Nuclear Science
NEA/NSC/R(2021)3
February 2023
www.oecd-nea.org

Statistical Tests for Diagnosing Fission Source Convergence and Undersampling in Monte Carlo Criticality Calculations

A Report by the Working Party on Nuclear Criticality Safety Subgroup 6 (SG-6)

OECD
BETTER POLICIES FOR BETTER LIVES

NEA
NUCLEAR ENERGY AGENCY

Nuclear Energy Agency

NEA/NSC/R(2021)3

Unclassified **English text only**

15 February 2023

NUCLEAR ENERGY AGENCY
NUCLEAR SCIENCE COMMITTEE

Statistical Tests for Diagnosing Fission Source Convergence and Undersampling in Monte Carlo Criticality Calculations

A Report by the Working Party on Nuclear Criticality Safety Subgroup 6 (SG-6)

This document is available in PDF format only.

JT03512370

ORGANISATION FOR ECONOMIC CO-OPERATION AND DEVELOPMENT

The OECD is a unique forum where the governments of 38 democracies work together to address the economic, social and environmental challenges of globalisation. The OECD is also at the forefront of efforts to understand and to help governments respond to new developments and concerns, such as corporate governance, the information economy and the challenges of an ageing population. The Organisation provides a setting where governments can compare policy experiences, seek answers to common problems, identify good practice and work to co-ordinate domestic and international policies.

The OECD member countries are: Australia, Austria, Belgium, Canada, Chile, Colombia, Costa Rica, the Czech Republic, Denmark, Estonia, Finland, France, Germany, Greece, Hungary, Iceland, Ireland, Israel, Italy, Japan, Korea, Latvia, Lithuania, Luxembourg, Mexico, the Netherlands, New Zealand, Norway, Poland, Portugal, the Slovak Republic, Slovenia, Spain, Sweden, Switzerland, Türkiye, the United Kingdom and the United States. The European Commission takes part in the work of the OECD.

OECD Publishing disseminates widely the results of the Organisation’s statistics gathering and research on economic, social and environmental issues, as well as the conventions, guidelines and standards agreed by its members.

NUCLEAR ENERGY AGENCY

The OECD Nuclear Energy Agency (NEA) was established on 1 February 1958. Current NEA membership consists of 34 countries: Argentina, Australia, Austria, Belgium, Bulgaria, Canada, the Czech Republic, Denmark, Finland, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Japan, Korea, Luxembourg, Mexico, the Netherlands, Norway, Poland, Portugal, Romania, Russia (suspended), the Slovak Republic, Slovenia, Spain, Sweden, Switzerland, Türkiye, the United Kingdom and the United States. The European Commission and the International Atomic Energy Agency also take part in the work of the Agency.

The mission of the NEA is:

– to assist its member countries in maintaining and further developing, through international co-operation, the scientific, technological and legal bases required for a safe, environmentally sound and economical use of nuclear energy for peaceful purposes;

– to provide authoritative assessments and to forge common understandings on key issues as input to government decisions on nuclear energy policy and to broader OECD analyses in areas such as energy and the sustainable development of low-carbon economies.

Specific areas of competence of the NEA include the safety and regulation of nuclear activities, radioactive waste management and decommissioning, radiological protection, nuclear science, economic and technical analyses of the nuclear fuel cycle, nuclear law and liability, and public information. The NEA Data Bank provides nuclear data and computer program services for participating countries.

Foreword

Monte Carlo criticality calculations involve a two-step procedure based on a stochastic implementation of the power iteration method: first, achieve convergence of the fission source distribution during the "inactive cycles", and then sample from this source during the "active cycles". Both phases involve distinct challenges in order to avoid potential issues related to lack of convergence (for inactive cycles) and undersampling and/or clustering (for active cycles). Over the past several years, under the auspices of the Nuclear Energy Agency (NEA), the Expert Group on Advanced Monte Carlo Techniques (EG-AMCT) of the Working Party on Nuclear Criticality Safety (WPNCS) investigated the phenomena of clustering and undersampling in Monte Carlo criticality calculations. A previous Expert Group on Monte Carlo Source Convergence developed breakthrough methods for graphically assessing the initial convergence of the Monte Carlo fission source, using Shannon entropy or Brownian bridge metrics. Much was accomplished in understanding these phenomena, from both theoretical and practical approaches. Those efforts have led to a number of ideas and challenges for new subgroup study topics. The WPNCS Subgroup 6 (SG-6), which ran during 2019-2020, is a direct follow-on to those previous efforts to improve the understanding of and capabilities for Monte Carlo criticality calculations.

There is a strong need for statistical testing to determine fission source convergence in Monte Carlo criticality calculations. Automation of such tests will greatly streamline and support the work carried out by nuclear criticality safety practitioners. Recent research and development (R&D) work has shown that no single statistical test for convergence is sufficiently reliable, robust and "guaranteed". However, a combination of several standard statistical tests for the similarity of distributions, coupled with a high-fidelity estimate of the fission-matrix source, is sufficiently robust, reliable and repeatable that convergence can be "guaranteed".

During the course of the EG-AMCT studies, a number of statistical metrics and tests were proposed for diagnosing clustering and undersampling. None of these was robust and reliable enough for practical use in production codes. However, the expert group efforts came close. Some recent R&D work stemming from those past efforts has been successful and promising.

SG-6 was established to provide international input and collaboration on the development and implementation of statistical tests for convergence, with the primary goal of having the Monte Carlo codes automatically detect convergence (or lack thereof). Newly proposed statistical tests to detect undersampling (after convergence) are also reviewed.

Acknowledgements

The NEA would like to express its sincere gratitude to the participants from the following institutions: Commissariat à l'énergie atomique et aux énergies alternatives (CEA) Cadarache, CEA Saclay, Gesellschaft für Anlagen- und Reaktorsicherheit (GRS), Institut de radioprotection et de sûreté nucléaire (IRSN), Nuclear Regulation Authority of Japan (NRA), Japan Atomic Energy Agency (JAEA), Institute for Physics and Power Engineering (IPPE), Los Alamos National Laboratory (LANL), Oak Ridge National Laboratory (ORNL), Idaho National Laboratory (INL), US Nuclear Regulatory Commission (NRC), Jacobs, Canadian Nuclear Science Council (CNSC), Teollisuuden Voima Oyj (TVO), the University of Warsaw, the University of Michigan and the University of New Mexico.

List of contributors

Co-ordinator & Editor	**Forrest B. Brown**	fbrown@unm.edu	Dept. Nuclear Engineering, University of New Mexico, MSC01 1120, Albuquerque NM 87131, United States
Contributors	**Forrest B. Brown** **Colin J. Josey** **Michael E. Rising**	cjosey@lanl.gov mrising@lanl.gov	Los Alamos National Laboratory PO Box 1663, MS A143, Los Alamos, NM 87545, United States
	Shawn Henderson	sjhende@sandia.gov	Sandia National Laboratory 1515 Eubank Blvd SE, MS 1141, Albuquerque, NM 87123, United States
	William R. Martin	wrm@umich.edu	Nuclear Engineering Dept., University of Michigan, 2355 Bonisteel Blvd, Ann Arbor, MI, 48109, United States
	Paul Smith **Christophe Murphy**	paul.smith8@jacobs.com christophe_murphy1@jacobs.com	Jacobs, ANSWERS Kings Point House, Queen Mother Square, Poundbury, Dorchester, Dorset DT1 3BW, United Kingdom
	Eric Dumonteil	eric.dumonteil@cea.fr	CEA/DRF/IRFU/DPhN Batiment 703, Orme des Merisiers, 91191 Gif/Yvette, FR
	John D. Bess	john.bess@inl.gov	Idaho National Laboratory PO Box 1625, Idaho Falls, ID 83415-3855, United States
	Andrea Zoia	andrea.zoia@cea.fr	Université Paris-Saclay, CEA, Service d'études des réacteurs et de mathématiques appliquées, 91191, Gif-sur-Yvette, France
	Christopher Perfetti	cperfetti@unm.edu	Dept. Nuclear Engineering, University of New Mexico, MSC01 1120, Albuquerque NM 87131, United States

Table of contents

List of figures

List of abbreviations

ACRR	Annular Core Research Reactor (Sandia National Laboratory, United States)
AGN	Aerojet General Nucleonics (United States)
ATR	Advanced test reactor
CEA	Commissariat à l'énergie atomique et aux énergies alternatives (France)
CNSC	Canadian Nuclear Science Council (Canada)
CRS	Compressed row storage
ENDF	Evaluated Nuclear Data File
EG-AMCT	Expert Group on Advanced Monte Carlo Techniques
F or *FM*	Fission-matrix
GRS	Gesellschaft für Anlagen- und Reaktorsicherheit (Germany)
H	Shannon entropy of fission neutron distribution
HEU	High enrichment uranium
ICSBEP	International Criticality Safety Benchmark Evaluation Project
INL	Idaho National Laboratory (US)
IPPE	Institute for Physics and Power Engineering (Russia)
IRSN	Institut de radioprotection et de sûreté nucléaire (France)
JAEA	Japan Atomic Energy Agency (Japan)
LANL	Los Alamos National Laboratory (US)
LEU	Low enrichment uranium
MC	Monte Carlo
MCNP	Monte Carlo N-Particle transport
MOX	Mixed oxide
NEA	Nuclear Energy Agency
NCS	Nuclear Criticality Safety
NRA	Nuclear Regulation Authority of Japan (Japan)
NRC	Nuclear Regulatory Commission (United States)
ORNL	Oak Ridge National Laboratory (United States)
PWR	Pressurised water reactor
R&D	Research and development
RMS	Root mean square
S	Fission neutron distribution

SG-6	Subgroup 6 of the NEA WPNCS, for work on statistical tests for diagnosing fission source convergence and undersampling in Monte Carlo criticality calculations
SPERT	Special power excursion reactor test
TVO	Teollisuuden Voima Oyj (Finland)
WPNCS	Working Party on Nuclear Criticality Safety (NEA)

Executive summary

Monte Carlo (MC) methods have been used for over 60 years in nuclear criticality safety (NCS) calculations. Significant burdens are placed on nuclear criticality safety analysts to properly run the calculations: (1) the initial guess for fission sites is defined by user input; (2) users must ensure that sufficient neutrons/cycle are used to prevent bias; and (3) users must ensure that enough cycles are discarded so that k_{eff} and the fission source have converged. In practice, a short run produces plots of k_{eff} and Shannon entropy of the fission neutron distribution, then the number of inactive cycles is manually set in the MC code input file, and a final run is made. NCS work often requires parameter studies with hundreds of runs. For these studies, it is not practical to follow all the recommended procedures, and conservative over-estimates are used for the number of inactive cycles.

Recent work has addressed these burdens, providing automated acceleration of the convergence process, statistical tests for automatically determining convergence, and additional tests to assess whether a sufficient number of neutrons/cycle was used. These automated methods do not require user input and provide quantitative evidence of convergence. Testing on a wide range of problems has demonstrated that the methods are robust and reliable.

Subgroup 6 (SG-6), under the auspices of the Nuclear Energy Agency (NEA) Working Party on Nuclear Criticality Safety (WPNCS), was established to exchange technical information on statistical testing to determine convergence of the MC power iteration process. Such techniques, representing the current state of the art for convergence testing, are reviewed and presented in this report, in particular:

- seven tests using the slope-test on metrics computed within a block of cycles;
- an eighth test to compare the average Shannon entropy for the block against a reference Shannon entropy from the fission-matrix solution at the end of the block;
- three goodness-of-fit tests to compare the reference solution and the fission neutron distribution averaged over the cycles in the block.

The ultimate goal is a collection of statistical tests for convergence that provides overwhelming evidence of convergence, a “guarantee” that convergence was achieved.

This proposed series of 11 tests was trialled against an extended series of typical criticality calculation cases covering a variety of features, and has proved to be effective and robust. The outcome of such a trial is detailed in the report.

Other statistical tests under investigation are discussed as potential further additions to the convergence tests suite.

Finally, this report demonstrates that the automated convergence and statistical testing methods presented here remove common user pitfalls and provides quantitative, robust evidence of convergence, benefitting the work of nuclear criticality safety practitioners.

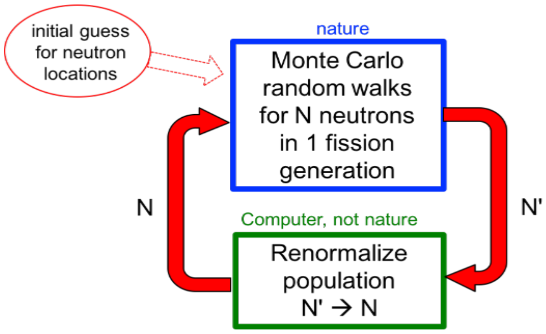
1. Introduction and background

1.1. Monte Carlo criticality calculations – current state of the art

For over 60 years, Monte Carlo (MC) criticality calculations for k_{eff} and the fission distribution have been solved using the power method, also called the method of successive iterations (Goad and Johnston, 1959; Brown, 2016).

- A generation-based iteration scheme is used, and the neutron population is renormalised between successive iterations, as shown in Figure 1. Each generation is started with the same total number of neutrons N (or equivalently for mcnp6, total weight) and a number of neutrons N' is produced. N' is a stochastic variable, hence the renormalisation to N neutrons for the next generation is a biased process (Brissenden and Garlick, 1986). Further, the renormalisation reduces the number of independent neutron fission chains, introducing correlation among cycles. The correlation manifests as clustering for very low N (Dumonteil et al., 2014; Brown, 2017; Nowak et al., 2016).

Figure 1. MC iteration scheme



Source: NEA, 2020.

- The bias in k_{eff} introduced by the renormalisation process is negative and proportional to 1/N (Gelbard and Prael, 1974; Brissenden and Garlick, 1986). The eigenfunction (i.e. fission distribution) is also biased proportional to 1/N, with additional complications of being too low in high-importance regions and too high in low-importance regions.
- The MC iteration scheme begins with an initial guess for k_{eff} and the fission distribution. Iterations (called inactive cycles) are performed without tallies until k_{eff} and the fission distribution have converged to their stationary state. After convergence, tallies are turned on, and iterations (called active cycles) are continued until sufficiently small uncertainties are obtained for desired results.

Significant burdens are placed on nuclear criticality safety (NCS) analysts, however, to properly run the MC calculations:

- The initial guess for the fission source locations must be defined by user input. Ideally, the initial guess should be similar to the steady-state stationary distribution (i.e. the final result). Care must be taken to select the initial starting sites such that there are some fission sites located in each fissionable region of the problem.
- Users must ensure that a sufficient number of neutrons per cycle is used to prevent bias in k_{eff} and errors in the fission source shape. For small to moderate sized problems, it has been shown that using $N > 10\,000$ neutrons per generation effectively removes the bias in k_{eff} (Brown, 2009; Perfetti et al., 2017). For larger physical systems and for loosely-coupled problems, 100 000 or 1 M or more neutrons per generation may be needed.
- Users must also ensure that a sufficient number of initial cycles is discarded so that k_{eff} and the fission source have converged to the steady-state, stationary distribution. In a typical traditional calculation, a short trial run is made to determine the number of inactive cycles based on plots of k_{eff} and the Shannon entropy (Ueki and Brown, 2002) of the fission neutron source distribution, H. Then, the number of inactive cycles to achieve convergence is manually set in the MC code input file. Finally, a definitive run is made to determine results, with additional cycles run until the uncertainties on results are small enough.

While the procedures for running MC criticality problems properly are straightforward, they can be burdensome when many different problems need to be analysed:

- It is common in NCS work to perform parameter studies, varying one or more parameters such as spacing or density and then running a MC criticality problem. Often 100s or 1 000s of runs must be made to span the range of expected conditions. For these parameter studies, it is not practical to follow all of the recommended procedures discussed above, and typically very conservative over-estimates are used for the number of inactive cycles, resulting in excessive computer time. Additionally, determining proper convergence of the iteration process using plots of k_{eff} and the H is highly subjective due to the statistical variations in cycle-wise variations in k_{eff} and the H, and most users will conservatively choose the number of inactive cycles to be much larger than necessary.
- Similar considerations apply to the running of large suites of benchmark problems as part of NCS validation of computational methods.
- There have been no tests available for determining whether a sufficient number of neutrons per cycle was used to reduce the renormalisation bias to negligible levels (other than repeated calculations with different numbers of neutrons per cycle, an extremely burdensome task).
- Separate from most NCS work, but important to the analysis of nuclear reactor core physics, is the issue of MC source convergence when a higher-level iterative scheme is used to alternate between MC neutronics and thermal hydraulic codes to determine consistent power and temperature distributions. These coupled multi-physics calculations are not amenable to manual observation of the MC source convergence. In addition, increased demands on resolution detail for fission rate tallies and material temperature distributions often require 500k or 1M or more neutrons/cycle.

1.2. Past efforts on convergence testing for MC criticality calculations

From the 1950s through the early 2000s, convergence testing consisted of monitoring k_{eff} vs iteration cycle to determine when the asymptotic steady state was achieved. In practice, it was necessary to make a trial run and plot k_{eff} vs cycle, and then invoke a significant amount of practitioner judgement to determine the asymptotic condition in the presence of statistical noise from the MC simulation. From the 1950s through the 1990s, practitioners generally used 100s or 1 000s of neutrons/cycle due to limitations on computer memory, leading to significant statistical noise in the cycle plots. In those times, practitioners were unaware of the need to run 10k or more neutrons/cycle to avoid source renormalisation bias and were also unaware of the fact that the fission distribution can take significantly more cycles to converge than k_{eff}. Some very experienced users would also monitor fissions at symmetric locations in a reactor core, and if there were differences would continue to run more neutrons until symmetric results agreed within statistics.

In the early 2000s, Shannon entropy of the fission neutron distribution was introduced into MC criticality calculations (Ueki and Brown, 2002; Ueki and Brown, 2005; Brown, 2006) as a metric for monitoring the convergence of the fission source, based on information-theoretic considerations. It rapidly became evident that Shannon entropy (H) should be used to determine problem convergence, rather than k_{eff}. At about the same time, it was demonstrated that there was significant source renormalisation bias in results if only 100s or 1 000s of neutrons/cycle were used, and the "best practices" recommendations for using 10k or more neutrons/cycle (and 100k or more for large problems) were introduced (Brown, 2009). In the mid-2000s, basing convergence testing on H vs cycle and with less statistical noise due to larger neutrons/cycle, the assessment of convergence of the MC criticality power iteration process was greatly improved. The assessment process was still manually intensive due to the need for trial runs, examining plots and eyeballing the cycle that produced asymptotic behaviour.

In the late 2000s, there were a few attempts at automated convergence testing (Nease and Brown, 2005; Brown et al., 2013). These previous attempts at automated convergence testing performed statistical slope-tests for only two quantities, k_{eff} and H, for blocks of cycles. While that is essentially what users do in examining plots of k_{eff} and H to assess convergence, the schemes were not sufficiently reliable or conclusive for NCS purposes. There were false positives or negatives 5-10% of the time in trying to automatically determine the cycle at which asymptotic behaviour was reached, an error rate much too high for NCS calculations.

1.3. Current efforts on convergence testing for MC criticality calculations

In the current work described in this report, 11 statistical tests are performed, including additional tests with sensitivity to spatial effects (e.g. the entropy for x-, y-, and z-marginal distributions) and tests for goodness-of-fit for distributions, with the fission-matrix eigenfunction used as a reference solution. While the previous work suffered from occasional false positives or false negatives, the current approach using 11 statistical tests has been reliable and robust for all problems tested. In addition, reporting of the results for the statistical tests provides documented evidence of the convergence assessment. It is quite evident that one or two statistical tests for convergence are not sufficient to "guarantee" that the power iterations have converged. Many more tests are needed to provide sufficient confirmation of convergence, and it is also important to include tests on a wide variety of both metric quantities and actual fission neutron distributions.

Recent work (Brown and Josey, 2018; Brown and Martin, 2018; Brown et al., 2019) has addressed these burdens, providing automated acceleration of the convergence process, statistical tests for automatically determining convergence, and additional tests to assess whether a sufficient number of neutrons per cycle was used. These automated methods determine the mesh spacing used for both statistical tests of the neutron distribution and the fission-matrix (with no user input) from cycle one estimates of physics results, provide adaptive meshing and a fission-matrix reference solution, and provide quantitative evidence of convergence. Testing on a wide range of problems has demonstrated that the methods are robust and reliable.

2. Calculational framework to support automated convergence testing

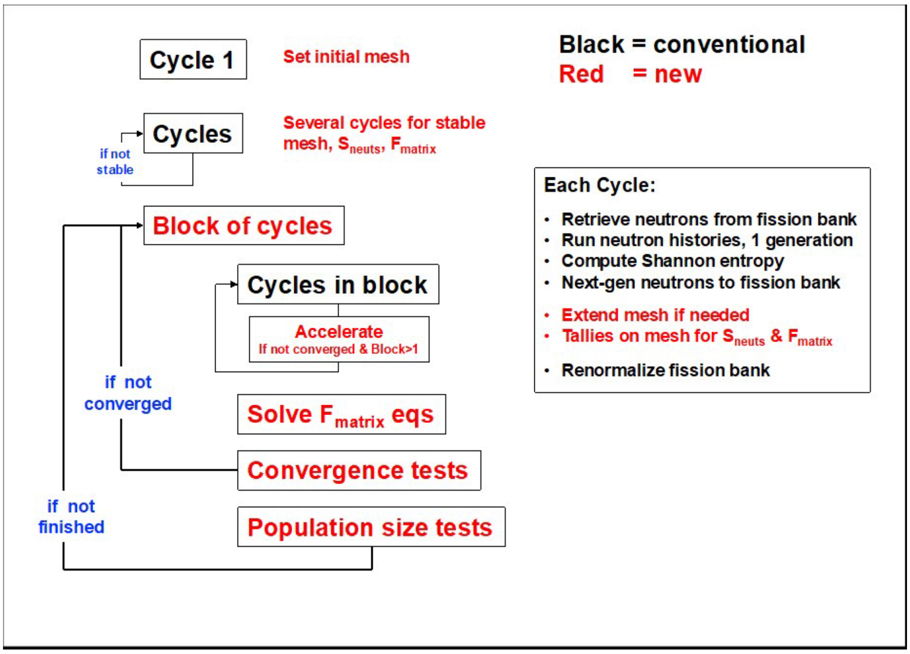
In discussing statistical testing for the convergence of the MC power iteration process, a number of terms will be used to describe k_{eff} and the fission neutron distribution in a cycle, as well as a reference solution obtained using the fission-matrix method. In the definitions below, the subscript "neut" denotes quantities that refer to the actual distribution of neutrons in the MC simulation, while the subscript "FM" refers to separate reference quantities obtained using the fission-matrix method:

k_{eff} or k_{neut}	k-effective for a cycle determined in the normal manner by the MC code, based on the neutron simulation within a cycle;
k_{FM}	k-effective for a reference solution obtained by solving the fission-matrix equations;
$\boldsymbol{S}$ or $\boldsymbol{S}_{neut}$	the distribution of fission neutrons at the end of a cycle in the MC simulation, tallied in bins using the mesh for Shannon entropy calculations;
$\boldsymbol{S}_{FM}$	the distribution of fission neutrons for the fission-matrix reference solution obtained by solving the fission-matrix equations, based on the mesh for Shannon entropy;
H or H_{neut}	the Shannon entropy computed for $\boldsymbol{S}_{neut}$, the fission neutron distribution in the MC simulation;
H_{FM}	the Shannon entropy computed for $\boldsymbol{S}_{FM}$, the reference solution.

In all that follows, a mesh is needed for computing Shannon entropy of the fission neutron distribution, and that same mesh is used for tallying the neutron distribution for statistical testing and the elements of the region-to-region fission-matrix probabilities. The reference solution from the fundamental mode eigenfunction of the fission-matrix corresponds to that same mesh.

In order to support the automated use of statistical testing for convergence analysis, some changes are required to existing MC code algorithms. These changes are needed at a high level, and do not alter the low-level coding used for neutron simulation. Figure 2 illustrates the basic algorithmic flow required. A few initial cycles are needed to establish a mesh (for tallying the fission neutron distribution, in order to compute Shannon entropy, and for beginning related tallies of the fission-matrix for determining a reference solution). Then a block of cycles is run, where the number of cycles in a block is typically 10 or greater. Various metrics are computed and retained for each cycle in the block (e.g. pathlength estimate of k_{eff}, Shannon entropy of the fission neutron distribution, etc.). At the end of the block of cycles, statistical tests are performed on the sequence of metrics for the block and also goodness-of-fit tests can be performed comparing the fission neutron distribution at the end of the block to a reference solution. Currently, the reference solution is the fission neutron distribution obtained by solving the fission-matrix equations for the fundamental eigenfunction.

Figure 2. Calculational framework to support automated convergence testing



Source: NEA, 2020.

This SG-6 report is focused on the statistical testing for source convergence analysis and adequate population size (i.e. neutrons/cycle), and not on a particular MC computer code. Nevertheless, practical demonstration of the statistical testing methods is necessary for a wide variety of realistic MC application problems. To this end, the current development version of *mcnp6 – mcnp6.3*, tentatively scheduled for general release in 2021 – was modified to provide the calculational framework illustrated in Figure 2. The statistical testing methods detailed in Section 3 and Section 4 were then implemented and applied to the test problems detailed in Section 5. A few details on the calculational framework are provided in the next subsections on the automated adaptive meshing, fission-matrix method, and tallying and solving the fission-matrix equations.

2.1. Automated meshing for convergence analysis

An adaptive Cartesian mesh is used for determining Shannon entropy, H_{neut}, of the fission neutron source distribution, $\boldsymbol{S}_{neut}$, and the starting/ending fission point bins for the fission-matrix method (Brown et al., 2013; Carney et al., 2013). By default, a mesh spacing of L_{fiss} is used, where L_{fiss} is the root mean square (RMS) distance from birth to fission determined automatically during the initial cycle:

$$L_{fiss} = \sqrt{\sum_{n=1}^{N'} w_n \left|\vec{r}_n - \vec{r}_{0,n}\right|^2 \Big/ \sum_{n=1}^{N'} w_n}\,,$$

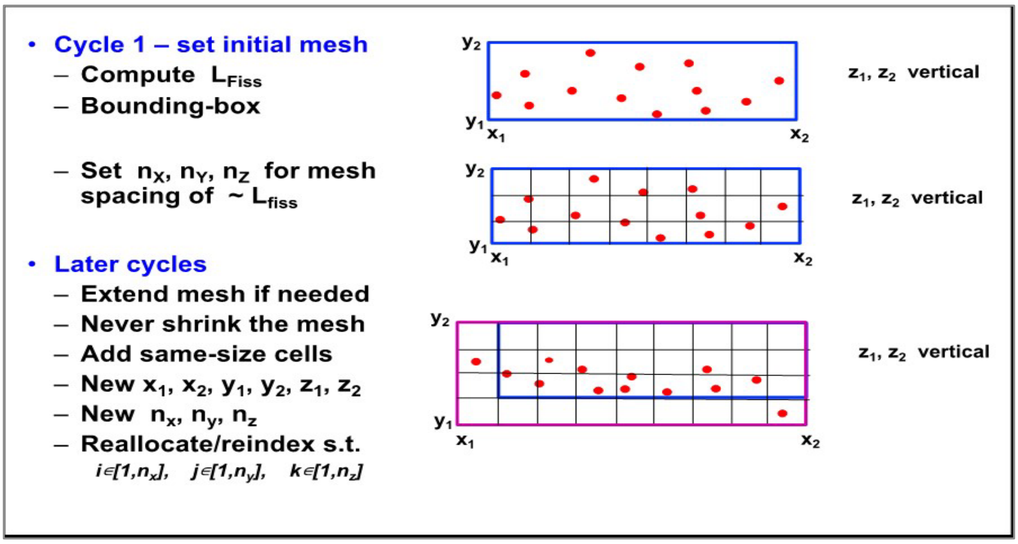
where $\vec{r}_n$ is the location of a next-generation neutron born in fission, w_n is the weight of that neutron, $\vec{r}_{0,n}$ is the location where the parent neutron started, and N' is the number of next-generation neutrons created in the cycle. L_{fiss} is trivial to calculate, using just the starting birth site and the resultant next-generation sites within a cycle, with tallies made only at the end of a cycle (i.e. there is no need to tally during the neutron random walks).

L_{fiss} is essentially the "migration length" from a fission in one generation to a fission in the next generation.

For the present work where only the global shape of the fundamental mode eigenfunction is needed, numerical experiments were performed on the test problems cited below and on 100s of the International Criticality Safety Benchmark Evaluation Project (ICSBEP) (NEA, 2019) benchmark problems to determine a recommended mesh resolution. It was found that using $1.0{*}L_{fiss}$ was suitable for all problems tested. That is, using a coarser mesh did not adequately capture the global shape of the fission distribution; using a finer mesh required many additional iterations for the fission-matrix to stabilise and produce a reliable solution. (If the mesh is too fine, very many neutrons may be needed to produce fission-matrix tallies that are not too noisy.) While there are user input options to override the factor applied to L_{fiss} for determining the mesh spacing, all of the testing discussed below used the default spacing of $1.0{*}L_{fiss}$.

An adaptive Cartesian mesh is used for determining Shannon entropy, H_{neut}, of the fission source distribution, $\boldsymbol{S}_{neut}$, and the starting/ending fission point bins for the fission-matrix method. During the first (inactive) cycle, tallies are made to estimate the RMS distance to fission, L_{fiss}. While the value of L_{fiss} is not precise, since it is based on the initial source distribution guess and only 1 cycle of neutrons, it provides an adequate physics-based distance for setting a mesh size for convergence analysis. By default, a mesh spacing of L_{fiss} is used. Then a Cartesian mesh is defined for the axis-aligned bounding box of fission points at the end of the first cycle, with $N = n_x n_y n_z$ mesh cells. The mesh storage is adaptive: if during subsequent cycles a source tally must be made in a location outside the mesh, the appropriate mesh dimensions n_x, n_y, n_z are extended (keeping the same spacing) to include that region. Figure 3 illustrates the automated process of establishing and extending the mesh.

Figure 3. Automated adaptive meshing



Source: NEA, 2020.

2.2. Computing Shannon Entropy

At the end of each cycle, Shannon entropy (Ueki and Brown, 2002; Ueki and Brown, 2005; Brown, 2006; Cover and Thomas, 2006) of the fission neutron distribution can be computed by simply counting the number of end-of-cycle fission neutrons in each of the mesh cells. Since the mesh is adaptive and may change during the iteration process, it is appropriate to use a normalised variant of Shannon entropy, where the conventional entropy of the distribution is divided by its maximum value. The normalised Shannon entropy then varies between 0 and 1, even when the mesh is expanded. For a mesh with $n_X \cdot n_Y \cdot n_z$ mesh cells and a (normalised) fission neutron distribution $S_{i,j,k}$ on the mesh:

$$H = \frac{\sum_{i=1}^{n_X} \sum_{j=1}^{n_Y} \sum_{k=1}^{n_Z} \; S_{i,j,k} \cdot \ln S_{i,j,k}}{\ln(n_X \cdot n_Y \cdot n_Z)}$$

A related metric for the fission neutron distribution that accounts for spatial variation is the marginal entropy, H_X, obtained by first collapsing the fission neutron distribution in the y- and z-dimensions, and then computing Shannon entropy for the 1-dimensional distribution in x:

$$H_X = \frac{\sum_{i=1}^{n_X} \left\{ \sum_{j=1}^{n_Y} \sum_{k=1}^{n_Z} S_{i,j,k} \right\} \cdot \ln \left\{ \sum_{j=1}^{n_Y} \sum_{k=1}^{n_Z} S_{i,j,k} \right\}}{\ln(n_X)}$$

Similarly, H_Y and H_Z marginal entropies can be computed for the y- and z-directions.

The marginal entropies can be useful in detecting side-to-side shifts or oscillations in the neutron distribution during a block of cycles.

2.3. Using the fission-matrix method to obtain a reference solution

The fission-matrix, $\boldsymbol{F}$, is an $N \times N$ matrix (where N is the number of mesh cells), where each element $\boldsymbol{F_{I,J}}$ represents the number of fission neutrons produced in mesh region $\boldsymbol{I}$ per fission neutron source in region $\boldsymbol{J}$. It is a set of N discretised point-to-point Green's functions representing the connectivity among all of the mesh cells. Given the fission-matrix $\boldsymbol{F}$, the fundamental mode eigenvalue and eigenfunction are obtained by solving the equations:

$$S_{FM} = \frac{1}{k_{FM}} \cdot F \cdot S_{FM}$$

The $\boldsymbol{F_{I,J}}$ elements can be tallied even during the inactive cycles in the MC power iterations and can be accumulated over both inactive and active cycles (Brown et al., 2013). The fundamental eigenfunction of the fission-matrix, $\boldsymbol{S_{FM}}$, is an accurate solution of the criticality problem, and is not subject to bias from source renormalisation like the neutron distribution is. $\boldsymbol{S_{FM}}$ converges much more rapidly than the single-cycle fission neutron distribution, and can be used as a reference solution for assessing convergence of the fission neutron distribution.

While the fission-matrix method has been known since the 1950s (Morton, 1956; Kaplan, 1958), it suffers from memory storage issues. The fission-matrix $\boldsymbol{F}$ potentially requires N^2 tally bins, so that a modest mesh of 100*100*100 would require 10^{12} tally bins, an excessive amount of memory storage. However, in the current implementation, tallies for $\boldsymbol{F}$ are stored using a compressed row storage (CRS) scheme, such that only nonzero tallies are stored. If tallies are needed in an empty slot (i.e. a previously-zero location that is not stored), then the CRS tallies are automatically extended to include the new entries. The

CRS tallies for $\boldsymbol{F}$ are also automatically re-indexed and extended if the underlying mesh is extended.

After cycle 1, inactive cycles proceed in the normal manner, with tallies of $\boldsymbol{S}_{neut}$ and $\boldsymbol{F}$ made at the end of each cycle. The tallies for $\boldsymbol{F}$ are cumulative, including both inactive and active cycles. During these cycles, the $\boldsymbol{S}_{neut}$ and $\boldsymbol{F}$ tallies may be extended if any neutron fission sites are found outside the previous mesh.

The fission-matrix equations are solved after each block of M cycles, where currently M defaults to 10 but can be overridden by user input. M is effectively a "window" for solving the fission-matrix equations and checking on convergence of the fission neutron distribution. Choosing a small value for M reduces the reliability of convergence checking and requires more frequent fission-matrix solutions, whereas a large value for M may unnecessarily delay the convergence checking; a default of 10 was found sufficient for all application problems that were tested.

In the initial iteration cycles, the $\boldsymbol{S}_{neut}$ and $\boldsymbol{F}$ tallies may have significant statistical noise from the MC random walks, and solution of the fission-matrix equations may be unreliable. During the initial iteration cycles, the numbers of nonzero entries in the $\boldsymbol{S}_{neut}$ and $\boldsymbol{F}$ tallies are monitored for changes from one cycle to the next. Currently, if the number of nonzero entries in the $\boldsymbol{S}_{neut}$ tallies changes by more than 2% in successive cycles, or if the number of nonzero entries in the $\boldsymbol{F}$ tallies changes by more than 5% in successive cycles, then the convergence window is shifted. That is, the block of M cycles is reset and shifted by one cycle. The $\boldsymbol{S}_{neut}$ and $\boldsymbol{F}$ tallies are declared stable enough for solution only after the stability tests are met for a consecutive block of M cycles.

The fission-matrix equations are solved at the end of a block of cycles using a standard, brute-force power iteration method. The method is robust, accommodates the non-symmetric $\boldsymbol{F}$ matrix, and requires only matrix-vector products, thus preserving the sparsity of $\boldsymbol{F}$. Solution of the fission-matrix equations yields k_{FM}, $\boldsymbol{S}_{FM}$, and ρ_{FM} (the dominance ratio).

3. Statistical tests for convergence of MC criticality calculations

At the end of a block of cycles, statistical tests for convergence are performed. A fundamental question is "What are appropriate statistical tests for convergence?" The *raison d'être* for SG-6 is to consider that question, render comments on existing proposals for convergence testing, suggest possible additional convergence tests, comment on the quality and effectiveness of the tests, and consider the impact of automated statistical convergence testing on both MC code developers and NCS practitioners.

In many other iterative computational methods for nuclear engineering, convergence of the iterations is often found by computing the relative change from one iteration to the next in some norm of the solution. Often for criticality calculations using deterministic methods, an L_2 or L_∞ norm of the fission neutron source distribution is compared between successive cycles, and convergence is assumed when the relative change in that norm between cycles is less than some prescribed tolerance (Golub and Van Loan, 1989). However, such an approach is not reliable and does not work effectively for MC calculations due to the statistical noise present in each of the cycles; the change in norm may not be large enough relative to the inherent statistical noise to be significant.

To combat the issue of statistical noise present in each cycle of MC power iterations, one approach is to look at a sequence of cycles – a block of cycles – and examine the average value of some metric in one block vs the previous block. Another approach is to compute by least-squares fitting the slope of some metric over the block of cycles. For a converged fission source distribution, there should be steady-state, equilibrium conditions, and every metric should have a slope of 0 over the cycles in the block. Due to statistical noise, it is appropriate to compute the uncertainty on the least-squares slope, and declare a metric to be converged if its least-squares slope is 0 with some statistical confidence level (e.g. 95%).

For a metric Y_i tallied for cycles $i=1,\ldots,n$ in a block of cycles, least-squares fitting to $y=a+bi$ is summarised by Bevington (1992) in Equations 6.13, 6.15, 6.23:

$$s_x = \sum_{i=1}^{n} i, \quad s_{xx} = \sum_{i=1}^{n} i^2, \quad s_y = \sum_{i=1}^{n} Y_i, \quad s_{yy} = \sum_{i=1}^{n} {Y_i}^2, \quad s_{xy} = \sum_{i=1}^{n} i \cdot Y_i, \quad \Delta = n \cdot s_{xx} - s_x^2$$

$$a = \frac{s_{xx} \cdot s_y - s_x \cdot s_{xy}}{\Delta}, \qquad b = \frac{n \cdot s_{xy} - s_x \cdot s_y}{\Delta}$$

The 1-sigma uncertainties on the fitting parameters *a* and *b* are:

$$\sigma^2 = \frac{\sum_{i=1}^{n} (Y_i - a - b \cdot i)^2}{n-2}, \qquad \sigma_a = \sqrt{s_{xx} \cdot \sigma^2 / \Delta}, \qquad \sigma_b = \sqrt{n \cdot \sigma^2 / \Delta}$$

Since the slope, *b,* has a Student's *t* distribution, a 95% confidence level for the slope is $\pm t_{0.025} \cdot \sigma_b$. In addition, $|b|<0.0001$ is an absolute test for the slope being negligibly small. The "slope test" then becomes:

If $\quad |b| < t_{0.025} \cdot \sigma_b \quad or \quad |b| < 0.0001,$

then the slope is not different from 0 at a 95% confidence level, and the metric Y_i is assumed to be stationary over the block of cycles

3.1 Statistical tests on metrics for cycles within the block

Seven tests are performed using the slope-test on metrics computed for cycles in the block.

1. The slope-test is applied to the single-cycle track-length estimates of k_{eff}.
2. The slope-test is applied to the single-cycle collision estimates of k_{eff}.
3. The slope-test is applied to the single-cycle absorption estimates of k_{eff}.
4. The slope-test is applied to the single-cycle Shannon entropy of $\boldsymbol{S_{neut}}$. That is, H_{neut} is computed from $\boldsymbol{S_{neut}}$ separately for each cycle in the block, then the slope-test is applied to the H_{neut} values within the block.
5. The slope-test is applied to the single-cycle marginal distribution in x of $\boldsymbol{S_{neut}}$, H_X.
6. The slope-test is applied to the single-cycle marginal distribution in y of $\boldsymbol{S_{neut}}$, H_Y.
7. The slope-test is applied to the single-cycle marginal distribution in z of $\boldsymbol{S_{neut}}$, H_Z.

An eighth test is applied to compare the average H_{neut} for the block against a reference H_{FM} from the fission-matrix solution at the end of the block:

8. The $\boldsymbol{S_{neut}}$ tallies for each cycle in the block are accumulated and H_{block} is computed for the cumulative sources in the block. H_{FM} is then computed for the fission-matrix eigenfunction determined at the end of the block. If H_{FM} and H_{block} agree within 1%, this test provides strong evidence that the reference fission-matrix eigenfunction and the cumulative fission neutron source distribution for the block agree. However, H_{FM} and H_{block} may differ if there is significant source renormalisation bias in the neutron distribution. Source renormalisation bias is independent of the convergence. That is, if too few neutrons per cycle are used in the calculation, the fission neutron source would still converge, but to the wrong, biased solution. ($\boldsymbol{S_{FM}}$ from the fission-matrix method is not subject to source renormalisation bias.) Thus, passing this test provides evidence of both convergence and an adequate neutron population. If the test is not passed, convergence is not precluded, however, since the cause may be an inadequate population size.

3.2. Goodness-of-fit statistical tests on the fission neutron distribution

At the end of a block of cycles, the fission-matrix equations are solved to provide a reference solution, $\boldsymbol{S_{FM}}$, and the fission neutron distributions, $\boldsymbol{S_{neut}}$, from each cycle in the block are combined into an average distribution, $\boldsymbol{S_{block}}$, for the fission neutrons. Then three goodness-of-fit statistical tests are performed using the distributions $\boldsymbol{S_{block}}$ and $\boldsymbol{S_{FM}}$ (Brown and Martin, 2018):

9. The Kolmogorov-Smirnov goodness-of-fit test is applied at the 95% confidence level to compare the distributions given by $\boldsymbol{S_{block}}$ and $\boldsymbol{S_{FM}}$. Since these are multidimensional distributions, the test is repeated for many random permutations of the ordering (e.g. 25 or more), with the worst-case statistic used to determine the test outcome.
10. The Chi-squared 2-point distribution goodness-of-fit test is performed on $\boldsymbol{S_{block}}$ and $\boldsymbol{S_{FM}}$ at the 95% confidence level.
11. The relative entropy (Kullback-Leibler discrepancy) (Cover and Thomas, 2006) between the distributions for $\boldsymbol{S_{block}}$ and $\boldsymbol{S_{FM}}$ is computed. The Kullback-Leibler discrepancy is related to the G-test for goodness-of-fit (McDonald, 2015), and is an alternative to the chi-squared test. A 95% confidence level may be determined from the chi-squared distribution. If this test is passed, it provides strong evidence of both fission source convergence and adequate population size. However, as for Test 8, this test may fail if the population size is too small (leading to source renormalisation bias). Thus, passing this test provides evidence of both

convergence and an adequate neutron population. If the test is not passed, convergence is not precluded, however, since the cause may be an inadequate population size.

It should be noted that Tests 8 and 11 that involve entropy or relative entropy are sensitive to the neutron population size (number of neutrons per cycle), whereas Tests 9 and 10 are not. The underlying causes of the different sensitivities are not known, and are certainly in need of further investigation.

3.3. Overall convergence assessment

If all of these statistical tests pass (more precisely, if none fail), then convergence is achieved and locked-in for the remainder of the calculation, and active cycles with tallies will begin with the next cycle. To declare convergence, Tests 1-7 and 9-10 are required to pass. Tests 8 and 11 are not required to pass (since they may be affected by source renormalisation bias), but provide additional strong evidence of convergence if they are passed.

Due to the statistical nature of the testing, it is likely that some of the convergence tests may not pass in later cycles. Convergence is not rescinded, however. Typically, some tests that occasionally fail after convergence are passed on most subsequent cycles. After convergence, statistical testing continues to be performed and reported at the end of each block of cycles.

4. Statistical testing for adequate population size (neutrons per cycle)

After convergence, two novel statistical tests are made to assess population size (Brown and Josey, 2018). The tests are intended to detect whether the number of neutrons/cycle is large enough to ensure that the bias in k_{eff} is negligible and the shape of the fission distribution is correct. The tests are based on comparing the Shannon entropy and relative entropy (Kullback-Leibler discrepancy) of Sneut and S_{FM}. If these tests indicate that an insufficient number of neutrons/cycle was used, warning messages are issued. In the current implementation, the number of neutrons/cycle is not altered in the calculation due to concerns over increasing the computer memory and cpu-time, possibly beyond the limits of system resources. Future work may lead to an automatic increase in the number of neutrons/cycle if the tests are not passed.

5. Testing

The automated acceleration and convergence testing methods have been applied to an assortment of criticality problems, including:

- the Monte Carlo N-Particle transport (MCNP) *validation_criticality* suite, containing 31 ICSBEP (NEA, 2019) benchmark problems;
- the MCNP *validation_crit_extended* suite, containing 119 ICSBEP benchmark problems;
- a 2D model of a commercial pressurised water reactor (PWR);
- the Aerojet General Nucleonics (AGN-201) research reactor at the University of New Mexico;
- the Advanced Test Reactor (ATR) at the Idaho National Laboratory;
- the Annular Core Research Reactor (ACRR) burst reactor at the Sandia National Laboratory;
- the NEA Hoogenboom-Martin 3D reactor computer-performance benchmark (Hoogenboom et al., 2011);
- the 3D C5G7 U-mixed oxide (MOX) OECD/NEA benchmark problem;
- the ICSBEP benchmark case low enrichment uranium (LEU) LEU-COMP-THERM-078 (a Sandia experiment);
- a large 3D storage pool with checkerboard arrangement (NEA EG on source convergence benchmark (NEA, 2006) #1);
- a 400 cm tall single reactor fuel-pin unit cell with reflecting boundary conditions;
- the Whitesides problem (k_{eff} of the world);
- a 3D Triga reactor model;
- the OECD/NEA Source Convergence Benchmark (NEA, 2006) #4, *test4s*;
- the Godiva high enrichment uranium (HEU) sphere.

All tests were performed using *mcnp6.3*, an early version of the next release of *mcnp* tentatively scheduled for 2021. For all of these cases, standard *mcnp6* input files were used with Evaluated Nuclear Data File (ENDF/B-VII.1) nuclear data. The only additional input supplied consisted of commands to activate the fission-matrix treatment, automated convergence testing, and fission source acceleration:

```
kopts  fmat= yes
       fmatconvrg= yes
       fmataccel=  yes
```

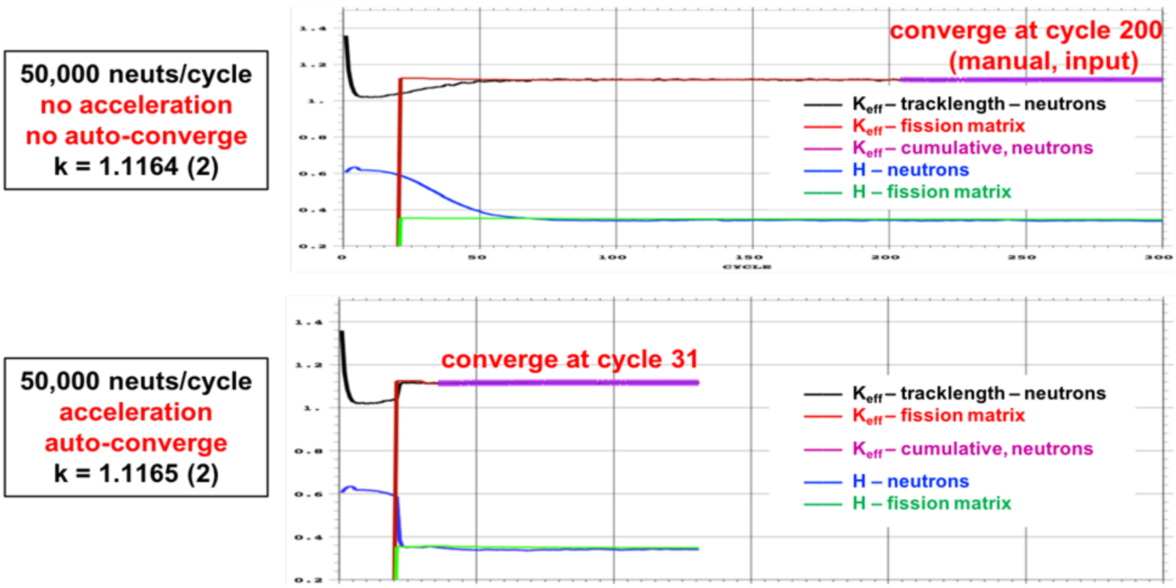
The acceleration of fission source convergence (Brown et al., 2019) is a separate topic from the statistical testing for convergence discussed in this report, but relies on the same mesh, neutron distribution tallies and fission-matrix solution referred to herein. If acceleration of the fission source convergence is to be performed, importance sampling weights of ($\boldsymbol{S_{FM}^{(m)}/S_{neut}^{(m)}}$) are determined for each fission site, where *(m)* represents the mesh bin containing the fission site. The importance sampling weights are used in sampling starting sites for a cycle from the previous-cycle's fission bank. While $\boldsymbol{S_{FM}}$ is determined only at the end of each block of cycles, the importance sampling weights are updated for each cycle, based on the current $\boldsymbol{S_{neut}}$. Acceleration is performed only during inactive cycles. The acceleration method is a nonlinear acceleration method that essentially pushes the neutron distribution towards the reference distribution obtained from the fission-matrix eigenfunction. It typically reduces the number of inactive cycles required for convergence by factors of 2-20x.

In all cases tested, the acceleration and diagnostic tests were effective. Following the warning advice, increasing the number of neutrons/cycle to larger values resulted in passing the population size tests (no warning issued).

Figure 4 shows some results from testing on the *test4s* problem (OECD/NEA Source Convergence Benchmark 4). The upper plots of k_{eff} (for neutrons, fission-matrix, and cumulative) and H (for neutrons and fission-matrix) are from a typical traditional calculation, where a short trial run is made to determine the number of inactive cycles based on plots of k_{eff} and H_{neut}, and then a final run is made after manually setting the number of inactive cycles in the *mcnp6* input file. The number of inactive cycles is somewhat arbitrary, and typical conservative-minded users would choose 150-200 inactive cycles. The lower plots of k_{eff} and H show results from the automated acceleration and convergence methods, with convergence achieved after 31 cycles with no additional user input or trial runs.

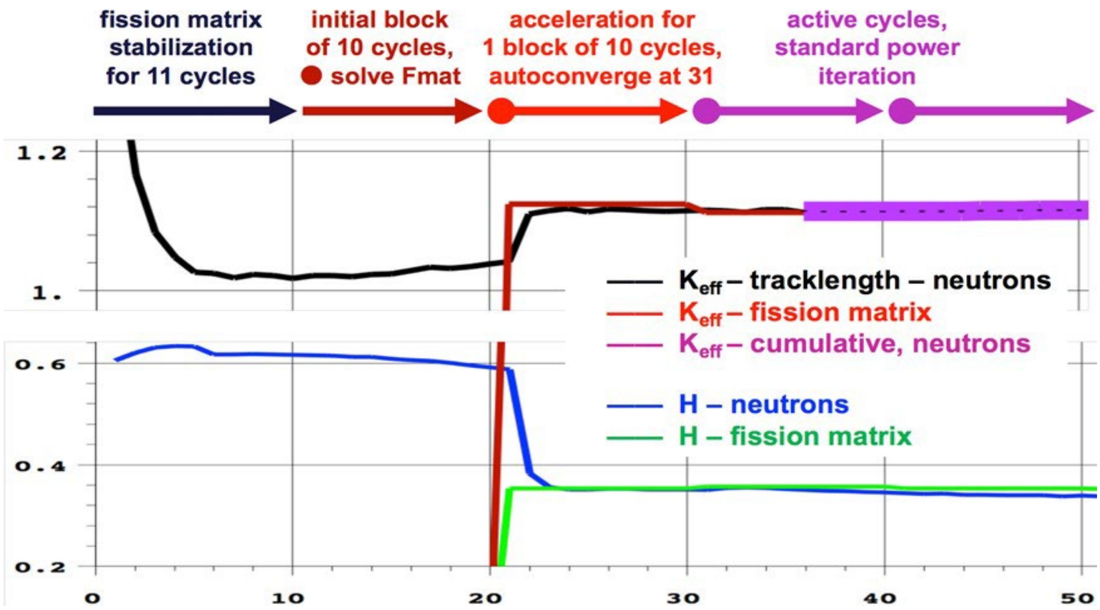
Figure 5 shows detailed results for plots of k_{eff} (for neutrons, fission-matrix, and cumulative) and H (for neutrons and fission-matrix) for the *test4s* problem for the first 50 cycles. For the first 11 cycles, changes in either the number of $\boldsymbol{S_{neut}}$ or $\boldsymbol{F}$ entries delayed the start of the auto-convergence testing. After that, $\boldsymbol{S_{neut}}$ and $\boldsymbol{F}$ are accumulated for a block of 10 cycles, with metrics such as the track-length k_{eff}, collision k_{eff}, H and its marginal values determined for each cycle in the block.

Figure 4. K_{eff} and H by cycle for problem *test4s*, without/with automated acceleration and convergence



Source: NEA, 2020.

Figure 5. Details of initial cycles for problem *test4s*



Source: NEA, 2020.

Figure 6 shows the corresponding screen output for the initial cycles, where the adaptive mesh, source tallies and fission-matrix tallies stabilise, followed by a block of cycles. At the end of the block, the fission-matrix equations are solved for the fundamental mode eigenvalue and eigenfunction. Statistical testing for convergence is then performed, and convergence is declared if all of the 11 tests are satisfied. After the initial block of cycles is completed and the S_{FM} eigenfunction is known, acceleration of the iteration process can be applied.

Figure 7 shows the results of statistical testing at the end of a block when convergence is achieved. It should be noted that the output shown in Figure 7 provides solid statistical evidence that convergence has been achieved. The statistical metrics and target values are provided for each of the tests. In traditional MC calculations, plots of just k_{eff} and H would be generated in a trial run and "eyeballed" to assess convergence. The automated methods provide far more quantitative evidence for convergence. After convergence, the acceleration is discontinued, active cycles are begun and all standard tallies are performed. Also after convergence, the statistical tests continue to be performed at the end of each block (but convergence is never rescinded) along with two additional tests to source shape.

Regarding robustness and reliability of the automated methods, all of the 163 test problems performed as expected, with no false positives for convergence or false negatives (which would not affect results, but would increase computer time). When run using conventional methods, the 31 problems in the *validation_criticality* suite (Mosteller, 2002) required 108 minutes of run time with *mcnp6* using 12 threads. Using the automated acceleration and convergence, the same suite of 31 problems required only 70 minutes to complete correctly. These and other results demonstrate that the combination of automated methods provides effective acceleration and reduces unnecessary conservatism in the number of inactive cycles.

Figure 6. Details of initial cycles for problem *test4s*

```
cycle     k(col)        ctm   entropy    active     k(col)      std dev     chains
    1     1.35733       0.04    0.60521                                    35416
                                          extend H-mesh to:   36 x 35 x 4
    2     1.16857       0.10    0.62080                                    22433
                                          extend H-mesh to:   37 x 35 x 4
    3     1.08223       0.13    0.63109                                    17100
                                          dS=  3%, dF= 34%, shift window
                                          extend H-mesh to:   37 x 36 x 4
    4     1.05100       0.17    0.63410                                    13800
                                          dS=  2%, dF= 19%, shift window
                                          extend H-mesh to:   37 x 37 x 4
    5     1.02827       0.21    0.63348                                    11529
                                          dS=  1%, dF= 14%, shift window
                                          extend H-mesh to:   37 x 37 x 5
    6     1.02118       0.25    0.61732                                     9997
                                          dS=  0%, dF= 10%, shift window
    7     1.02018       0.29    0.61762                                     8746
                                          dS=  1%, dF=  9%, shift window
    8     1.02413       0.32    0.61845                                     7790
                                          dS=  0%, dF=  7%, shift window
    9     1.01974       0.37    0.61766                                     6974
                                          dS=  0%, dF=  7%, shift window
   10     1.01709       0.43    0.61656                                     6313
                                          dS=  1%, dF=  5%, shift window
   11     1.02129       0.48    0.61606                                     5815
                                          dS=  1%, dF=  5%, shift window
   12     1.01705       0.53    0.61452                                     5351
   13     1.02459       0.58    0.61263                                     4975
   14     1.02193       0.65    0.61214                                     4640
   15     1.02741       0.70    0.60894                                     4372
   16     1.03005       0.73    0.60600                                     4091
   17     1.03266       0.78    0.60435                                     3852
   18     1.03369       0.83    0.60065                                     3628
   19     1.03485       0.87    0.59622                                     3426
   20     1.03631       0.91    0.59177                                     3245
   21     1.04159       0.96    0.58774                                     3074

          fmatrix keff= 1.12401,   DR= 0.91098,   iters=    199
```

Source, fission matrix, & mesh stabilization

Block of cycles

Source: NEA, 2020.

Figure 7. Automated convergence reporting for problem *test4s*

```
31    1.11257        1.12   0.35069                                          680

      fmatrix keff= 1.11187,   DR= 0.91653,   iters=    138

      CONVERGENCE INFO & CHECKS:          (based on last  10 cycles)

        entropy for fmatrix eigenvector     =  0.35656
        entropy for neutron  last cycle     =  0.35069    dif=    -1.65%
        relative entropy for last cycle     =  0.00972

        slope of keff (tracklen)       =  4.2E-03,  target:  < 5.1E-03   PASS
        slope of keff (collide)        =  4.6E-03,  target:  < 4.9E-03   PASS
        slope of keff (absorb)         =  4.6E-03,  target:  < 4.9E-03   PASS
        slope of entropy               = -1.4E-02,  target:  < 1.6E-02   PASS
        slope of entropy X marginal    = -1.8E-02,  target:  < 1.9E-02   PASS
        slope of entropy Y marginal    = -1.8E-02,  target:  < 1.9E-02   PASS
        slope of entropy Z marginal    =  1.3E-03,  target:  < 1.6E-03   PASS
        entropy dif, neuts vs fmat     = -9.1E-04,  target:  < 1.0E-02   PASS
        Kolmo-Smirnov, distrib, stat =  2.5E-03,  target:  < 9.1E-02   PASS
        Chi-square,     distrib, stat =  9.0E+01,  target:  < 5.1E+02   PASS
        rel-h-block,    distrib, stat =  2.8E-03,  target:  < 5.1E-03   PASS

        *******************************************************************
        *******************************************************************
        ** FISSION SOURCE HAS CONVERGED, based on last       10    cycles  **
        **    Metrics:                                                    **
        **       slope of keff (tracklen)          is 0   (within uncert) **
        **       slope of keff (collide)           is 0   (within uncert) **
        **       slope of keff (absorb)            is 0   (within uncert) **
        **       slope of entropy                  is 0   (within uncert) **
        **       slope of entropy X marginal       is 0   (within uncert) **
        **       slope of entropy Y marginal       is 0   (within uncert) **
        **       slope of entropy Z marginal       is 0   (within uncert) **
        **       entropy dif, neuts vs fmat        is 0   (within uncert) **
        **    Distribution checks:                                        **
        **       Kolmo-Smirnov, distrib, stat,   neut vs fmat (within conf) **
        **       Chi-square,     distrib, stat,   neut vs fmat (within conf) **
        **       rel-h-block,    distrib, stat,   neut vs fmat (within conf) **
        *******************************************************************
        *******************************************************************
        Convergence is locked-in, even if some tests fail in future cycles

        Active cycles will begin with cycle =        32
        Active cycles will end   with cycle =       131
        Total active cycles to be run        =       100
```

Quantitative Evidence For Convergence

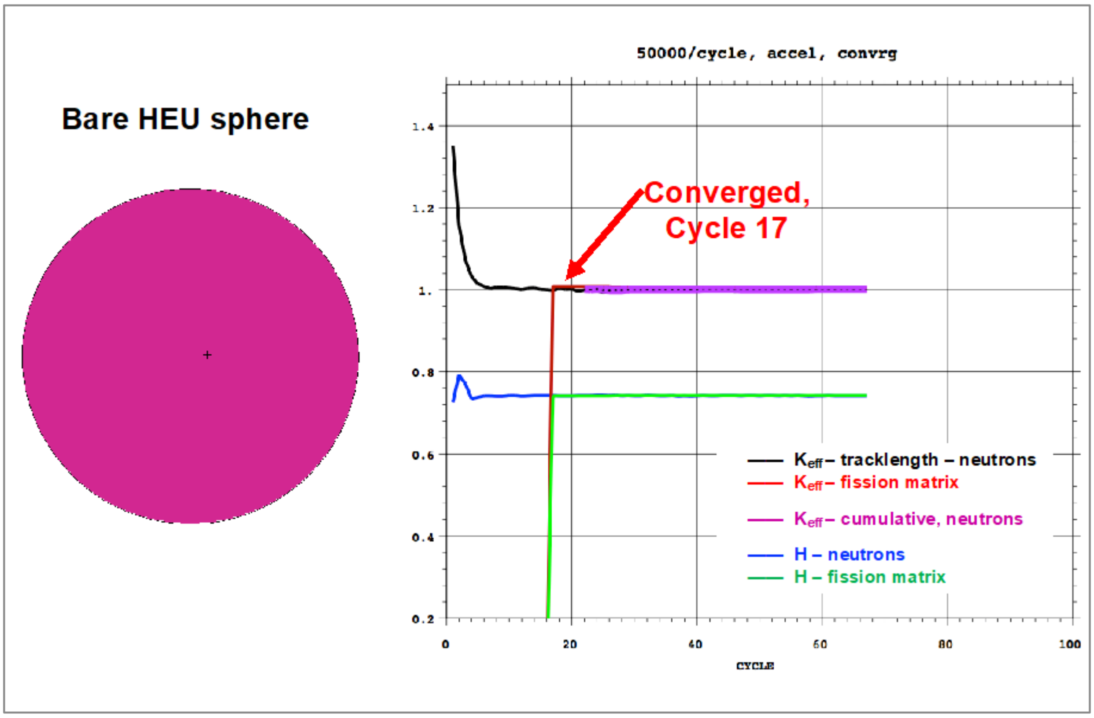
Quantitative Evidence For Convergence

Source: NEA, 2020.

Figures 8-19 show the results of convergence testing for a wide assortment of problems. In each of these figures, plots of k_{eff} and H are shown for calculations run using the fission-matrix, acceleration and automated convergence checking. On each of the plots, the number of inactive cycles used for a conventional calculation (without acceleration or automated convergence checking) is noted. These numbers are typical values, with some coming from trial runs to check on convergence of k_{eff} and H, and some coming from typical user experience.

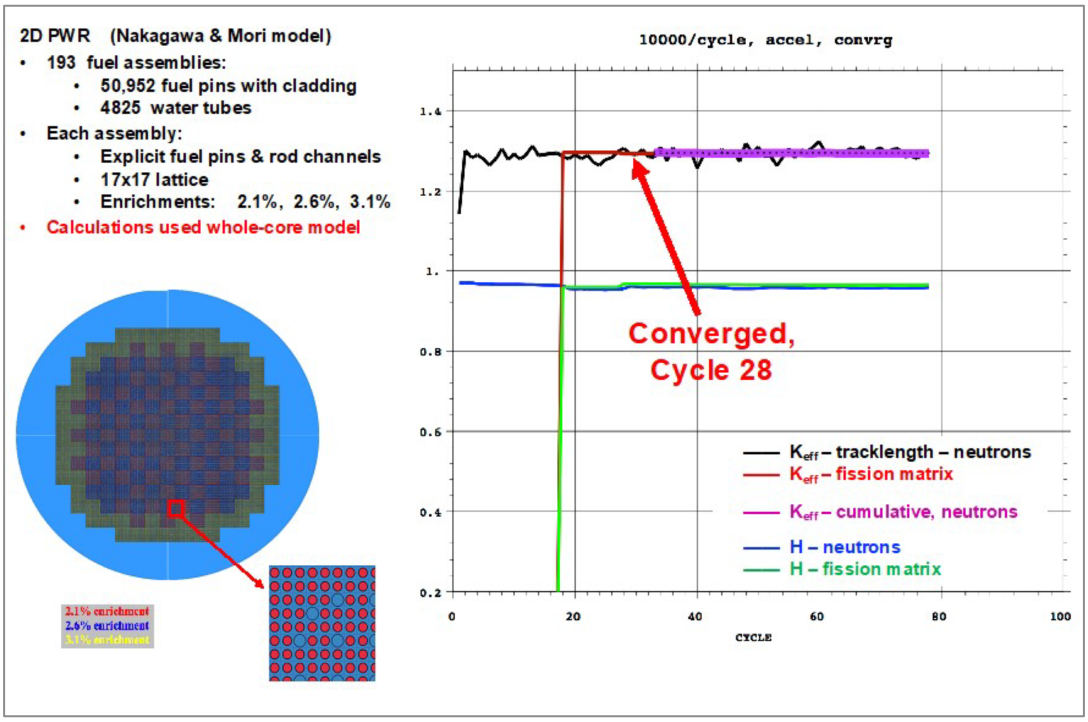
The most difficult problem for convergence is the OECD/NEA Source Convergence Benchmark #1, a large fuel storage pool where over 2 000 inactive cycles are required for convergence using standard methods. With the automated methods, that problem converged properly in only 108 cycles, a factor of 20x improvement due to the acceleration methods using the fission-matrix. Results for that problem are shown in Figure 19.

Figure 8. Godiva sphere test problem



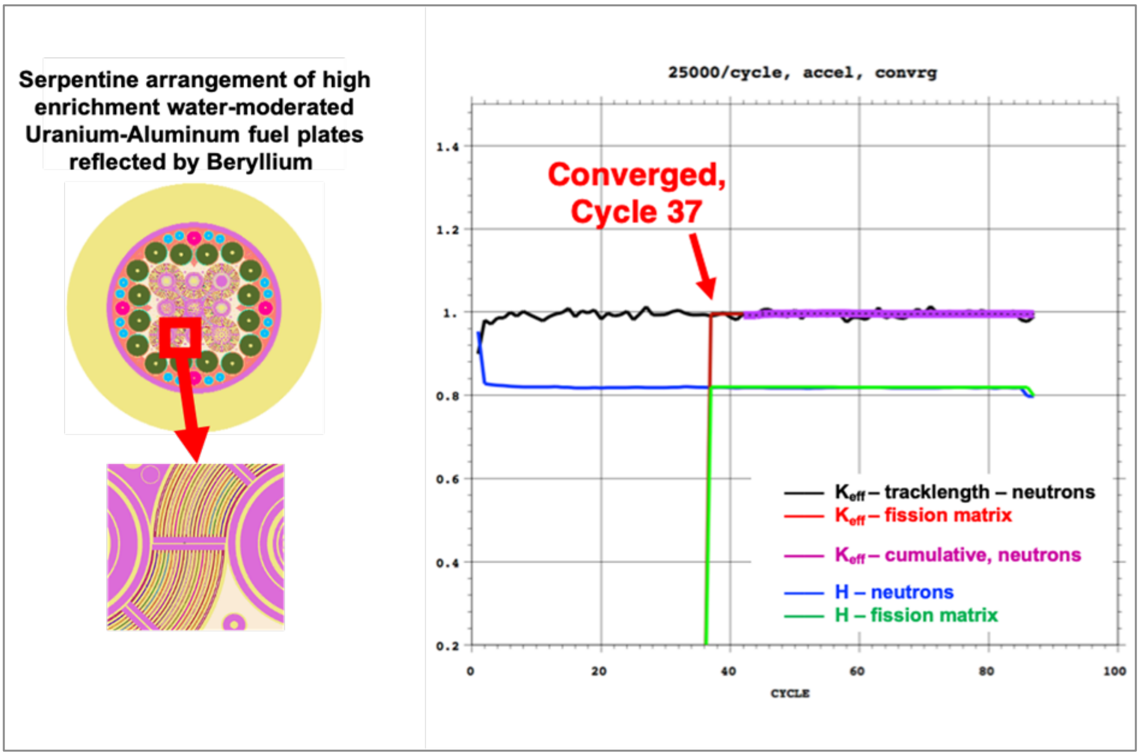
Source: NEA, 2020.

Figure 9. Full-core 2-dimensional PWR test problem



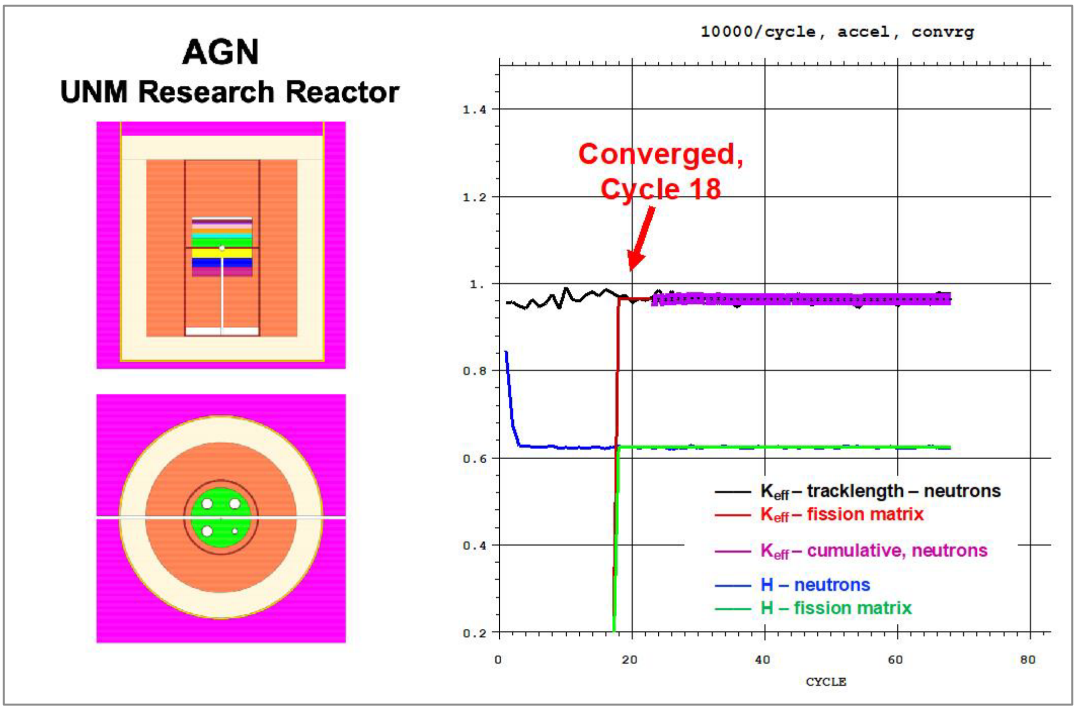
Source: NEA, 2020.

Figure 10. ATR test problem



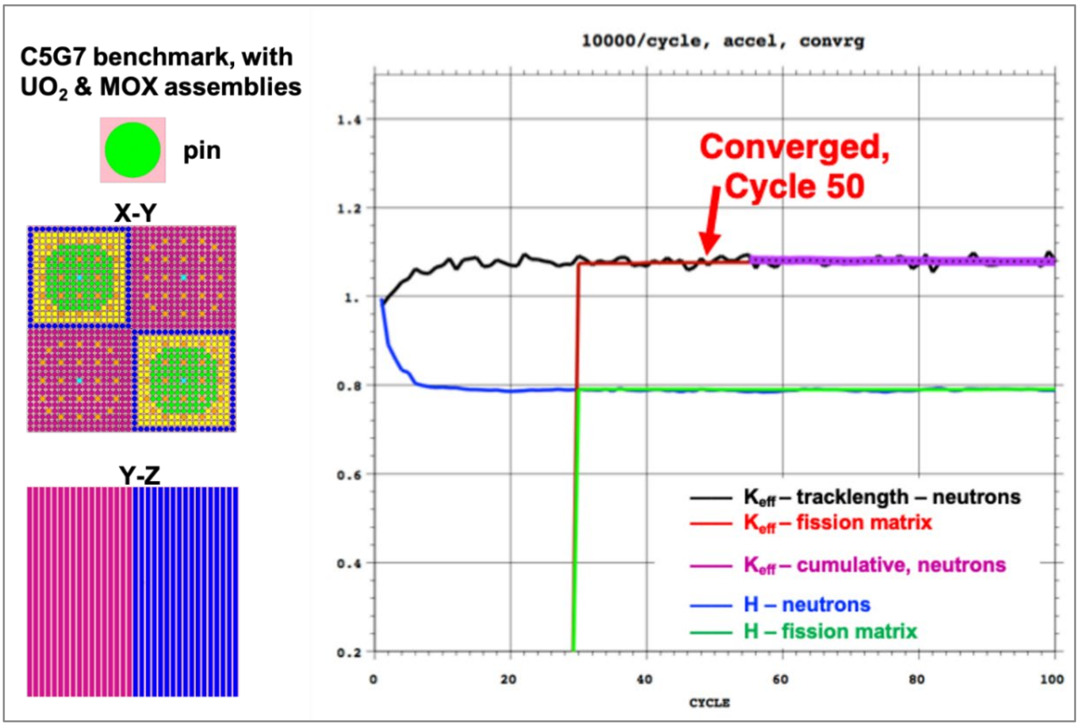
Source: NEA, 2020.

Figure 11. AGN-201 Research reactor test problem



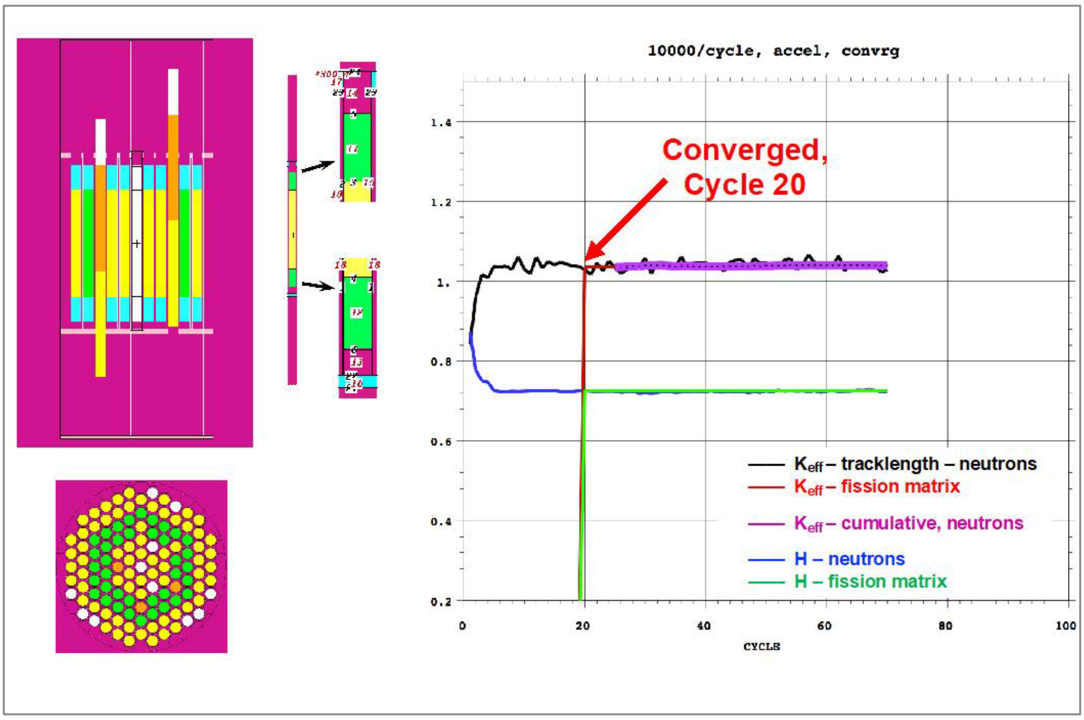
Source: NEA, 2020.

Figure 12. C5G7 Benchmark test problem



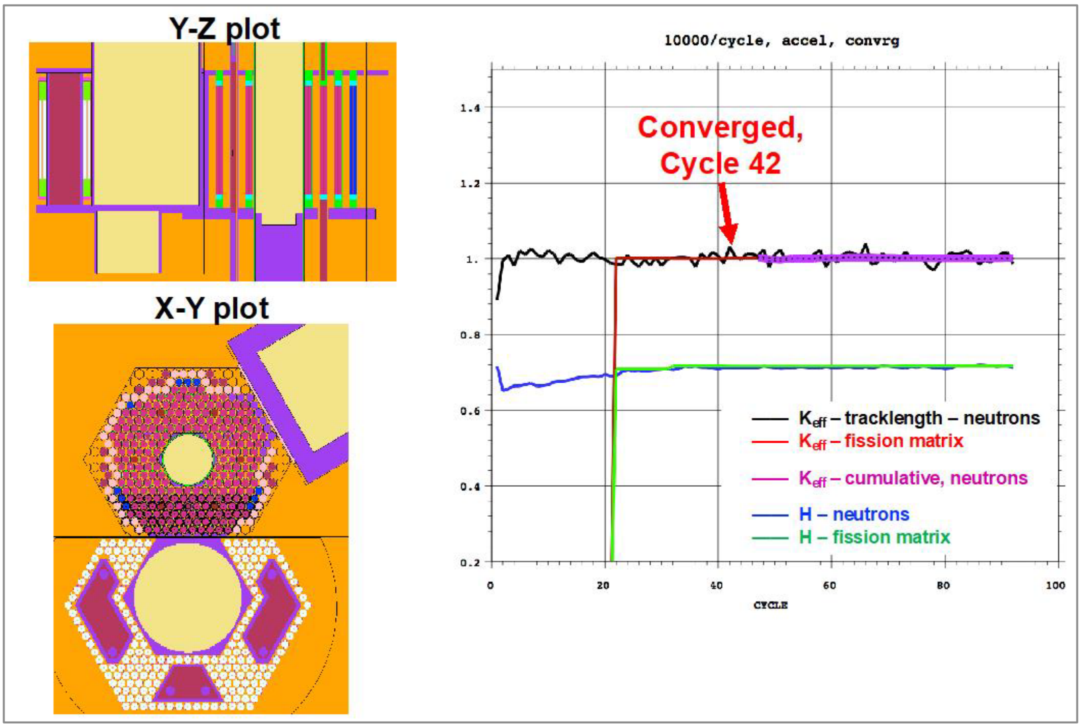
Source: NEA, 2020.

Figure 13. Triga reactor test problem



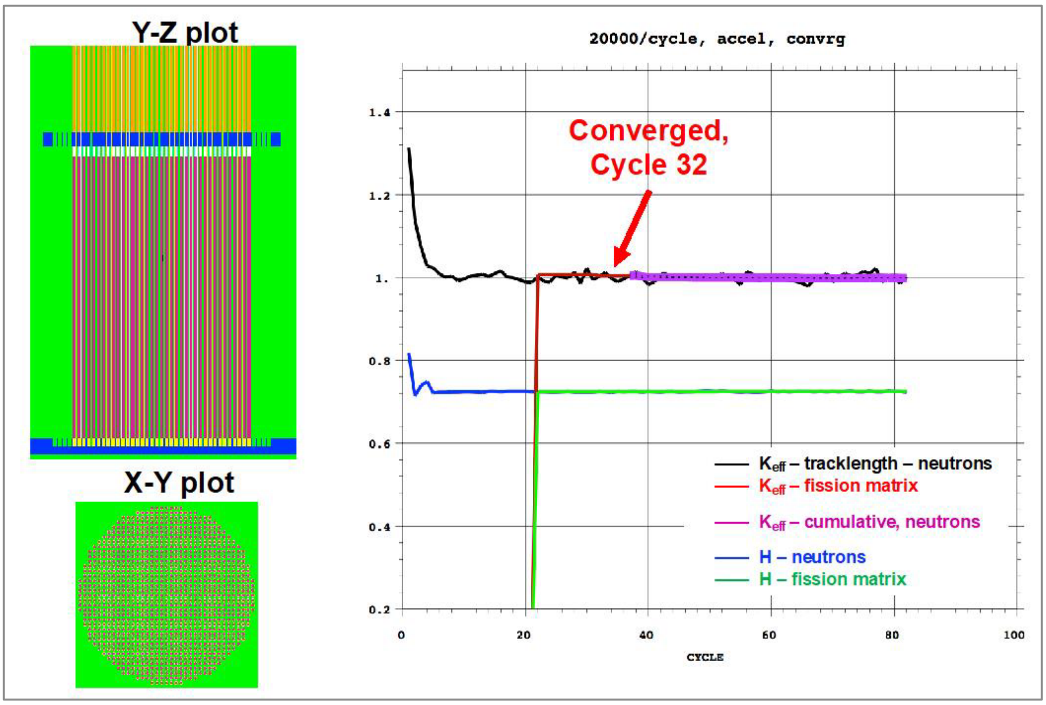
Source: NEA, 2020.

Figure 14. ACRR Sandia reactor test problem



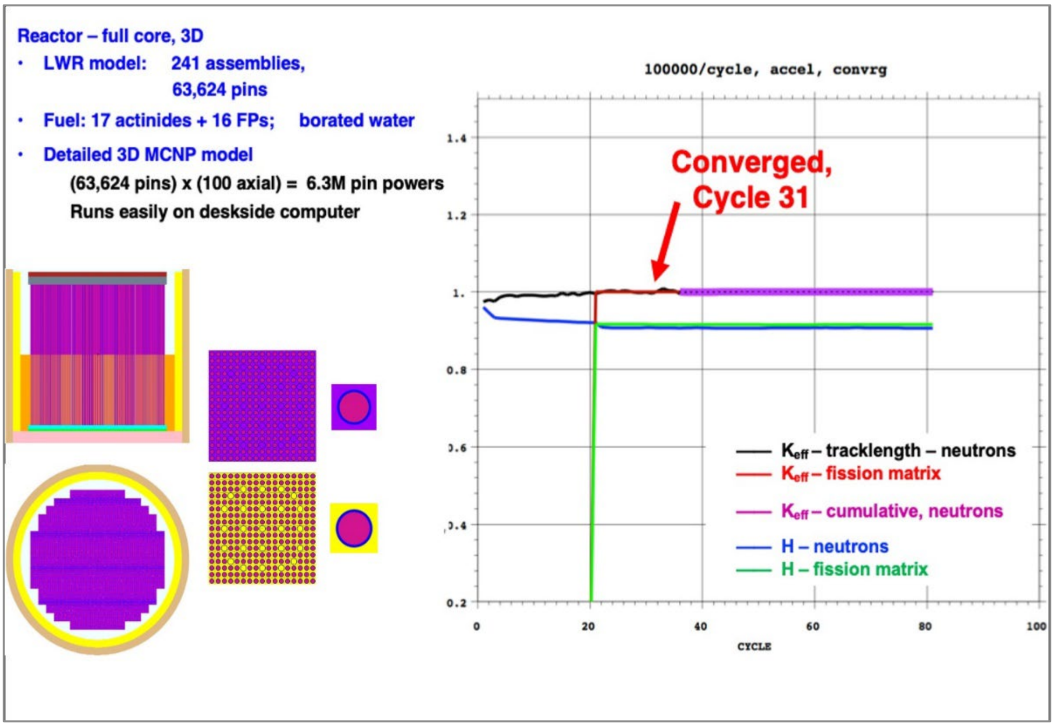
Source: NEA, 2020.

Figure 15. Sandia critical experiment LCT-078-001 (1057 rods) test problem



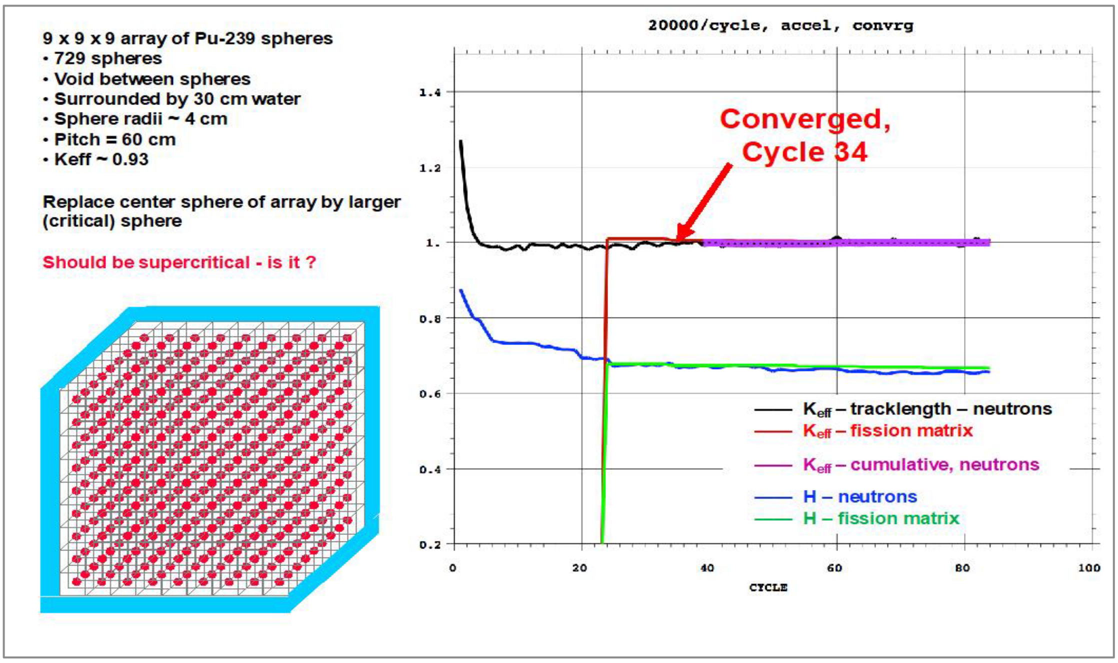
Source: NEA, 2020.

Figure 16. NEA Hoogenboom-Martin Performance Benchmark test problem



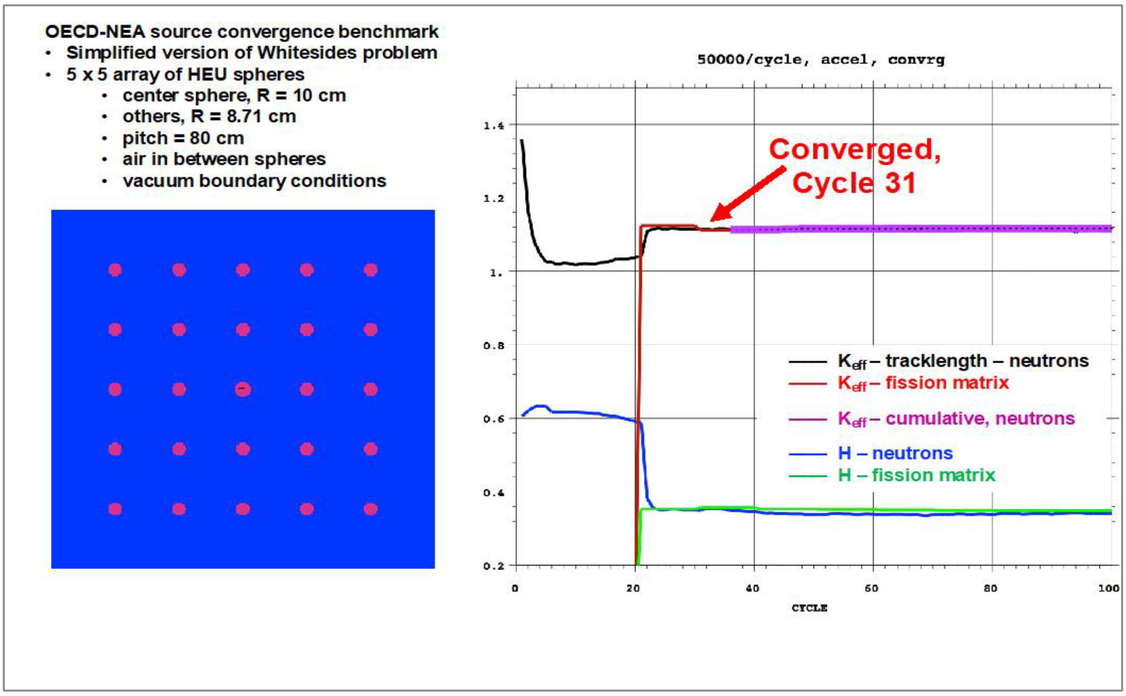
Source: NEA, 2020.

Figure 17. Whitesides "K_{eff} of the World" test problem



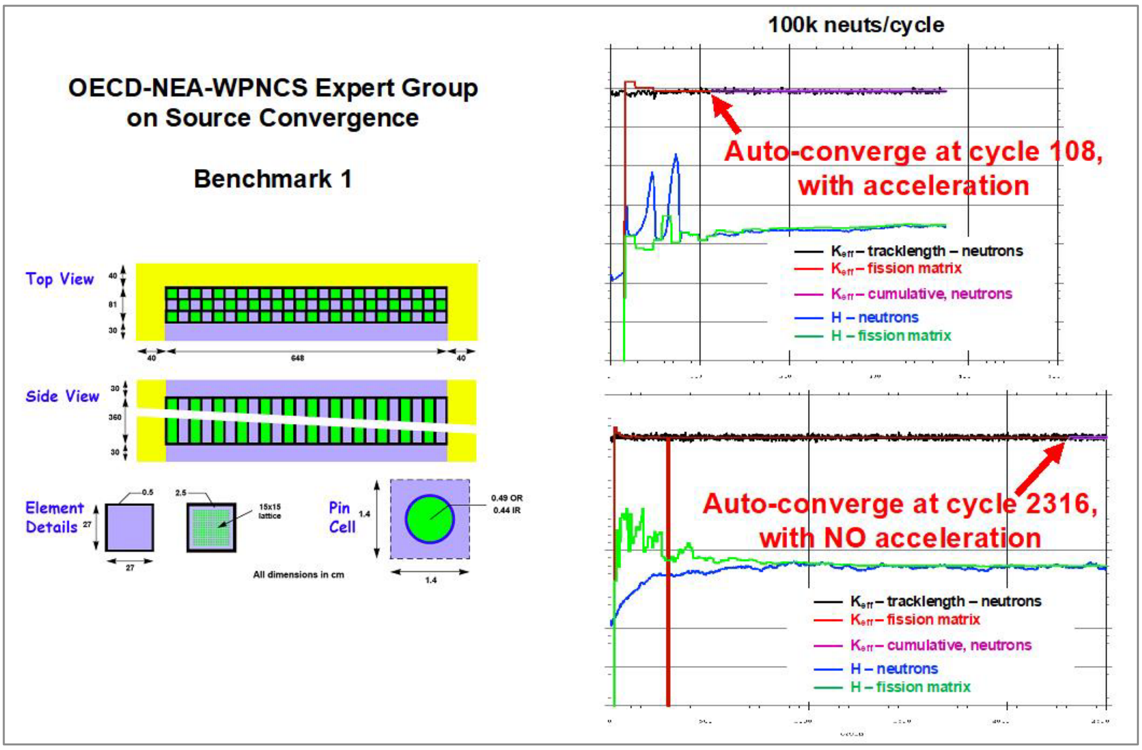
Source: NEA, 2020.

Figure 18. NEA Source Convergence Problem *test4s* test problem



Source: NEA, 2020.

Figure 19. NEA WPNCS Expert Group on Source Convergence, Benchmark 1



Source: NEA, 2020.

6. Additional statistical tests for convergence analysis

In this section, some additional statistical tests are briefly discussed. These tests were examined individually for their effectiveness, and it is highly likely that some or all of these proposed tests could be included with the larger suite of tests described in preceding portions of this report. As noted several times, no single test can provide overwhelming evidence of convergence, but a large collection of tests can; expanding the collection of tests is a general goal of SG-6.

Because the slope-test described above can be applied to any metric that is tallied during the cycles in a block, there are many additional test possibilities:

- 1st, 2nd, and possibly higher moments of the fission neutron distribution in x, y and z. Such moments are easy to compute, and convergence of these moments may be an indicator of overall convergence of the fission neutron distribution. Higher moments may, however, be noisy and less reliable.
- Nowak (2016) has investigated the use of a generalised entropy function that explicitly includes spatial information:

$$S^*_{u,v,w} = -\sum_{i,j,k} \mathcal{L}_u(\xi_i)\,\mathcal{L}_v(\xi_j)\mathcal{L}_w(\xi_k)\cdot p_{i,j,k}\,log_2(p_{i,j,k})$$

  where $\mathcal{L}_q(\xi)$ are the Legendre polynomials (other basis sets could also be used), ξ_i is the x-coordinate of the centre of mesh cell I,j,k normalised to the interval [-1,1]. For polynomials of order u=v=w=0, the regular Shannon entropy function results. Choosing different values for {u,v,w} provides shape-dependent entropy information that could be used as metrics for a slope-test of convergence.
- For loosely-coupled problems, it may be useful to perform cluster analysis of the fission neutron distribution to determine the number of clusters, the cluster size and cluster separation. Slope-tests could be used to detect stability in these quantities.

7. Additional statistical tests investigated using the MONK code

This section provides a summary of some of the statistical tests for convergence investigated using the MONK code.

Mesh-free fission source convergence tests were constructed based on (1) nearest neighbour neutron births in successive cycles, and (2) the differential entropy of the fission neutron distribution. For each test, the pass criterion was a p-value greater than 0.05. Testing with the NEA Source Convergence Benchmark, the ATR, the Special Power Excursion Reactor Test (SPERT) reactor, and 43 MONK validation cases showed very good agreement with reference results for k_{eff}.

7.1. Introduction

In previous sections, it is shown that no individual test for fission source convergence is sufficient to provide a reliable diagnostic of convergence. Rather, a range of tests is required and only when all of the tests pass does it give confidence that the fission source is adequately converged. The aim of the work presented here is to provide two additional candidate tests for fission source convergence. This extends the range of tests available for diagnosing convergence. The tests have been investigated using the MONK® Monte Carlo criticality code (Richards et al., 2019) and initial testing using: the NEA Source Convergence Working Group model, models of the SPERT reactor and ATR, along with 43 of the MONK validation cases, show promising results.

The fission source distribution in a Monte Carlo criticality calculation consists of a set of points in a three-dimensional spatial domain, defining the locations of the fission source neutrons. Both tests are based on a technique (a k-d tree) to efficiently identify the nearest neighbour to any of the points in the three-dimensional domain (Bentley, 1975). This is described in Section 7.2.

The test described in Section 7.3 consists of combining the fission source distributions from different inactive cycles and determining whether the nearest neighbour of each point comes from the same cycle or not, in order to construct a test statistic (Schilling, 1986).

In Section 7.4, the differential entropy is described, which is a continuous analogue of the Shannon entropy discussed in the main text (Singh et al., 2003; Kiedrowski and Beyer, 2017). This provides a scalar value derived from the fission source distribution for each cycle of the calculation. Evaluating the differential entropy for each inactive cycle provides a time series with significant stochastic noise. The main text identifies methods for diagnosing stationarity of such sequences, such as when the slope of the series is estimated to be zero, to within statistics. Another potential method for identifying stationarity is presented (Neumann, 1941), providing an alternative diagnostic.

7.2. Finding the nearest neighbour

Given a set of points in a three-dimensional space, the nearest neighbour and differential entropy tests discussed below both require an efficient method for finding the nearest neighbour to a point in the three-dimensional space. One technique for achieving this in an efficient manner is to store the points in a k-d tree (Bentley, 1975).

To build the k-d tree for a set of points in a three-dimensional space, the extent of the space in each dimension is determined. At each branch of the tree, the point space is split through the median point along the dimension, which currently has the greatest extent. This continues recursively for each branch until there is only one point left, which becomes a leaf on the tree. A one-dimensional example of a k-d tree is displayed in Figure 20, with the leaves shown as circles.

Finding the nearest neighbour to a point involves travelling from the root (6.34 in the example shown in Figure 20) down the tree going left or right at each branch depending on which side of the split the point falls, as shown in Figure 21. Once a leaf is reached it is stored as the current nearest neighbour (as long as it is not the same as the chosen point to within a specified tolerance).

The algorithm then retraces its steps back up the tree and at each branch checks to see if the distance from the point to the split along the current dimension is less than that of the point and its current nearest neighbour. If it is, the algorithm travels down the other side of the split checking for leaves that are closer than the current nearest neighbour before continuing back up the tree, as shown in Figure 21. For a set of n points, the time taken to find the nearest neighbour to a specific point is proportional to $\log(n)$ on average.

7.3. Nearest neighbour test

This test relies on the efficient location of nearest neighbour points in three-dimensional space, facilitated by the k-d tree data structure described in Section 7.2.

The nearest neighbour test is based on a statistical test formulated by Schilling (Schilling, 1986). Consider samples of points in k-dimensional space with sizes $n_1, \dots, n_m$ respectively. The test statistic is calculated by combining the samples and calculating the following sum over the $n = \sum_{j=1}^{m} n_j$ points:

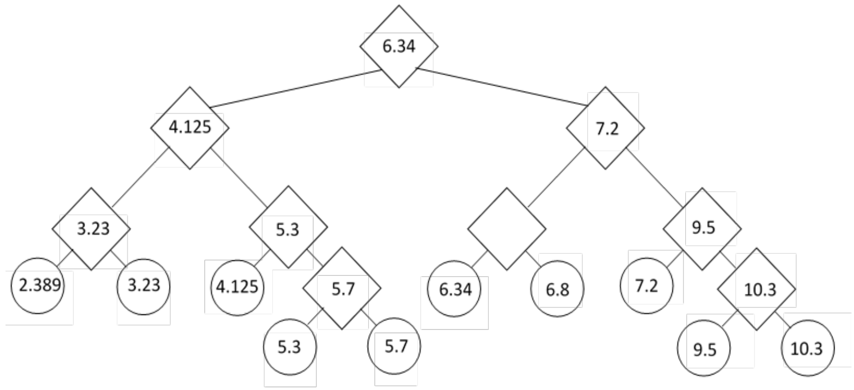
$$T_n = \frac{1}{n}\sum_{i=1}^{n} I_i$$

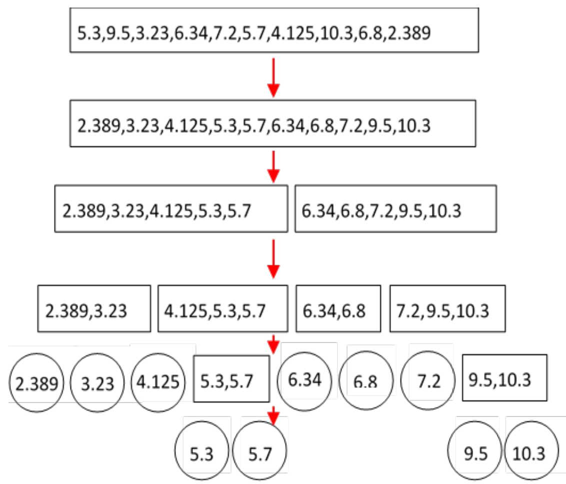
I_i is an indicator function that is equal to 1 when the nearest neighbour of the i$^{\text{th}}$ point belongs to the same sample and is equal to 0 otherwise. For the purposes of diagnosing source convergence, we focus on the particular case of two equal sized samples in three-dimensional space. Through simulation, for this particular case, Schilling found that samples drawn from the same distribution produce statistics that have the following distribution:

$$T_n \sim \sqrt{\frac{0.398}{n}}\mathcal{N} + 0.5$$

where $\mathcal{N}$ is the standard normal distribution, $N(0, 1)$. This can be used to compare birth stores in different cycles and calculate a p-value, as described below.

Figure 20. Example 1-dimensional k-d tree

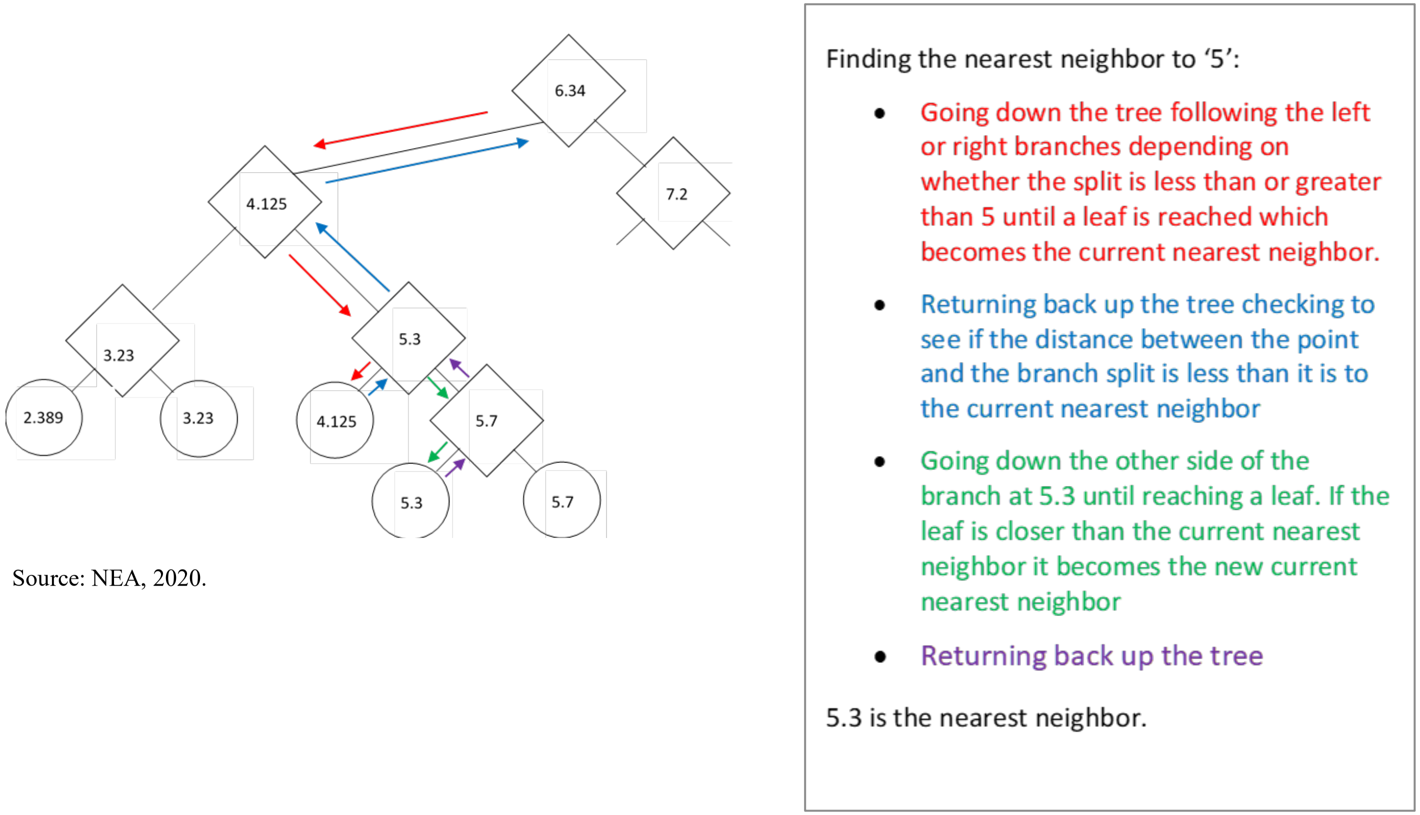




Note: At each branch, values are sorted and divided at the median with the median value always going on the right.

Source: NEA, 2020.

Figure 21. Example nearest neighbour algorithm in a 1-dimensional k-d tree

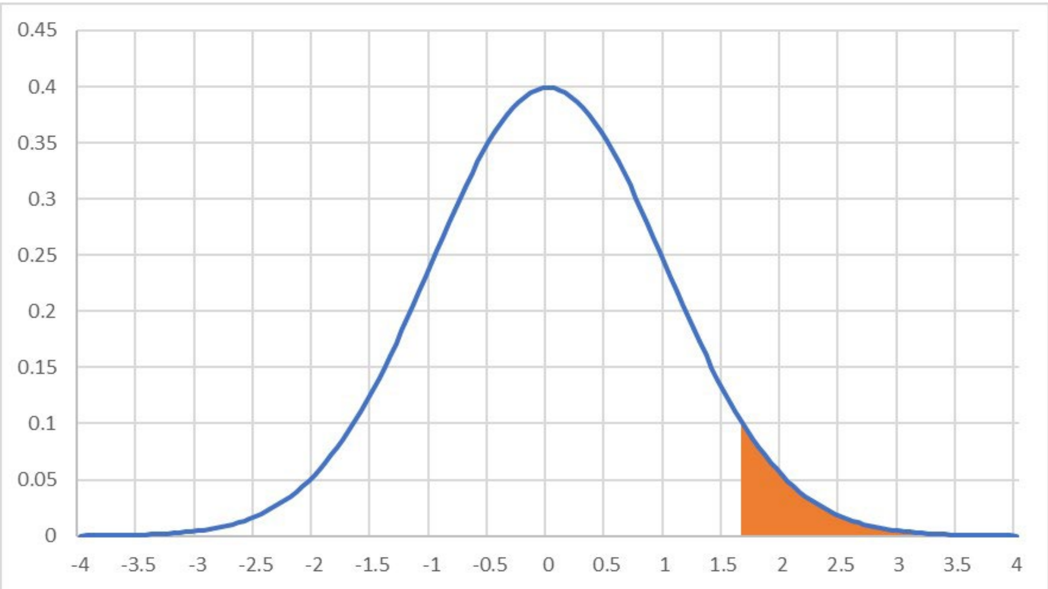


Source: NEA, 2020.

Consider the null hypothesis that two birth stores are samples drawn from the same distribution. The test statistics will follow the distribution above. If the samples differ greatly, the statistic will have a value that is far to the right of the distribution. If the statistic has a value that falls far to the left of the distribution, this could suggest that there is a strong correlation between the samples.

As an example, a right tailed p test is performed with a confidence interval of 95%. Figure 22 shows the standard normal distribution. Only 5% of samples from the distribution are expected to lie in the shaded area of the right tail of the distribution. For two samples both of size 100, this region corresponds approximately to a statistic $T_n > 0.573$ and $p < 0.05$. If a statistic in this shaded region is obtained, then the null hypothesis that both samples are from the same distribution is rejected and the test fails. If it does not, the null hypothesis cannot be rejected and it passes.

Figure 22. For the standard normal distribution, the shaded area corresponds to α=0.05 for a right tailed test



Note: Values to the left of this have a p > 0.05. The x-axis shows standard deviations from the mean.
Source: NEA, 2020.

The test can be used to compare the birth store in the current cycle with a number of previous cycles defined by the user. In spite of the use of super history powering in the MONK code (Brissenden and Garlick, 1986; Richards et al., 2019), some correlation between cycles is still seen, particularly for adjacent cycles. This is model-dependent and may be mitigated by increasing the number of generations tracked in a super history. In the testing performed to date, skipping the previous three cycles was found to eliminate the majority of the effect of correlation on the nearest neighbour test.

7.4. Differential entropy

The differential entropy is similar to the Shannon entropy and was proposed by Shannon as a continuous extension. It is defined as (Singh et al., 2003):

$$h = -\int_V p(x)\ln\big(p(x)\big)\,dx$$

Further study has shown that the differential entropy is actually not equivalent to the Shannon entropy in the continuous limit (Cover and Thomas, 2006; Kiedrowski and Beyer, 2017); however, it can still be useful as a source convergence diagnostic. It has the advantage over the Shannon entropy of not requiring a mesh to be overlaid on the model. This removes the responsibility of the user to determine the optimal mesh dimensions for a given model.

Implementing the differential entropy in its explicit form is non-trivial. Fortunately, it can be estimated in three-dimensional space, using an approximation to the differential entropy (Kiedrowski and Beyer, 2017):

$$\hat{h} = \frac{3}{N}\sum_{i=1}^{N}\ln(\rho_i) + \ln(N) + \ln\left(\frac{4\pi}{3}\right) + \gamma$$

where ρ_i is the distance between the i^{th} point and its nearest neighbour, N is the number of points and $\gamma \simeq 0.5772156649$ is the Euler-Mascheroni constant.

This lends itself well to the way that the positions of the neutron births are kept in the birth store. The computationally difficult step is to find the nearest neighbour distances, which can be achieved using a k-d tree as described in Section 7.2.

It remains to be determined when the resulting time series of differential entropy values has settled down to its asymptotic value. That is, a method is required to diagnose stationarity of a noisy time series. A method for doing this is discussed in Section 7.5.

7.5. Testing for stationarity of a noisy time series

A test is required to determine when a time series of data subject to stochastic fluctuations, such as those often produced by cycle-wise source convergence diagnostics, has reached a steady state. At this point, any remaining variation should be due to stochastic fluctuations. A method to test this is proposed based on work by John von Neumann using the relationship between the mean squared successive difference and the variance of a sequence of observations (Neumann, 1941).

The mean squared successive difference (mssd) of a data set $x_1, \dots, x_n$ is defined as:

$$\delta^2 = \frac{1}{n-1}\sum_{i=1}^{n-1}(x_{i+1} - x_i)^2$$

It has been found that the mssd divided by 2 is a good estimate of the variance of a random sequence in the absence of autocorrelation.

Consider drawing pairs of samples $u_1, \dots, u_n$ and $v_1, \dots, v_m$ from the same normal distribution, with variances ${\sigma_u}^2$ and ${\sigma_v}^2$, respectively. The ratio of these variances follows the F-distribution (Bevington and Robinson, 1992):

$$\frac{{\sigma_u}^2}{{\sigma_v}^2} \sim f(x; d_u, d_v) \quad \text{where} \quad d_u = n - 1 \quad \text{and} \quad d_v = m - 1$$

This forms the basis of the F-test. For equality of variance, the two-tailed test should be performed.

Thus, a stationarity test is constructed by taking the ratio of the mssd, δ^2, and the variance, σ^2, as follows:

$$\phi = max\left\{\frac{2\sigma^2}{\delta^2}, \frac{\delta^2}{2\sigma^2}\right\}$$

It is easier to calculate the p-value for the right tail of the F-distribution than the left so a right tailed test is always performed by having the larger variance in the numerator. The two-tailed p-value is then obtained by multiplying by two. Two batches of size of ten both have nine degrees of freedom so the two-tailed p-value can be determined using the cumulative F-distribution as follows:

$$p = 2\big(1 - F(\phi; 9{,}9)\big)$$

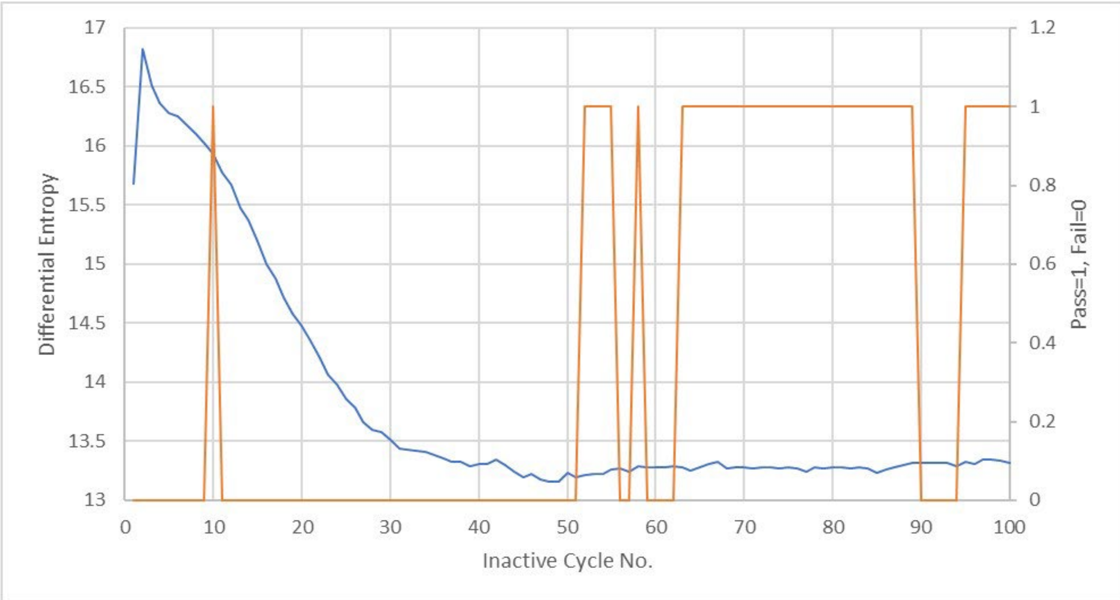
Since a batch of data points is required, the batch size will determine the minimum number of cycles of the Monte Carlo calculation that need to be performed before stationarity of a source convergence diagnostic can be tested. A batch size of ten was found to provide a good compromise between the accuracy of the test and the speed of determining stationarity.

Limited testing has shown this method to be reasonably effective; however, there is a risk of false positives occurring, particularly when there is a turning point in the series that occurs on a length scale that is smaller or similar to the batch size. In Figure 23, this is seen to occur in the first batch of ten cycles and also for batches between cycles 50 and 60.

This issue is largely mitigated by the use of multiple diagnostics (such as those described in the previous sections) where the risk of coincidental false positives is low. False positives could be further mitigated by increasing the threshold of the p-value for passing or by requiring multiple successive passes. However, this could result in an overly conservative estimate of the cycle where the source has converged, particularly if many diagnostics are being used.

The batch size can be varied. A smaller batch size is more sensitive to changes on a smaller length scale; however, there is a degree of autocorrelation between nearby cycles. Since the test relies on the assumption of no autocorrelation for the null hypothesis, the false negative rate might be too high for a smaller batch size.

Figure 23. Plot of the differential entropy diagnostic overlaid with the result of the F-test with a passing threshold set to p>0.05



Note: 1 is a pass and 0 is a fail.
Source: NEA, 2020.

8. Conclusion

The use of many statistical checks for convergence has proven to be robust and reliable. Reliably and automatically determining convergence can save computer time, but more importantly provides solid quantitative evidence that convergence was achieved. That quantitative evidence is important in today's regulatory environment – "eyeballing the plots" is difficult to defend, whereas documented evidence of passing 11 or more statistical tests is clear-cut.

SG-6 was established to exchange technical information on statistical testing to determine convergence of the MC power iteration process. Techniques presented in this report represent the current state of the art for such testing, but are still a work in progress. It is likely that further discussion will lead to the addition of more tests and different types of tests. The ultimate goal is a collection of statistical tests for convergence that provides overwhelming evidence of convergence, a "guarantee" that convergence was achieved.

In the future, NCS practitioners should not have to make trial runs or plot Shannon entropy. Common user pitfalls and annoyances are removed by the automated convergence and statistical testing methods, and quantitative evidence of convergence is provided. Further work in progress includes the automated generation of the initial guess for the fission neutron locations.

References

Bentley, J.L. (1975), "Multidimensional binary search trees used for associative searching", *Communications of the Association for Computing Machinery*, vol. 18, no. 9, pp. 509-517.

Bevington, P.R. and D.K. Robinson (1992*), Data Reduction and Error Analysis for the Physical Sciences* (2nd ed), McGraw-Hill.

Brissenden, R.J. and A.R. Garlick (1986), "Biases in the estimation of Keff and its error by Monte Carlo methods", *Ann. Nucl. Energy*, Vol. 13, No. 2, pp. 63-83.

Brown, F.B. (2019), "Automated Acceleration & Convergence Testing for Monte Carlo Criticality Calculations", LA-UR-19-20984 in DOE NCSP Technical Program Review, 2019-03-26/2019-03-27, Amarillo, Texas, United States.

Brown, F.B. (2017), "Investigation of clustering in MCNP6 Monte Carlo criticality calculations", Int. Conf. on Transport Theory, Monterey CA, Oct 2017, [also LANL report LA-UR-17-29261].

Brown, F.B. (2016), "Monte Carlo Techniques for Nuclear Systems", LANL report LA-UR-16-29043.

Brown, F.B. (2009), "A Review of Best Practices for Monte Carlo Criticality Calculations", ANS NCSD 2009, Hanford WA, United States [also LANL report LA-UR-09-03136].

Brown, F.B. (2006), "On the use of Shannon entropy of the fission distribution for assessing convergence of Monte Carlo criticality calculations", proceedings PHYSOR-2006, Vancouver, British Columbia, Canada [also LANL reports LA-UR-06-3737 and LA-UR-06-6294].

Brown, F.B., B.R Nease and J. Cheatham (2013), "Convergence testing for MCNP5 Monte Carlo eigenvalue calculations", Mathematics & Computation 2007, Monterey CA, [also LANL report LA-UR-07-1123].

Brown, F.B. and C.J. Josey (2018), "Diagnostics for Undersampling and Clustering in Monte Carlo Criticality Calculations", LANL report LA-UR-18-27656.

Brown, F.B. and W.R. Martin (2018), "Statistical Tests for Convergence in Monte Carlo Criticality Calculations", LANL report LA-UR-18-28764.

Brown, F.B., C.J. Josey, S. Henderson and W.R. Martin (2019), "Automated acceleration and convergence testing for Monte Carlo criticality calculations", ICNC 2019, Paris, France.

Brown, F.B., S.E. Carney, B.C. Kiedrowski and W.R. Martin (2013), "Fission-matrix capability for MCNP, Part I - Theory", Mathematics & Computation 2013, Sun Valley, ID, [also LANL report LA-UR-13-20429].

Carney, S.E., F.B. Brown, B.C. Kiedrowski and W.R. Martin (2013), "Fission-matrix capability for MCNP, Part II - Applications", Mathematics & Computation 2013, Sun Valley, ID, [also LANL report LA-UR-13-20454].

Cover, T.M. and J.A. Thomas (2006), *Elements of Information Theory* (2nd Ed), Wiley-Interscience.

Dumonteil, E., F. Malvagi, A. Zoia, A. Mazzolo, D. Artusio, C. Dieudonne and C. de Mulatier (2014), "Particle clustering in Monte Carlo criticality simulations", *Ann. Nucl. Energy*, Vol. 63, pp. 612-618, Elsevier, DOI:10.1016/j.anucene.2013.09.008.

Gelbard, E.M. and R.E. Prael (1974), "Monte Carlo work at Argonne National Laboratory", in Proc. NEACRP Meeting of a Monte Carlo Study Group, ANL-75-2, Argonne National Laboratory, Argonne, United States.

Goad, W. and R. Johnston (1959), "A Monte Carlo method for criticality problems", *Nucl. Sci. Eng.*, Vol.5, pp. 371-375.

Golub, G.H. and C.F. Van Loan (1989), *Matrix Computations* (2nd Edition), Johns Hopkins University Press.

Hoogenboom, J.E., W.R. Martin and B. Petrovic (2011), "The Monte Carlo performance benchmark test-aims, specifications and first results", International Conference on Mathematics and Computational Methods Applied to (Vol. 2, p. 15).

Kaplan, E.L. (1958), "Monte Carlo Methods for Equilibrium Solutions in Neutron Multiplication", Lawrence Radiation Laboratory, UCRL-5275-T.

Kiedrowski, B.C. and K.A. Beyer (2017), "Monte Carlo fission source convergence with nearest-neighbor estimates of the differential entropy", in M&C 2017 International Conference on Mathematics & Computational Methods Applied to Nuclear Science & Engineering, Jeju.

McDonald, J.H. (2015), *Handbook of Biological Statistics* (3rd ed.), www.biostathandbook.com.

Morton, K.W. (1956), "Criticality Calculations by Monte Carlo Methods", United Kingdom Atomic Energy Research Establishment, Harwell, Report T/R-1903.

Mosteller, R.D. (2002), "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.

NEA (2019), *International Handbook of Evaluated Criticality Safety Benchmark Experiments*, OECD Publishing, Paris.

NEA (2006), *Source Convergence in Criticality Safety Analyses*, ISBN 92-64-02304-6, OECD Publishing, Paris.

Nease, B.R. and F.B. Brown (2005), "Convergence Tests for the Fission Source Distribution in MCNP5", LANL report LA-UR-05-6226.

Neumann, J. (1941), "Distribution of the ratio of the mean square successive difference to the variance", *Annals of Mathematical Statistics*, vol. 12, no. 4, pp. 367-395.

Nowak, M., J. Miao, E. Dumonteil, B. Forget, A. Onillon, K.S. Smith and A. Zoia (2016), "Monte Carlo power iteration: Entropy and spatial correlations", *Annals of Nuclear Energy*, Vol. 94, No. 856-868, Elsevier, DOI:10.1016/j.anucene.2016.05.002.

Perfetti, C.M., B.T. Rearden, W.J. Marshall (2017), "Diagnosing undersampling biases in Monte Carlo eigenvalue and flux tally estimates", *Nucl. Sci. Eng.*, Vol.185, pp.139-158.

Richards, S., G. Dobson, T. Fry, D. Hanlon, D. Long, R. Perry, P. Smith, F. Tantillo and T. Ware (2019), "Recent developments to the Monk Monte Carlo code for criticality safety and reactor physics analyses", Proc. ICNC 2019 - 11th International Conference on Nuclear Criticality Safety, 15-20 September 2019, Paris, France.

Schilling, M.F. (1986), "Multivariate two-sample tests based on nearest neighbors", *Journal of the American Statistical Association*, vol. 81, no. 395, pp. 799-806.

Singh, H., N. Misra, V. Hnizdo, A. Fedorowicz, E. Demchuk (2003), "Nearest neighbor estimates of entropy", *American Journal of Mathematical and Management Sciences*, Vol. 23, no. 3-4, pp. 301-321.

Ueki, T., F.B. Brown (2002), "Stationarity and source convergence in Monte Carlo criticality calculations", ANS Topical Meeting on Mathematics & Computation, Gatlinburg, TN, 6-11 April 2003 [also, LANL report LA-UR-02-6228], September 2002.

Ueki, T. and F.B. Brown (2005), "Stationarity modeling and informatics-based diagnostics in Monte Carlo criticality calculations", *Nucl. Sci. Eng.*, Vol. 149, pp. 38-50 [also LANL report LA-UR-03-8343].