LA-UR-13-27979

Approved for public release; distribution is unlimited.

Title: Monte Carlo Application ToolKit (MCATK)

Author(s): Adams, Terry R.
Nolen, Steven D.
Sweezy, Jeremy Ed
Zukaitis, Anthony J.
Campbell, Joann M.
Goorley, John T.
Greene, Simon
Aulwes, Robbie T.

Intended for: arXiv
Joint International Conference in Supercomputing on Nuclear
Applications and Monte Carlo 2013 (SNA&MC 2013).,
2013-10-27/2013-10-31 (Paris, France)
Report

Issued: 2014-10-07 (rev.2)

Disclaimer:

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 approving this article, the publisher recognizes that the U.S. Government retains 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.

Monte Carlo Application ToolKit (MCATK)

Terry Adams*, Steve Nolen, Jeremy Sweezy, Anthony Zukaitis,
Joann Campbell, Tim Goorley, Simon Greene¹, and Rob Aulwes

Los Alamos National Laboratory, P.O. Box 1663, Los Alamos, New Mexico 87545

¹Design Physics Department, AWE Aldermaston, Berkshire RG7 4PR, U.K.

*Corresponding Author, E-mail: tadams@lanl.gov

Abstract

The Monte Carlo Application ToolKit (MCATK) is a component-based software library designed to build specialized applications and to provide new functionality for existing general purpose Monte Carlo radiation transport codes. We will describe MCATK and its capabilities along with presenting some verification and validations results.

KEYWORDS: *Monte Carlo Particle Transport, Component Software, Agile Development, Parallel Computing, Time-Dependent, Population Control, Mesh Based, MCATK*

I. Introduction

The Monte Carlo Application ToolKit (MCATK) is a C++ component-based software library developed at Los Alamos National Laboratory (LANL). It is designed to build specialized applications and to provide new functionality for existing general purpose Monte Carlo radiation transport codes, such as MCNP⁽¹⁾. The library is developed using Agile⁽²⁾ software engineering methodologies. Over the past five years, the primary focus of feature support has been for Monte Carlo neutron transport.

Monte Carlo Background

The Monte Carlo (MC) technique⁽³⁾ uses statistical methods that leverage the use of random events to sample for numerical results. A few application examples are numerical integration, fluid dynamics, quantum chemistry, and particle transport in nuclear reactors. In our case, MCATK uses MC to model neutron transport. At LANL, MCATK follows a long history of computational MC that was started at Los Alamos at the end of War World II.

The application of statistical methods in the physical sciences really came into its own right with the advent of computers in the 1940s. The initial proposal was made by Stan Ulam and John von Neumann^(4,5,6). According to Ulam⁽⁴⁾,

“It occurred to me then that this could be equally true of all processes involving branching of events, as in the production of

further multiplication of neutrons in some kind of material containing uranium or other fissile elements. At each stage of the process, there are many possibilities determining the fate of the neutron.”

In 1946, Ulam described this to von Neumann^(4,5,6) and they started planning an actual calculation. Much of this detailed planning is described in the 1947 correspondence between Robert Richtmyer and John von Neumann⁽⁶⁾. An early and interesting general overview of the MC method for various mathematical physics was given by Nick Metropolis and Stan Ulam in 1949⁽⁷⁾. Since that ground breaking work at Los Alamos in the 1940s, the MC method has had a continuous presence at the national laboratory up to the present computer code, MCNP⁽¹⁾, whose lineage can be traced directly to the efforts described above.

In particle transport there are basically two solution methods of choice. One solving the transport problem using integro-differential equations (ie, the Boltzmann transport equation)⁽⁸⁾ that models the system of choice and is often called deterministic transport, and the second method is the MC transport^(9,10) method which uses statistical processes to model the system.

Briefly, we will mention several characteristics⁽⁸⁾ when considering between Monte Carlo and deterministic transport. One advantage MC brings over deterministic is that MC can point-wise sample the space-angle-energy-time phase space where deterministic method must be discretized and can introduce discretization errors. Unlike deterministic methods, MC is extended easily into complicated

three-dimensional configurations. Due to this fact, MC often brings computer savings in memory, time, and accuracy. Another reason to consider MC is due to its natural ability to model particle sources in greater detail for the phase space.

A couple of disadvantages MC as compared to deterministic methods are that the statistical uncertainty can grow rapidly to unacceptable levels in some instances and MC, while nice for localized solutions, often has difficulties in obtaining global solutions.

Brief MCATK History

The MCATK project was initiated in response to a LANL white paper by Webster⁽¹¹⁾ that found several issues facing the LANL code development strategy. Some of the issues included redundant code packages, high development & maintenance costs, and loss of senior staff.

To address some of these issues, MCATK software development started in May 2008 with a team of five scientific developers, using Agile software practices, targeted a very basic serial neutron eigenvalue solver. And at present, we have parallel, mesh-based, time-independent and time-dependent continuous energy neutron transport solver applications.

II. Software Development Environment

The MCATK team uses an Agile methodology of Extreme Programming (XP) as prescribed by Shores and Warden⁽¹²⁾. Several of the specific practices and processes used are team colocation, informative workspace, the planning game, pair-programming, test-driven development, daily stand-up meetings, two-week iterations, root-cause analysis, retrospectives, continuous integration, and ten-minute builds.

The team uses local development servers to help speed up building and testing feedback for their software development and testing. Using C++^(13,14), the Boost library⁽¹⁵⁾, and the unit-test framework, UnitTest++⁽¹⁶⁾, code dependencies are more readily “broken” to develop and test a component-based library. The build and test system is supported by Kitware open source software applications CMake, CTest, and CDash⁽¹⁷⁾.

The majority of the development depends on the unit-test framework not only for unit-testing but also for focused-integrated and nightly regression testing. While unit-tests are meant to test single concepts like an object or function, the focused-integrated are meant to test the combination of two or three objects or functions. These types of tests are intended to be executed quickly and

setup easily with *every* code change and build during the development phase. This practice provides fast feedback to the developer. The nightly tests are run once a day and are meant to test multiple objects combined such as in actual applications. Note that all testing includes testing of the parallel models as described in the Geometry section III.1, later.

As the supported applications became more mature, the team developed a stand-alone driver application that can be setup by a MCATK developed Qt⁽¹⁸⁾ based GUI. Also we developed a handle⁽¹⁹⁾ based “Flat” Application Programming Interface (FlatAPI) making the MCATK library callable by C and Fortran applications.

III. Components

Most of the software components were developed for Monte Carlo radiation transport purposes with an emphasis on neutron transport. The presentation of the MCATK component capabilities is organized in three sections: supported geometries, physics, and tallies as illustrated in Figure 1.

Figure 1. Conceptual diagram illustrating one way a Monte Carlo transport code can be compartmentalized. We have organized the paper in this fashion.

1. Geometries

MCATK uses mesh-based geometries and supports several coordinates systems and dimensions as shown in Table I. All geometries support transmissive, reflective, and periodic boundary conditions where appropriate. The geometry information can be supplied through arrays passed to the MCATK library or in the form of a PARTISN LNK3DNT⁽²⁰⁾ file.

Table I. The MCATK supported mesh geometries in coordinate systems⁽²¹⁾ and dimensions.

	1D	2D	3D
Spherical	R	-	-
Cylindrical	R	RZ	RZθ
Cartesian	X	XY	XYZ

An important point to understand is that MCATK actually transports (moves) particles and does the collision physics in Cartesian-XYZ (3D) space while the material lookup and tallying of properties (eg., track-length estimators) are done in the chosen mesh geometry. For example, in Figure 2, the spherical-R (1D) mesh can be considered basically a series of concentric spherical shells whose inner and outer boundaries are defined by the radial vertices. The particle moves in 3D Cartesian space but keeps track of which shell (by mesh boundaries) it is located in for material lookup and tallying purposes.

Figure 2. Shows a particle moving in 3D Cartesian-XYZ space through a 1D spherical-R mesh.

Similarly, cylindrical-RZ is made up of stacked (in the Z-direction) concentric (in the R-direction) rings. MCATK also supports a limited set of combinatorial solid geometries (CSG). This includes the ability to embed a mesh geometry inside a solid geometry. An example, is a spherical-R mesh embedded inside a sphere geometry.

The MCATK parallel paradigm is Message Passing Interface (MPI)⁽²²⁻²⁵⁾ for mesh domain replication, domain decomposition, and a domain hybrid of each. The *domain replicated* (often referred to as particle decomposed) model simply uses a copy of the full geometry domain on each process and decomposes the particle population across processes to reduce the particle tracking work. This model is often said to be “embarrassing parallel.” If one has the computational memory necessary for this model and a given problem, then replicated usually is the most efficient MPI parallel treatment for a MC neutral particle transport code. At the other extreme, if one can no longer afford to domain replicate the geometry due to its memory footprint, then domain decomposition is an option. In *domain decomposition*, one decomposes the geometry into subdomains where there is one subdomain per process. Then particles are tracked across and between subdomains. Generally, this parallel model of particle transport does not scale well and use of domain decomposition is discouraged when domain replicated can be applied.

If one needs to domain decompose their problem, it is best to decompose the domain into as few subdomains as possible and replicate the less decomposed geometries as seen in Figure 3. This is a *hybrid model* between the two extreme models of full replicated and decomposed. With this you only decompose the domain as necessary for your problem.

At this time in MCATK, though our design supports it, we do not have an algorithm for balancing the work load to guide our decomposition of the geometry.

For the domain decomposed model, MCATK at present uses synchronous particle communication for moving particles between domains. There are asynchronous parallel communication models by Brunner, *et al.*^(26,27) which may be implemented in the future.

Figure 3. A hybrid model example, that starts with an original 4x4 mesh on the left. Then using four processes, shown on the right, we have two replicas that have a two subdomain decomposition of the original mesh on each replica.

The MCATK parallel models include the library’s ability to read in a single LNK3DNT file using the domain information on each processor (eg., for domain decomposed the sub-domain information needed is read from the file per processor.) Note that the MCATK MPI parallel models are similar to another LANL Monte Carlo code, MC++⁽²⁸⁾, except that MC++ uses asynchronous particle communication in the domain decomposed model.

2. Physics

MCATK reads continuous-energy neutron cross-section information from ACE formatted files generated from NJOY⁽²⁹⁾. This is the same cross-section format read by MCNP. At the current time, MCATK only transports neutrons. The energy range of the neutrons is limited by the energy range of the neutron cross-section tables which is approximately 0-20 MeV. Presently, MCATK *DOES NOT USE* the continuous energy unresolved resonance range probability tables⁽³⁰⁾ when sampling for neutron cross-sections. This means the resonances in the unresolved energy range are not treated properly and thus any energy self-shielding effects⁽³¹⁾ in this unresolved region are not currently modeled.

Other neutron physics capabilities in MCATK include thermal treatments for free-gas scattering⁽³²⁾ and Doppler broadening of the elastic scattering cross-section⁽³³⁾.

Free Gas Model (Scattering and Doppler Broadening)

In some regimes, the kinematics of a neutron collision with a

nucleus is strongly affected by the thermal motion of the nucleus. If the target nucleus is in motion, a colliding neutron may gain energy by up-scattering or lose energy by down-scattering. The assumption is that the neutrons are scattering off unbound nuclei (free gas) that are distributed energy-wise in a Maxwell-Boltzmann distribution. The output from this scattering correction is the exiting relative neutron energy and direction⁽³²⁾ which will be used later for scattering selection.

MCATK at this time *ignores* molecular and crystalline effects where the bound nucleus cannot recoil freely and the interference effects that can arise due to the small de Broglie wavelength of the neutron relative to the inter-nuclear separation of the atoms. These molecular and crystalline effects could be accounted for with the use of thermal $S(\alpha, \beta)$ tables⁽³⁴⁾ in the future.

MCATK performs a free-gas Doppler (thermal) broadening⁽³⁵⁾ on the neutron scattering cross section as a function of cell material temperature⁽³³⁾. The actual code flow of how MCATK incorporates these thermal corrections into the reaction sampling scheme is described by Sweezy, *et al.*⁽³⁶⁾.

The library provides support for several source models. Both a simple burst point source along with support for time, space, and energy varying sources. A spontaneous fission source is presently being added to the library.

Algorithms

The MCATK libraries has several applications that can be run from the Qt based GUI. MCATK algorithms include static K_{eff} eigenvalues⁽³⁷⁾ using a power iteration method, and static neutron population time constants⁽³⁷⁾ using a search method similar to the Monte Carlo eigenvalue methods described by Brockway, *et al.*⁽³⁸⁾.

In addition to a static-eigenvalue capability, MCATK can calculate dynamic neutron population time constants⁽³⁹⁾ using population control⁽⁴⁰⁾. Many codes, like MCNP, follow each particle through a complete history adjusting the particle's simulation time at each event to represent its progression in time. In contrast, MCATK can treat the time boundary as a simulation boundary accruing the contribution from the entire population before proceeding.

MCATK provides the means for analyzing time varying problems. These include problems with evolving material properties or dimensions, tallies requiring non-static weight windows, and any simulation that could benefit from information that is itself a product of the simulation. Unlike other approaches, this technique allows continuous feedback from such things as the particle population which is then available for fine tuning everything from weight window parameters to problem configuration. This, in turn, allows the user to set up and run a problem skipping previously required knowledge like the system's proximity to criticality.

Particle population control methods are often required in both dynamic and static Monte Carlo methods. If no population control is performed the particle population can die out or grow exponentially. Although particle population methods are commonly used in dynamic Monte Carlo methods, little literature is available that describe the methods that are used. The MCATK particle population control method⁽⁴⁰⁾ has been derived using basic splitting and Russian Roulette techniques⁽⁹⁾ and easily uses message passing parallel communication.

3. Tallies

At this time there are only a few tallies in MCATK. Currently there are track-length estimators for K_{eff} ⁽³⁷⁾, the static neutron population time constant⁽³⁷⁾, and scalar fluxes. There are two dynamic neutron population time constant estimators⁽³⁹⁾: a track-length estimator and a collision estimator. There are also material region population tallies for static and dynamic calculations, along with a chain-based neutron multiplication tally.

IV. Verification and Validation Tests

The MCATK library's time-independent and time-dependent solvers are applied to Verification and Validation tests by Sood^(42,43) and Sweezy, *et al.*⁽⁴⁴⁾.

1. The Analytic Benchmark Verification Tests

The analytic benchmark tests⁽⁴²⁾ are a useful step to isolate and verify algorithm execution independent of the neutron data. These benchmarks can test things like particle tracking through the geometries in serial and parallel, the boundary conditions, the static eigenvalues and dynamic population time constant algorithms, and the static and dynamic flux tallies.

MCATK has setups for at least 38 of the analytic benchmark tests. Each test is a one-energy isotropic scattering problem with single or multiple materials. The problem names are defined directly out of Sood, *et al.*⁽⁴²⁾ with one extension. This extension designates an infinite medium problem with the notation (IN). Only three mesh types are used and are designated by spherical-R (SP), cylindrical-RZ (CY), and Cartesian-XYZ (SL).

All test results and statistics presented are based on 1,000 experiments per problem.

Time-Independent Eigenvalue: K_{eff}

Setups for 38 of the analytic benchmark tests⁽⁴²⁾ have been executed using the K_{eff} algorithm in MCATK and the results are shown in Tables II(a,b). Table IIa presents results for each problem's exact and simulated eigenvalue average over 1,000 experiments. Each K_{eff} eigenvalue experiment used 5,000 Monte Carlo particles per generation.

Accompanying the eigenvalue results of Table IIa, is a statistical analysis of the 1,000 K_{eff} experiments which are

shown in Table IIb. For each analytic test problem the normal distribution percentages of the 1,000 experiments are binned based in one, two, and three-sigmas (σ). Also included are the plus (“+”) and minus (“-”) columns that are the percentage of experiments in which the mean eigenvalue is above(+) or below(-) the exact eigenvalue. The exact normal distribution percentages are shown at the bottom of Table IIb for comparison.

Table IIa. MCATK Analytic Verification Suite⁽⁴²⁾ K_{eff} results for one-energy group isotopic scattering. Averages based on 1,000 experiments for each problem.*

Problem Name	Exact	Mean	Std. Dev.	# σ
PUa-1-0-SL	1.000000	1.000008	0.000032	≤ 1
PUa-1-0-SL-IN	2.612903	2.612892	0.000084	≤ 1
PUa-H2O(1)-1-0-SL	1.000000	1.000027	0.000032	≤ 1
PUa-H2O(0.5)-1-0-SL	1.000000	1.000024	0.000033	≤ 1
PUB-1-0-CY	1.000000	1.000021	0.000030	≤ 1
PUB-1-0-CY-IN	2.290323	2.290209	0.000077	≤ 2
PUB-1-0-SL	1.000000	1.000000	0.000030	≤ 1
PUB-1-0-SL-IN	2.290323	2.290365	0.000074	≤ 1
PUB-1-0-SP	1.000000	0.999978	0.000031	≤ 1
PUB-1-0-SP-IN	2.290323	2.290376	0.000073	≤ 1
PUB-H2O(10)-1-0-CY	1.000000	1.000049	0.000030	≤ 2
PUB-H2O(1)-1-0-CY	1.000000	0.999970	0.000029	≤ 2
Ua-1-0-CY	1.000000	0.999990	0.000031	≤ 1
Ua-1-0-CY-IN	2.250000	2.249985	0.000071	≤ 1
Ua-1-0-SL	1.000000	0.999998	0.000030	≤ 1
Ua-1-0-SL-IN	2.250000	2.250141	0.000070	≤ 3
Ua-1-0-SP	1.000000	1.000020	0.000031	≤ 1
Ua-1-0-SP-IN	2.250000	2.249979	0.000072	≤ 1
Ub-1-0-SP-IN	2.330917	2.330956	0.000075	≤ 1
Ub-H2O(1)-1-0-SP	1.000000	0.999917	0.000031	≤ 3
Uc-1-0-SP-IN	2.256083	2.256065	0.000070	≤ 1
Uc-H2O(2)-1-0-SP	1.000000	1.000022	0.000030	≤ 1
Ud-1-0-SP-IN	2.232667	2.232813	0.000066	≤ 3
Ud-H2O(3)-1-0-SP	1.000000	1.000010	0.000031	≤ 1
UD2O-1-0-CY	1.000000	0.999945	0.000034	≤ 2
UD2O-1-0-CY-IN	1.133333	1.133362	0.000035	≤ 1
UD2O-1-0-SL	1.000000	0.999988	0.000031	≤ 1
UD2O-1-0-SL-IN	1.133333	1.133268	0.000035	≤ 2
UD2O-1-0-SP	1.000000	1.000030	0.000035	≤ 1
UD2O-1-0-SP-IN	1.133333	1.133413	0.000036	≤ 3
UD2O-H2O(10)-1-0-CY	1.000000	0.999961	0.000032	≤ 2
UD2O-H2O(10)-1-0-SL	1.000000	0.999974	0.000032	≤ 1
UD2O-H2O(1)-1-0-CY	1.000000	1.000012	0.000033	≤ 1
UD2O-H2O(1)-1-0-SL	1.000000	0.999953	0.000032	≤ 2
Ue-1-0-CY-IN	2.180667	2.180728	0.000066	≤ 1
Ue-1-0-SL-IN	2.180667	2.180756	0.000069	≤ 2
Ue-1-0-SP-IN	2.180667	2.180599	0.000068	≤ 1
Ue-Fe-Na-1-0-SL	1.000000	0.999965	0.000031	≤ 2

* Note that problem name with “IN” stand for infinite medium.

Table IIb. MCATK Analytic Verification Suite⁽⁴²⁾ K_{eff} statistical analysis. Based on 1,000 experiments for each problem. A continuation of Table IIa results. *

Problem Name	$\leq 1\sigma$	$\leq 2\sigma$	$\leq 3\sigma$	$> 3\sigma$	"+"	"-"
	%	%	%	%	%	%
PUa-1-0-SL	65.8	28.6	5.3	0.3	50.3	49.7
PUa-1-0-SL-IN	67.1	27.0	5.7	0.2	49.7	50.3
PUa-H2O(1)-1-0-SL	66.5	27.3	5.8	0.4	51.5	48.5
PUa-H2O(0.5)-1-0-SL	67.2	27.9	4.3	0.6	53.4	46.6
PUB-1-0-CY	67.6	27.0	4.9	0.5	51.6	48.4
PUB-1-0-CY-IN	66.4	26.4	6.8	0.4	49.4	50.6
PUB-1-0-SL	69.7	25.6	4.7	0.0	51.1	48.9
PUB-1-0-SL-IN	67.1	28.1	4.0	0.8	51.9	48.1
PUB-1-0-SP	63.5	29.4	6.7	0.4	50.3	49.7
PUB-1-0-SP-IN	67.6	27.6	4.4	0.4	52.3	47.7
PUB-H2O(10)-1-0-CY	67.2	28.0	4.6	0.2	51.3	48.7
PUB-H2O(1)-1-0-CY	68.8	27.3	3.7	0.2	49.3	50.7
Ua-1-0-CY	65.5	28.7	5.6	0.2	49.6	50.4
Ua-1-0-CY-IN	68.5	27.7	3.4	0.4	49.7	50.3
Ua-1-0-SL	67.5	27.8	4.4	0.3	49.1	50.9
Ua-1-0-SL-IN	70.2	24.8	4.5	0.5	51.5	48.5
Ua-1-0-SP	67.7	25.2	6.3	0.8	50.2	49.8
Ua-1-0-SP-IN	68.6	26.0	4.8	0.6	49.0	51.0
Ub-1-0-SP-IN	67.0	28.2	4.1	0.7	50.8	49.2
Ub-H2O(1)-1-0-SP	67.9	26.7	4.7	0.7	47.2	52.8
Uc-1-0-SP-IN	67.4	28.7	3.4	0.5	49.8	50.2
Uc-H2O(2)-1-0-SP	66.7	30.0	3.0	0.3	50.4	49.6
Ud-1-0-SP-IN	71.2	25.5	3.3	0.0	53.5	46.5
Ud-H2O(3)-1-0-SP	66.2	28.6	4.7	0.5	51.0	49.0
UD2O-1-0-CY	64.7	28.4	6.3	0.6	48.4	51.6
UD2O-1-0-CY-IN	68.9	27.9	3.0	0.2	51.9	48.1
UD2O-1-0-SL	68.1	27.0	4.8	0.1	48.7	51.3
UD2O-1-0-SL-IN	69.3	25.8	4.7	0.2	47.3	52.7
UD2O-1-0-SP	64.0	27.7	7.5	0.8	51.0	49.0
UD2O-1-0-SP-IN	68.3	26.3	4.8	0.6	52.7	47.3
UD2O-H2O(10)-1-0-CY	68.0	27.5	4.0	0.5	46.4	53.6
UD2O-H2O(10)-1-0-SL	68.4	26.3	4.8	0.5	49.8	50.2
UD2O-H2O(1)-1-0-CY	66.0	28.3	5.3	0.4	49.7	50.3
UD2O-H2O(1)-1-0-SL	68.3	27.0	4.3	0.4	48.8	51.2
Ue-1-0-CY-IN	69.7	26.3	3.8	0.2	51.3	48.7
Ue-1-0-SL-IN	68.2	26.9	4.7	0.2	52.9	47.1
Ue-1-0-SP-IN	68.5	27.1	4.3	0.1	49.8	50.2
Ue-Fe-Na-1-0-SL	65.0	29.8	4.9	0.3	49.2	50.8
	$\leq 1\sigma$	$\leq 2\sigma$	$\leq 3\sigma$	$> 3\sigma$	"+"	"-"
Exact	68.27	27.18	4.28	0.27	50.0	50.0
Averages (over 38 exp.)	67.48	27.38	4.74	0.39	52.6 [§]	47.4 [§]
Stand. Err. (over 38 exp.)	0.27	0.20	0.17	0.04		

* Note that problem name with “IN” stand for infinite medium.
[§] This is the percentage, of the 38 problems, that are above (“+”) or below (“-”) the exact values; and not an average over the 38 problem percentages in the (“+”)/ (“-”) columns.

All K_{eff} results in Table IIa are within 3σ . All the problem analysis for Table IIb, looks to be reasonable in comparison to the exact percentages at the bottom of the table.

Time-Independent Eigenvalue: Population Time Constants (α)

In an earlier report on analytic benchmarks by Sood, *et al.*⁽⁴³⁾, they pointed out that the time-independent (*eg.*, static alpha eigenvalue, α) and time-dependent neutron population constant (*eg.*, dynamic alpha, α) could be examined by modifying the critical system's capture and total cross sections by α/v to represent subcritical and supercritical systems, where v is the neutron speed.

In Tables III(a,b), we present a few static alpha results preformed by MCATK. As with the K_{eff} tables, Table IIIa is a comparison of the exact alpha eigenvalue to the simulated mean eigenvalue for 1,000 experiments. Each experiment is using 10,000 particles per generation. Table IIIb presents for each analytic test problem the normal distribution percentages of the 1,000 experiments as binned based on one, two, and three-sigmas(σ) distributions.

The three problem shown in Tables III(a,b) are within statistical reason and look good within the normal distribution.

Table IIIa. MCATK Analytic Verification Suite⁽⁴²⁾ static alpha (α) eigenvalue results for one-energy group isotopic scattering. Averages based on 1,000 experiments for each problem.

Problem Name	Exact	Mean	Std. Dev.	# σ
	[gen/sh]	[gen/sh]	[gen/sh]	
PUa-H2O(0.5)-1-0-SL	0.2	0.200092	0.000196	≤ 1
PUB-1-0-SP	0.1	0.100182	0.000239	≤ 1
UD2O-H2O(10)-1-0-CY	0.2	0.199955	0.000070	≤ 1

Table IIIb. MCATK Analytic Verification Suite⁽⁴²⁾ static alpha (α) eigenvalue statistics. Based on 1,000 experiments for each problem. Percentages compared to a normal distribution.

Problem Name	$\leq 1\sigma$	$\leq 2\sigma$	$\leq 3\sigma$	$> 3\sigma$	"+"	"-"
	%	%	%	%	%	%
Exact	68.27	27.18	4.28	0.27	50.0	50.0
PUa-H2O(0.5)-1-0-SL	68.5	27.2	4.0	0.3	49.2	50.8
PUB-1-0-SP	65.9	28.0	5.1	1.0	50.5	49.5
UD2O-H2O(10)-1-0-CY	70.3	26.0	3.5	0.2	49.6	50.4

Time-Dependent: Population Time Constants (α)

Using one of same analytic benchmark setups [PUa-H2O(0.5)-1-0-SL] from the eigenvalue simulations, we performed several time-dependent population time constants (α) simulations⁽⁴³⁾ using MCATK. Tables IV(a,b) show

similar outputs as in Tables III. In this case, we used the problem setup but with three different exact α targets ($\alpha = -0.1, 0.0, 0.2$). Each experiment used 100,000 Monte Carlo particles per time-step. Note the need for a larger number of MC particles compared to the time-independent alpha. This is because the time-independent alpha procedure is more efficient and is capable of producing more accurate results according to Brockway, *et al.*⁽³⁸⁾.

The three alphas shown in Tables IV, look to be statistically reasonable distributions. It is interesting to note in Table IVa all three mean α are below the exact α . And in Table IVb, all three "+/-" columns show more than 50% of the 1000 experiments are below, "-", the exact mean. This suggests the possibility of biasing in the time-dependent algorithm which is mentioned in our population control paper^(40,41). Until further studies of other benchmark problem simulations for dynamic α , then it would be premature to conclusively state this limited set of runs reflect this observed bias.

Table IVa. MCATK Analytic Verification Suite⁽⁴²⁾

dynamic alpha (α) results for one-energy group isotopic scattering. Averages based on 1,000 experiments for each problem.

Problem Name	Exact	Mean	Std. Dev.	# σ
	[gen/sh]	[gen/sh]	[gen/sh]	
PUa-H2O(0.5)-1-0-SL	-0.1	-0.100037	0.000020	≤ 2
PUa-H2O(0.5)-1-0-SL	0.0	-0.000055	0.000021	≤ 3
PUa-H2O(0.5)-1-0-SL	0.2	0.199992	0.000020	≤ 1

Table IVb. MCATK Analytic Verification Suite⁽⁴²⁾

dynamic alpha (α) statistics. Based on 1,000 experiments for each problem. Percentages compared to a normal distribution.

Problem Name	$\leq 1\sigma$	$\leq 2\sigma$	$\leq 3\sigma$	$> 3\sigma$	"+"	"-"
	%	%	%	%	%	%
Exact	68.27	27.18	4.28	0.27	50.0	50.0
PUa-H2O(0.5)-1-0-SL ¹	69.9	26.9	2.8	0.4	46.9	53.1
PUa-H2O(0.5)-1-0-SL ²	68.0	26.0	5.4	0.6	46.6	53.4
PUa-H2O(0.5)-1-0-SL ³	69.1	25.4	5.1	0.4	48.3	51.7

¹ $\alpha = -0.1$, ² $\alpha = 0.0$, ³ $\alpha = 0.2$

2. The Criticality Validation Suite

We have reported elsewhere a MCATK benchmark study for a Criticality Verification and Validation Suite by Sweezy, *et al.*⁽⁴⁴⁾. We present here only the validation results from this previous study. The MCATK K_{eff} estimates relative to the experimental benchmark values are presented in Table V. Also listed in Table V is the number of standard deviations that the differences fall from zero.

In Table VI is the number of problems with standard deviations less than 1, 2, and 3 standard deviations is summed and presented along with the expected number of matches at the 1, 2, and 3 standard deviation level. Only one problem, that does not require thermal $S(\alpha, \beta)$ tables, does not fall within a three standard deviation confidence interval, case ZEBR8H. The ZEBR8H problem difference is expected since a similar difference has been shown between MCNP6 and the experimental value⁽⁴⁵⁾. All other problems that exceed a three standard deviation confidence interval require thermal $S(\alpha, \beta)$ tables.

Table V. MCATK Criticality Validation Suite results relative to the experimental benchmark value.

	Experiment		MCATK		
Case	K_{eff}	Std Dev	Delta K_{eff}	Std Dev	$\#\sigma$
U233 Benchmarks					
JEZ233	1.0000	(10)	-0.0015	(13)	<1
FLAT23	1.0000	(14)	0.0005	(18)	<1
UMF5C2	1.0000	(30)	-0.0075	(32)	<3
FLSTF1*	1.0000	(83)	-0.0144	(85)	<2
ORNL11*	1.0006	(29)	-0.0098	(31)	>3
HEU Benchmarks					
Godiva	1.0000	(10)	0.0001	(16)	<1
FLAT25	1.0000	(30)	0.0018	(32)	<1
GodivaR*	0.9985	(11)	0.0221	(18)	>3
UH3C6	1.0000	(47)	-0.0039	(49)	<1
ZEUS2	0.9997	(8)	0.0000	(15)	<1
ORNL10*	1.0015	(26)	-0.0079	(29)	<3
IEU Benchmarks					
IMF03	1.0000	(17)	0.0032	(20)	<2
BIGTEN	0.9948	(13)	0.0001	(16)	<1
IMF04*	1.0000	(30)	0.0101	(32)	>3
ZEBR8H**	1.0300	(25)	-0.0192	(26)	>3
STACY36*	0.9988	(13)	-0.0131	(19)	>3
LEU Benchmarks					
LST2C2*	1.0024	(37)	-0.0233	(40)	>3
Pu Benchmarks					
JEZPU	1.0000	(20)	-0.0001	(22)	<1
JEZ240	1.0000	(20)	-0.0001	(22)	<1
FLATPU	1.0000	(30)	0.0014	(32)	<1
THOR	1.0000	(6)	-0.0010	(10)	<1
PUSH2O*	1.0000	(10)	0.0126	(16)	>3
HISHPG	1.0000	(110)	0.0103	(111)	<1
PNL2*	1.0000	(65)	0.0124	(67)	<2

* denotes systems that require $S(\alpha, \beta)$ thermal treatment.

** ZEBR8H is not matched by MCNP6 or MCATK indicating problems with the nuclear data or experimental error.

Table VI. Statistical analysis for Table V of the MCATK & experimental results for cases that do not require $S(\alpha, \beta)$ thermal tables.

	Difference MCATK to Exp.	Normal Distribution*
Number of Standard Deviations	Number of Cases	Expected Number of Cases
$\sigma < 1$	12	10
$1 < \sigma < 2$	1	4
$2 < \sigma < 3$	1	1
$\sigma > 3$	1	0
Total	15	

V. Summary

We have described the Monte Carlo Application ToolKit (MCATK) and its capabilities. We have presented several MCATK neutron transport results for a large set of verification and validation benchmarks.

During MCATK's short life-time, there also has been two research & develop (R&D) efforts associated with it. One was initial work for fault-tolerance in MCATK which eventually lead to contributions in the OpenMPI⁽²³⁾ fault-tolerant capabilities^(46,47). And the other R&D project was on MC photon transport on advanced architectures⁽⁴⁸⁾.

Near-term future features planned for the MCATK library include photon transport, variance reduction, and extending our sources & tally capabilities.

VI. Acknowledgements

The authors wish to thank Tony Hasenack, Jon Dahl, Francisco Souto, David Daniel, Sam Gutierrez, Jim Hill, Art Forster, Forest Brown, Avneet Sood, Bob Webster, Sriram Swaminarayan, Tom Booth, Rich Holmes, Pete Jaegers, Stephanie Frankle, Bob Little, Mike Burkett, Paul Henning, Bill Archer, Mary Beth Lee, Mark Gray, Kent Parsons, and Jeremy Conlin for their time, help, suggestions, and support for MCATK. We also want to credit Forest Brown for kindly letting us use the Figure 1 diagram which is originally found in his lecture notes on Monte Carlo.

References

- 1) T. Goorley, *et al.*, "Initial MCNP6 Release Overview", Nuclear Technology, **180**, 298-315 (2012).
- 2) K. Beck, *et al.*, "Manifesto for Agile Software Development," Agile Alliance (2001), Web 9 Dec 2012, <<http://agilemanifesto.org/>>.
- 3) M. H. Kalos and P. A. Whitlock, Monte Carlo Methods Volume I: Basics, John Wiley & Sons, Inc., New York, NY (1986).
- 4) S. M. Ulam, *Adventures of a Mathematician*, Charles Scribner's Sons, New York, NY (1976).
- 5) Nicholas C. Metropolis, OH 135. Oral history interview by William Aspray, 29 May 1987, Los Alamos, New Mexico, Charles Babbage Institute, University of Minnesota, Minneapolis. <<http://purl.umn.edu/107493>>.
- 6) R. D. Richtmyer and J. von Neumann, "Statistical Methods In Neutron Diffusion," LAMS-551, Los Alamos National Laboratory, Los Alamos, NM (1947).
- 7) N. Metropolis and S. Ulam, "The Monte Carlo Method," J. Am. Statist. Assoc., **44**(247), 335-341 (1949).
- 8) E. E. Lewis and W. F. Miller, *Computational Methods of Neutron Transport*, American Nuclear Society, Inc., La Grange Park, Illinois (1993).
- 9) I. Lux and L. Koblinger, *Monte Carlo Particle Transport Methods: Neutron and Photon Calculations*, CRC Press, Inc., Boca Raton, FL (1991).
- 10) S. A. Dupree and S. K. Fraley, *A Monte Carlo Primer: A Practical Approach to Radiation Transport*, Kluwer Academic/Plenum Publishers, New York, NY (2002).
- 11) R. Webster, *et al.*, "Technical White Paper: The LANL Code Development Strategy," LA-UR-08-0174, Los Alamos National Laboratory, Los Alamos, NM (2008).
- 12) J. Shore and S. Warden, *The Art Of Agile Development*, O'Reilly Media, Inc., Sebastopol, CA (2007).
- 13) MCATK's primary development compiler is GNU's g++.
- 14) MCATK's secondary compiler is Intel's icpc.
- 15) "Boost C++ Libraries," Web, 09 Dec 2012, <<http://www.boost.org/>>.
- 16) N. Llopis and C. Nicholson, "UnitTest++ version 1.4," Web, 9 Dec 2012, <<http://sourceforge.net/projects/unittest-cpp/>>.
- 17) "Kitware: Proven Open Source Solutions," Web, 9 Dec 2012, <<http://www.kitware.com/opensource/provensolutions>>.
- 18) "Qt Libraries version 5," Web, 09 Dec 2012, <<http://qt-project.org>>.
- 19) "Handles (Computing)," Web, 09 Dec 2012, <[http://en.wikipedia.org/wiki/Handle_\(computing\)](http://en.wikipedia.org/wiki/Handle_(computing))>.
- 20) R.E. Alcouffe, *et al.*, "PARTISN: A Time-Dependent, Parallel Neutral Particle Transport Code System," LA-UR-05-3925, Los Alamos National Laboratory, Los Alamos, NM (2005).
- 21) Note for Cylindrical-RZ θ that $\Delta\theta$ is restricted to a uniform 0.25 or 0.5 revolutions at present. And θ has periodic boundary conditions at $\theta = 0.0$ and 1.0 revolution.
- 22) "Message Passing Interface Forum," Web, 2013, <<http://www.mpi-forum.org/>>.
- 23) E. Gabriel, *et al.*, "OpenMPI: Goals, Concepts, and Design of a Next Generation MPI Implementation," Proceedings, 11th European PVM/MPI Users' Group Meeting, Budapest, Hungary, September, 97-104 (2004).
- 24) "MPICH," Web, 01 September 2013, <<http://www.mpich.org/>>.
- 25) MCATK uses boost::mpi⁽¹⁵⁾ with OpenMPI⁽²³⁾ and MPICH⁽²⁴⁾, in order of priority, as the underlying MPI libraries.
- 26) T. Brunner, *et al.*, "Comparison of Four Parallel Algorithms for Domain Decomposed Implicit Monte Carlo," J. Comp. Phys., **212**, 527-539 (2006).
- 27) T. Brunner and P. Brantley, "An Efficient, Robust, Domain-Decomposed Algorithm for Particle Monte Carlo," J. Comp. Phys., **228**, 3882-3890 (2009).

- 28) S. Nolen and T. Adams, "Status of MC++: Providing Flexible Parallelism for ASC," Nuclear Explosives Code Developers' Conference (NECDC-2005 at LLNL), LA-UR-05-0391, Los Alamos National Laboratory, Los Alamos, NM (2005).
- 29) R.E. MacFarlane and D.W. Muir, "The NJOY Nuclear Data Processing System Version 91," LA-12740-M, Los Alamos National Laboratory, Los Alamos, NM (1994).
- 30) L. Carter, *et al.*, "New Probability Table Treatment in MCNP for Unresolved Resonances," LA-UR-98-26, Los Alamos National Laboratory, Los Alamos, NM (1998).
- 31) J. Duderstadt and L. Hamilton, *Nuclear Reactor Analysis*, John Wiley & Sons, Inc., New York, NY (1976).
- 32) T. Adams, *et al.*, "THE FREE GAS THERMAL TREATMENT IN MCATK: Sampling the Thermal Motion of the Target Nucleus," LA-UR-12-25563, Los Alamos National Laboratory, Los Alamos, NM (2012).
- 33) J. Sweezy, T. Adams, and S. Nolen, "Implementation of Doppler Broadening of the Neutron Elastic Scattering Cross Section in MCATK," LA-UR-12-15645, Los Alamos National Laboratory, Los Alamos, NM (2012).
- 34) J. Conlin, *et al.*, "Continuous- $S(\alpha, \beta)$ Capability in MCNP," LA-UR-12-00155, Los Alamos National Laboratory, Los Alamos, NM (2012).
- 35) G. Bell and S. Glasstone, *Nuclear Reactor Theory*, Krieger Publishing Company, Malabar, Florida (1985).
- 36) J. Sweezy, T. Adams, and S. Nolen, "REACTION SAMPLING IN MCATK: Using the Thermal Motion of the Target Nucleus to Perform Elastic and Inelastic Scattering," LA-UR-13-21838, Los Alamos National Laboratory, Los Alamos, NM (2013).
- 37) S. Nolen, J. Sweezy, and T. Adams, "Integral Criticality Estimators in MCATK (U)," LA-UR-12-25458, Los Alamos National Laboratory, Los Alamos, NM (2012).
- 38) J. Brockway, P. Soran, and P. Whalen, "Monte-Carlo Eigenvalue Calculation," Monte Carlo Methods and Applications in Neutronics, Photonics, and Statistical Physics Lecture Notes in Physics, **240**, 378-390 (1985).
- 39) S. Nolen, *et al.*, "Time-Dependent Tracking in the Monte Carlo Application ToolKit (U)," Supercomputing in Nuclear Applications and Monte Carlo Conference (SNA+MC-2013 Paris), LA-UR-13-27069, Los Alamos National Laboratory, Los Alamos, NM (2013).
- 40) J. Sweezy, *et al.*, "A Weight Conserving Particle Population Control Method for Dynamic Monte Carlo (U)," Supercomputing in Nuclear Applications and Monte Carlo Conference (SNA+MC-2013 Paris), LA-UR-13-27123, Los Alamos National Laboratory, Los Alamos, NM (2013).
- 41) J. Sweezy, "Bias in Dynamic Monte Carlo Neutron Transport Calculations," LA-UR-14-22633, Los Alamos National Laboratory, Los Alamos, NM (2014).
- 42) A. Sood, R.A. Forster, and D.K. Parsons, "Analytical Benchmark Test Set for Criticality Code Verification," Progress in Nuclear Energy, **42**(1), 55-106 (2003).
- 43) A. Sood, R.A. Forster, and D.K. Parsons, "Analytical Benchmark Test Set for Criticality Code Verification," LA-UR-99-2567, Los Alamos National Laboratory, Los Alamos, NM (1999).
- 44) J. Sweezy, T. Adams, and S. Nolen, "Validation & Verification of the Monte Carlo Application Toolkit (MCATK) for Keff Eigenvalue Calculations (U)," Nuclear Explosives Code Developers' Conference (NECDC-2012 at LLNL), LA-UR-13-20269, Los Alamos National Laboratory, Los Alamos, NM (2013).
- 45) F.B. Brown, B.C. Kiedrowski, and J.S. Bull, "Verification of MCNP5-1.60 and MCNP6-Beta-2 for Criticality Safety," LA-UR-12-21041, Los Alamos National Laboratory, Los Alamos, NM (2012).
- 46) R. Aulwes, "Integrating Fault-Tolerance into the Monte Carlo Application ToolKit," Nuclear Explosives Code Developers' Conference (NECDC-2010) LA-UR-10-06583, Los Alamos National Laboratory, Los Alamos, NM (2010).
- 47) R. Aulwes, "Integrating Fault-Tolerance into the Monte Carlo Application ToolKit," LA-UR-13-23852, Los Alamos National Laboratory, Los Alamos, NM (2013).
- 48) R. Aulwes and A. Zukaitis, "Thread Divergence and Photon Transport on GPUs," Supercomputing in Nuclear Applications and Monte Carlo Conference (SNA+MC-2013 Paris), LA-UR-13-27057, Los Alamos National Laboratory, Los Alamos, NM (2013).