	0.9929	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
heu-met-inter-006-case-2	0.9968	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
heu-met-inter-006-case-3	1.0008	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
heu-met-inter-006-case-4	1.0072	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
u233-comp-therm-001-case-6	0.9988	(4)	0.0000	(5)	0.0000	(5)	0.0000	(5)
heu-sol-therm-013-case-1	0.9985	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
heu-sol-therm-013-case-2	0.9969	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
heu-sol-therm-013-case-3	0.9939	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
heu-sol-therm-013-case-4	0.9953	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
heu-sol-therm-032	0.9992	(2)	0.0000	(2)	0.0000	(2)	0.0000	(2)

Table 6c. VALIDATION_CRIT_EXPANDED – Linux – 2017-04-07

	610_12_71_lin		611_12_71_lin		621_17_71_lin		620_17_71_lin		
	keff	std	deltak	std	deltak	std	deltak	std	
Pu Benchmarks									
pu-met-fast-001	0.9993	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-met-fast-002	1.0003	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-met-fast-022-case-2	0.9984	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
mix-met-fast-001	0.9998	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
mix-met-fast-003	1.0004	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-met-fast-006	1.0001	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-met-fast-010	0.9996	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-met-fast-020	0.9983	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-met-fast-008-case-2	0.9977	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-met-fast-005	1.0019	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-met-fast-025-case-2	0.9991	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-met-fast-026-case-2	0.9987	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-met-fast-009	1.0048	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-met-fast-023-case-2	0.9994	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-met-fast-018	0.9993	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-met-fast-019	1.0004	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-met-fast-024-case-2	1.0025	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-met-fast-011	1.0000	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-met-fast-021-case-2	0.9935	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-met-fast-021-case-1	1.0047	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-met-fast-003-case-103	0.9990	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-comp-inter-001	1.0116	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
mix-comp-therm-002-case-pnl30	1.0002	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
mix-comp-therm-002-case-pnl31	1.0018	(4)	0.0000	(5)	0.0000	(5)	-0.0006	(5)*	coinc r/o
mix-comp-therm-002-case-pnl32	1.0020	(2)	0.0000	(4)	0.0000	(4)	-0.0003	(4)	coinc r/o
mix-comp-therm-002-case-pnl33	1.0063	(2)	0.0000	(4)	0.0000	(4)	-0.0001	(4)	coinc r/o
mix-comp-therm-002-case-pnl34	1.0045	(2)	0.0000	(4)	0.0000	(4)	0.0001	(4)	coinc r/o
mix-comp-therm-002-case-pnl35	1.0063	(2)	0.0000	(4)	0.0000	(4)	-0.0004	(4)	coinc r/o
pu-sol-therm-009-case-3a	1.0191	(2)	0.0000	(2)	0.0000	(2)	0.0000	(2)	
pu-sol-therm-011-case-16-5	1.0054	(4)	0.0000	(5)	0.0000	(5)	0.0000	(5)	
pu-sol-therm-011-case-18-1	0.9941	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-sol-therm-011-case-18-6	1.0005	(4)	0.0000	(5)	0.0000	(5)	0.0000	(5)	
pu-sol-therm-021-case-1	1.0053	(4)	0.0000	(5)	0.0000	(5)	0.0000	(5)	
pu-sol-therm-021-case-3	1.0043	(4)	0.0000	(5)	0.0000	(5)	0.0000	(5)	
pu-sol-therm-018-case-9	1.0026	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)	
pu-sol-therm-034-case-1	1.0007	(4)	0.0000	(5)	0.0000	(5)	0.0000	(5)	

Table 6d. VALIDATION_CRIT_EXPANDED – Linux – 2017-04-07

	610_12_71_lin		611_12_71_lin		621_17_71_lin		620_17_71_lin	
	keff	std	deltak	std	deltak	std	deltak	std
IEU Benchmarks								
ieu-met-fast-003-case-2	1.0028	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
ieu-met-fast-005-case-2	1.0024	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
ieu-met-fast-006-case-2	0.9958	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
ieu-met-fast-004-case-2	1.0075	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
ieu-met-fast-001-case-1	1.0009	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
ieu-met-fast-001-case-2	0.9999	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
ieu-met-fast-001-case-3	1.0011	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
ieu-met-fast-001-case-4	1.0015	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
ieu-met-fast-002	0.9991	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
ieu-met-fast-007-case-4	1.0045	(2)	0.0000	(2)	0.0000	(2)	0.0000	(2)
mix-met-fast-008-case-7	1.0192	(2)	0.0000	(2)	0.0000	(2)	0.0000	(2)
ieu-comp-therm-002-case-3	1.0038	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
leu-sol-therm-007-case-14	0.9947	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
leu-sol-therm-007-case-30	0.9971	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
leu-sol-therm-007-case-32	0.9959	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
leu-sol-therm-007-case-36	0.9990	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
leu-sol-therm-007-case-49	0.9972	(2)	0.0000	(4)	0.0000	(4)	0.0000	(4)
Wall-clock:	264.2 min		131.6 min		122.3 min		122.3 min	
Threads:	16		16		16		16	
Rel. Speed:	1.00		2.01		2.16		2.16	

Table 7. Validation Rossi Alpha Test Suite Results

	Benchmark		MCNP5 1.60 ENDF/B-VII.0		MCNP6.1 ENDF/B-VII.0		MCNP6.1 & MCNP6.1.1 ENDF/B-VII.1		MCNP6.2 ENDF/B-VII.1	
	rossi-α	std	rossi-α	std	rossi-α	std	rossi-α	std	rossi-α	std
U233 Benchmarks										
Jezebel-233	-100	(1)	-108	(1)	-108	(1)	-107	(1)	-107	(1)
Flattop-23	-26.7	(5)	-30.2	(4)	-30.2	(4)	-29.8	(4)	-29.8	(4)
HEU Benchmarks										
Godiva	-111	(2)	-113	(1)	-113	(1)	-113	(1)	-113	(1)
Flattop-25	-38.2	(2)	-39.7	(2)	-39.5	(2)	-39.6	(2)	-39.5	(2)
Zeus-1	-0.338	(7)	-0.363	(2)	-0.363	(2)	-0.360	(2)*	-0.360	(2)
Zeus-5	-14.8	(1)	-10.8	(1)	-10.8	(1)	-10.7	(1)	-10.7	(1)
Zeus-6	-3.73	(5)	-4.14	(3)	-4.16	(3)	-4.11	(3)*	-4.10	(3)
IEU Benchmarks										
BIG TEN	-11.7	(1)	-11.8	(1)	-11.8	(1)	-11.7	(1)	-11.7	(1)
STACY-30	-0.0127	(3)	-0.0133	(3)	-0.0133	(3)	-0.0121	(3)***	-0.0121	(3)
STACY-46	-0.0106	(4)	-0.0104	(2)	-0.0104	(2)	-0.0106	(2)	-0.0106	(2)
Pu Benchmarks										
Jezebel	-64.0	(10)	-65	(1)	-65.1	(8)	-63.2	(7)**	-63.2	(7)
Flattop-Pu	-21.4	(5)	-21.0	(3)	-21.0	(3)	-20.2	(3)**	-20.2	(3)
THOR	-19.7	(10)	-20	(1)	-19.7	(7)	-20.6	(7)*	-20.6	(7)

Notes
- All results in 10^4 generations/second
- Color indicates type of diff, * indicates magnitude of diff:

compiler/hardware	* = diff > 1 std
nuclear data	** = diff > 2 std
minor k-adjoint bug fix	*** = diff > 3 std

Table 8. Validation Suite Summary

VALIDATION_CRITICALITY – 31 ICSBEP cases

	Mac OS X match	Mac OS X Sab-fix	Mac OS X coinc r/o	Mac OS X compiler	Linux match	Linux Sab-fix	Linux coinc r/o
mcnp6.1 (ref)							
mcnp6.1.1	30	1	-	-	30	1	-
mcnp6.2 old coinc sur	29	1	-	1	30	1	-
mcnp6.2 new coinc sur	27	1	2	1	28	1	2

VALIDATION_CRIT_EXPANDED – 119 ICSBEP cases

	Mac OS X – 12 cores match	Mac OS X – 12 cores coinc r/o	Mac OS X – 12 cores speedup	Linux – 16 cores match	Linux – 16 cores coinc r/o	Linux – 16 cores speedup
mcnp6.1 (ref)						
mcnp6.1.1	119	-	1.9	119	-	2.0
mcnp6.2 old coinc sur	119	-	1.9	119	-	2.2
mcnp6.2 new coinc sur	108	11	1.9	108	11	2.2