M&C 2017 - International Conference on Mathematics & Computational Methods Applied to Nuclear Science & Engineering, Jeju, Korea, April 16-20, 2017, on USB (2017)

Monte Carlo Fission Source Convergence with Nearest-Neighbor Estimates of the Differential Entropy

Brian C. Kiedrowski, Kyle A. Beyer

Department of Nuclear Engineering and Radiological Sciences, University of Michigan, Ann Arbor, MI
bckiedro@umich.edu, beykyle@umich.edu

Abstract - *An algorithm to compute differential entropy of the fission source distribution in Monte Carlo k-eigenvalue calculations was developed and implemented into a research version of MCNP6®. Calculating the differential entropy involves finding the distance between each fission source location and its nearest neighbor. For a batch size of N, a simplistic algorithm would require $O(N^2)$ evaluations, which is computationally prohibitive. The efficiency is significantly improved using a search grid and near linear scaling is observed for test problems. The differential entropy and the Shannon entropy, which is a commonly used statistic for diagnosing fission source convergence, show similar convergence behavior if the user-selected mesh for the Shannon entropy is appropriate. Because of this, the differential entropy may be used in place of the Shannon entropy for fission source convergence with the added benefit of not requiring a user-specified mesh..*

I. INTRODUCTION

The Shannon entropy is one common approach to diagnose convergence of the fission source in Monte Carlo *k*-eigenvalue calculations [1] and is employed by codes including MCNP, Serpent, OpenMC, and KENO. The Shannon entropy requires the use of a mesh discretization superimposed upon the geometry. Obtaining a reliable estimate of the convergence of the fission source with the Shannon entropy requires that the selected mesh be reasonable. A mesh that is too coarse will not capture the changes in the fission source between iterations, leading to a poor estimate of source convergence. Conversely, a mesh that is too fine requires a very large number of particles per iteration to adequately sample the mesh; otherwise the presence of stochastic noise in the Shannon entropy makes detecting convergence difficult. Furthermore, some problems involving distributed regions of fissionable material may not be conducive to a regular mesh.

The differential entropy is the continuous analog of the Shannon entropy. It may be estimated without the requirement of discretization by finding the distances between the source points and their corresponding nearest neighbors. This work introduces the nearest-neighbor estimate of the differential entropy applied to fission source convergence. The method is implemented in a research version of MCNP6®[1] [2]. The results show that the differential entropy has very similar convergence behavior of the Shannon entropy with a good mesh and can be computed efficiently with near linear scaling on the batch size if a search grid is employed (if a simplistic approach is used, the scaling is quadratic, which is computationally prohibitive). The differential entropy, therefore, may be used in place of the Shannon entropy, thereby removing the requirement for a superimposed mesh.

II. THEORY

The Shannon entropy is defined by discretizing the problem space into M regions. In each region j, the fraction of fission source neutrons in that region, p_j, is determined, and the Shannon entropy is estimated by

$$H = -\sum_{j=1}^{M} p_j \ln p_j. \tag{1}$$

Here the natural logarithm is used throughout the paper, but the base of the logarithm is arbitrary. To assess convergence, the Shannon entropy is computed each iteration, and the convergence of the Shannon entropy is used as a surrogate for the convergence of the fission source.

The differential entropy is defined by the volume integral over the entire problem domain

$$h = -\int_V p(\boldsymbol{x}) \ln p(\boldsymbol{x})\, dV, \tag{2}$$

where $p(\boldsymbol{x})$ is the normalized (and non-dimensionalized) continuous fission neutron source density. Note that the differential entropy is not the infinitesimal limit of the Shannon entropy, which leads it to be unbounded and not a measure of uncertainty about a distribution. The two however, can be related in the the limit of a fine, uniform mesh for the Shannon entropy, where the difference between the Shannon and differential entropy is the logarithm of the mesh volume (see Sec. 8.3 in Ref. [3]). Therefore, the differential entropy should exhibit a similar convergence trend, albeit to a different value, to the Shannon entropy with an sufficiently fine mesh discretization.

It has been shown by authors in the statistical literature [4] that the differential entropy in D-dimensional space may be estimated by a nearest-neighbor estimate using a finite number of random sample points:

$$\hat{h} = \frac{D}{N}\sum_{i=1}^{N} \ln \rho_i + \ln N + \ln\left[\frac{\pi^{D/2}}{\Gamma\left(1+\frac{D}{2}\right)}\right] + \gamma. \tag{3}$$

Here N denotes the number of source neutrons in the current iteration, ρ_i is the distance between the ith neutron and its nearest neighbor (excluding ones at the same location because of the fission process), $\Gamma(x)$ is the gamma function, and γ is the Euler constant $\approx$ 0.5772. The third term is the logarithm of the volume of a D-dimensional unit sphere.

---

[1] MCNP® and Monte Carlo N-Particle® are registered trademarks owned by Los Alamos National Security, LLC, manager and operator of Los Alamos National Laboratory.

III. IMPLEMENTATION

A Fortran module computing the differential entropy was written and inserted into MCNP6.1.1, and is optionally called in place of the existing Shannon entropy calculation. Other minor modifications were made to the input parsing routines and common blocks for handling of restart files. The modifications to the source code were designed to be non-obtrusive so that they could be ported to other Monte Carlo software packages.

1. Nearest-Neighbor Search

The computation of the differential entropy requires a nearest-neighbor search, which is the most time consuming portion of the method. An exhaustive approach of calculating the distance between each source point and all other points and taking the minimum requires $O(N^2)$ function evaluations, which is unacceptable for recommended batch sizes of 100k or more found in production calculations. It is therefore necessary to develop a more efficient algorithm that scales acceptably. A simple approach to improve the efficiency that was adopted employs a structured, regular search grid. The user may specify the search grid, otherwise it is determined automatically. Note that, unlike the Shannon entropy, the search grid does not affect the result of the differential entropy, merely the efficiency of calculating it.

A. Automated search grid

The automatic determination of the search grid begins by determining the minimum and maximum coordinates in each cardinal (x, y, z) direction. This information is used to emplace an axis-aligned bounding box, which serves as the extents of the search grid. The extents are 10% larger than the most distant points in each cardinal direction to allow room for growth in subsequent iterations. Additionally, if any source points lie outside the user-specified bounds of the search grid, then the extents of the grid are expanded to encompass all fission source points.

Once the bounding box has been determined, a uniform mesh spacing in each cardinal direction is determined based upon the batch size. An appropriate guess for the number of elements in each cardinal direction is made based on empirical studies, and is 0.25 times the batch size to the $1/3$ power in each dimension, which is motivated by the fact that in 3D, the product of the number of mesh elements scales linearly with the batch size. To ensure the search grid is neither too coarse nor too fine, which limits performance of the search algorithm, the automatic mesh must have at least 5 or at most 100 elements in each cardinal direction.

The percentage of the CPU time required for the differential entropy routines is estimated during the calculation, and if its moving average over the last few cycles (currently 5) exceeds 5%, the search grid is refined along all valid directions. This is currently done by increasing the number of mesh spacing by 2 elements in 3D, 16 elements in 2D, and 256 elements in 1D; this was shown to be effective using empirical studies. The moving average is then reset so that refinement is not done too frequently. The algorithm will continue to refine based on the entropy CPU time percentage moving average as needed to be below 5% or until either the search grid size exceeds a specified limit (currently 10^6) or further refinement does not lead to any improvement. Further improvements or changes in the future are anticipated.

B. Search algorithm

At the end of the iteration, the algorithm first determines the search grid element of each fission point, building a data structure that lists the fission source points within each grid elements. As stated before, if any point lies outside the search grid, the entire grid is reconstructed and the process restarts with the new grid. When duplicate fission source points at the same location are encountered, a new entry is not added to the data structure. Rather, a counter is incremented that indicates the number of fission source sites at this location, and the contribution of that fission source point to the calculation of the differential entropy is multiplied by the number of points indicated. The reason for doing this simplifies the algorithm to automatically exclude fission source points at the same location from the nearest-neighbor distance calculation.

Additionally, to ensure the algorithm remains efficient, there is a maximum number of points allowed within each search grid element, currently 5000. If this number is exceeded, the mesh will be refined as discussed previously on the next iteration. While this limit does bias the differential entropy and can therefore, in principle, interfere with the convergence diagnosis, this limit rarely comes into play in practice except for the first few iterations when the initial source guess is a point source with a large batch size; it is for this situation that this limit was instituted, as without it, the time required to compute the differential entropy requires approximately N^2 calculations and is computationally prohibitive. In the problems tested, this situation rectifies itself after a few iterations as the source distribution spreads out toward its stationary distribution.

After the search grid data structure is built, the distance between each fission source point and its nearest neighbor is computed. For each fission point, the distance to every other point in its grid element is computed. Once the minimum distance is known, a check is made in each cardinal direction if it is possible for a point in each adjacent element to have a smaller distance. If so, then the search is expanded along those cardinal directions to encompass valid neighboring grid elements. This search continues to expand until it is impossible for any neighboring element to contain a point that has a smaller distance, and the nearest-neighbor distance is accepted and scored.

To improve the efficiency, the nearest-neighbor search is implemented to take advantage of OpenMP threading. (The neutron transport routines in MCNP6 already take advantage of threading, so the infrastructure for implementing this is already in place.)

IV. RESULTS

Results are shown for three cases: a 1-D uniform slab; the "K-Effective of the World" problem [5]; and the OECD/NEA source convergence benchmark 1 [6], a checkerboard of storage assemblies that herein is referred to as the fuel storage pool source convergence benchmark. Simulations were run on a Mac OS X computer with a 2.5 GHz Intel Core i7 processor having 16 GB RAM while using four OpenMP threads compiled with gfortran. The 1-D slab has an analytic diffusion theory solution for the differential entropy and shows the differential entropy is being computed correctly. All cases compare the differential entropy with the Shannon entropy on a well-chosen discrete mesh and show that the two entropies show similar source convergence behavior. Computation performance results are also presented, and they show that the differential entropy calculation scales favorably for large batch sizes and takes a few to several percent of the total computational time.

1. 1-D Uniform Slab

To ensure the modified version of MCNP6 is calculating the correct differential entropy, a simple test case of a 1-D uniform slab with width L mean-free paths centered about the origin is run. Fictitious nuclear data is used: $\Sigma_f = 0.2\ \text{cm}^{-1}$, $\Sigma_c = 0.3\ \text{cm}^{-1}$, $\Sigma_s = 0.5\ \text{cm}^{-1}$, $\Sigma_t = 1.0\ \text{cm}^{-1}$, $\nu = 2.5$. If diffusion theory and zero-flux boundary conditions are valid (the slab is not too strongly absorbing and is optically thick), then the fission neutron probability density is approximately

$$p(x) \approx \frac{\pi}{2L} \cos\left(\frac{\pi x}{L}\right), \quad -L/2 < x < L/2. \tag{4}$$

The differential entropy is

$$\begin{aligned} h &= -\int_{-L/2}^{L/2} \frac{\pi}{2L} \cos\left(\frac{\pi x}{L}\right) \ln\left[\frac{\pi}{2L} \cos\left(\frac{\pi x}{L}\right)\right] dx \\ &= 1 - \ln\left(\frac{\pi}{L}\right). \end{aligned} \tag{5}$$

For a slab of width $L = 100$, $h \approx 4.46$. Two source guesses are used: (i) a point source at the origin and (ii) a uniform source. The corresponding differential entropies of these are distributions are $h = -\infty$ and $h = \ln L \approx 4.61$ respectively.

The problem was run for 2000 iterations each having 20000 particles on average. Figure 1 shows the differential entropy as a function of iteration for the point and uniform source guesses. Both converge to around the expected result. The average differential entropy of the last 1000 iterations is 4.43 for the point source guess and 4.45 for the uniform source guess, which compare favorably with the reference result of 4.46.

The differential entropy is compared with the Shannon entropy on a discretized mesh with 20 spatial zones, such that each element is 5 mean-free paths thick. Figure 2 compares the differential and Shannon entropies. The Shannon entropy is scaled by an additive constant to put the curve on the same scale. Since both estimates used the same histories, the two curves show very similar trends, predicting convergence at around 500 iterations and having a Pearson correlation coefficient of 0.94 in the last 1000 iterations. The differential entropy does has a higher variance, but this is a consequence of the coarse mesh spacing possible in this simple problem.

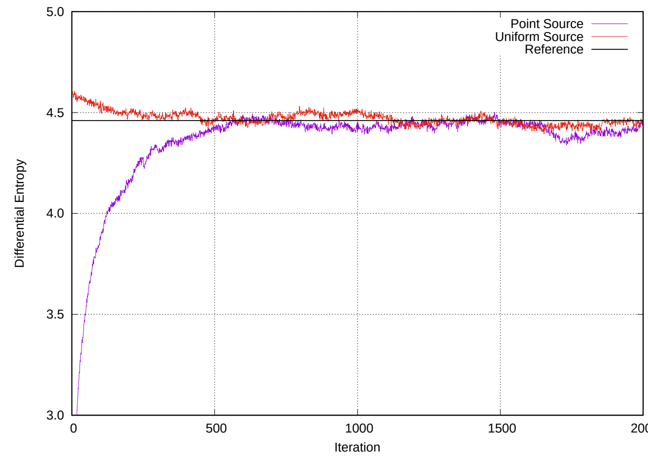

Fig. 1. Convergence of the differential entropy for the 1-D slab to the reference result.

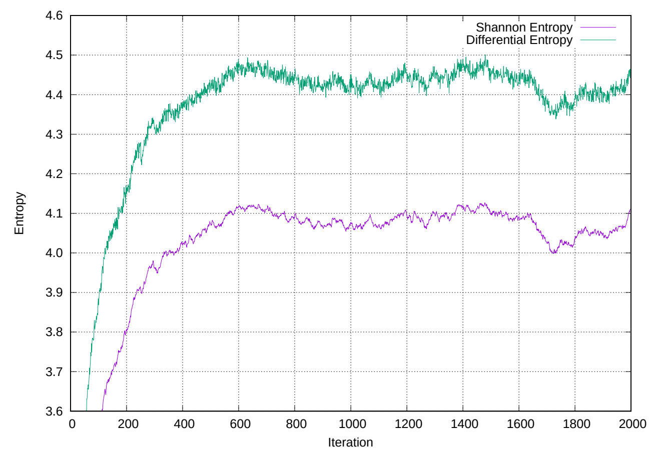

Fig. 2. Convergence of the differential and Shannon entropy for the 1-D slab.

2. K-Effective of the World

A variant of the K-Effective of the World problem was tested. This problem was specifically designed by Whitesides in 1971 to stress Monte Carlo criticality calculations with the computational capabilities at the time and still remains a good test problem because of its convergence behavior. The problem is a 30-cm pitch, $9 \times 9 \times 9$ array of subcritical Pu-239 spheres ($r = 4$ cm) in vacuum with the exception of the center sphere, which is supercritical ($r = 5$ cm). The array is surrounded by a neutronically infinite region of water. The source guess consists of uniformly selected point sources at the centers of each sphere. The true fission distribution, however, is peaked around the central, more reactive sphere. This problem was difficult because it is possible, with the small batch sizes of hundreds that were typical at the time, for the neutrons to entirely miss the central, most reactive sphere and therefore significantly underestimate the k eigenvalue.

The problem was run for 2000 iterations each having 20000 particles on average. This batch size ensures that con-

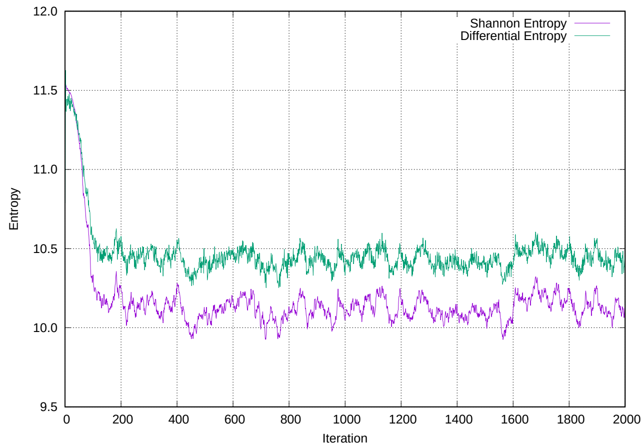

Fig. 3. Convergence of the differential and Shannon entropy for the K-Effective of the World problem.

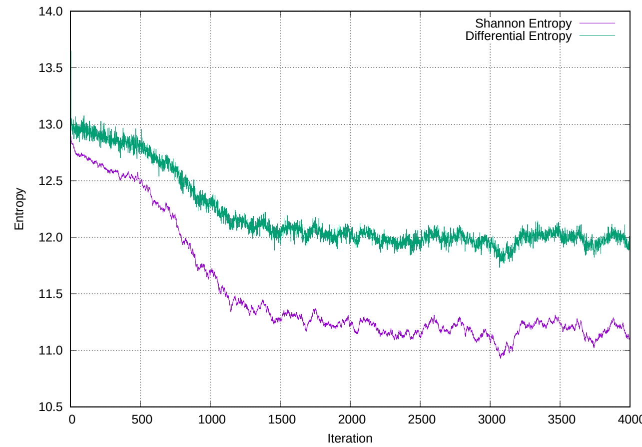

Fig. 4. Convergence of the differential and Shannon entropy for the fuel storage pool source convergence benchmark.

vergence from the near uniform source guess to the true fission distribution peaked around the central sphere is reliable.

Figure 3 shows the Shannon entropy (scaled by an additive offset) and the differential entropy. The Shannon entropy mesh is a uniform $9 \times 9 \times 9$ grid that covers the array of spheres. The two measures show similar trends, having a Pearson correlation of 0.91 in the last 1000 iterations. Both of them predict convergence at about 100-150 iterations.

3. Fuel Storage Pool Source Convergence Benchmark

The fuel storage pool source convergence benchmark was designed to stress the convergence of the fission source, having an extremely high dominance ratio that is nearly one. The problem is geometrically large consisting of a checkerboard of LEU fuel assemblies in water of fuel that are nearly neutronically decoupled. The checkerboard is 24×3 with each having a side length of 27 cm. The fuel height is 360 cm with 30 cm of water on top and bottom. Surrounding the checkerboard on the top, left, and right sides is 40 cm of concrete; the bottom side is water. The top-left element is adjacent to the concrete and has fuel that is reflected on both sides, whereas the the top-right corner has no fuel. This leads to a large tilt in the fission rate between the left and right sides that is several orders of magnitude. The initial source guess is a uniform box surrounding the active fuel region, which is a poor guess for this problem and leads to a long convergence time on the order of thousands of cycles.

The Shannon entropy mesh is $96 \times 24 \times 6$, covering the fuel-containing portion of the problem and ensures there is a 4×4 grid in each assembly zone, each having 6 axial zones. The problem was run 4000 iterations each having 20000 particles on average. Figure 4 compares the Shannon entropy (scaled by an additive offset) and the differential entropy. Both curves show that the entropy of the source takes over 1500 iterations to converge. Note that the chosen batch size is too small to sample the entire problem given the extreme tilt in the fission density shape, but the point is to show that the two measures give similar information. As before, the differential entropy has more statistical noise than the Shannon entropy from a well-chosen discrete mesh. The correlation coefficient for the last 2000 iterations is about 0.80; while not as high as the previous cases, the two entropy measures are still significantly correlated.

4. Computational Performance

The nearest-neighbor search algorithm may take a significant amount of computational effort. The measure of computational cost is the ratio of the computational time taken to compute the differential entropy to the total computational time. To help quantify this, a uniform, one-speed $20 \times 20 \times 20$ mean-free path cube (same nuclear data as the 1-D slab) was run for variable batch sizes with 100 inactive and 400 active cycles with a source guess of a product of cosines in x, y, z. No guess for the search mesh is provided, and the algorithm must find one itself.

The results are given in Table I. There appears to be a slight increase in the computational cost for large batch sizes ($\sim 10^6$ or more), but the cost is still on the order of 5% (the threshold used for refining the search grid), indicating it grows much slower than $O(N^2)$, the brute-force nearest-neighbor search complexity. Using one-speed data eliminates the computational cost associated with interpolating cross sections, which is often one of the most costly portions of a neutron transport simulation, so these represent a case where the cost should be proportionately similar or higher. Similar computational costs were obtained for continuous-energy cases (e.g., the fuel storage pool source convergence benchmark) that

Batch Size	Cost (%)	Grid Elements
10^3	4.6	20^3
10^4	5.8	37^3
10^5	4.8	38^3
10^6	6.0	83^3
10^7	7.0	100^3

TABLE I. Computational cost (fraction of CPU time) of computing the differential entropy for a one-speed cube.

indicate costs of a few percent, confirming this assertion.

Note that hese results were obtained with the parameters and methods described previously. Some amount of empirical tuning was done to reach these results, and it is quite likely that further optimization is possible.

V. SUMMARY & FUTURE WORK

The nearest-neighbor calculation of the differential entropy of the fission source was implemented in a research version of MCNP6.1.1. Testing shows that the method accurately computes the differential entropy and that it produces similar source convergence behavior observed with the Shannon entropy given a well-chosen mesh. This demonstrates that the differential entropy may be used in place of the Shannon entropy, and may be especially beneficial in problems where a suitable mesh may be difficult or impossible to find and eliminates a potential source of user error. The absence of a mesh requirement may also facilitate the development of methods to assess appropriate sampling or potentially calculations of the mutual information, which gives information about the fission source.

A significant disadvantage is that the nearest-neighbor algorithm is significantly more resource intensive compared with the straightforward Shannon entropy calculation; however, empirical results show that with the implemented accelerations, the scaling to large batch sizes is favorable with a computational cost of a few to several percent. Further efficiency improvements should be possible, and the requirement that the search mesh be regular needs to be addressed. Additionally, the differential entropy has a higher variance than a well-chosen coarse mesh for the Shannon entropy. A portion of the variance is because of the large negative scores when the distances ρ_i are small. A potential approach that can address this is to base the differential entropy on the Kth nearest neighbor, which limits these large negative contributions.

ACKNOWLEDGMENTS

Partial funding was provided by award NRC-HQ-13-G-38-0007 from the US Nuclear Regulatory Commission. The statements, findings, conclusions, and recommendations are those of the authors and do not necessarily reflect the view of the US Nuclear Regulatory Commission. Partial funding was supported by the Department of Energy National Nuclear Security Administration under Award Number(s) DE-NA0002576, the Consortium for Non-Proliferation Enabling Capabilities.

REFERENCES

1. T. UEKI and F. B. BROWN, "Stationarity Modeling and Informatics-Based Diagnostics in Monte Carlo Criticality Calculations," *Nucl. Sci. Eng.*, **149**, 38–50 (2003).
2. J. T. GOORLEY ET AL., "MCNP6 Initial Release," *Nucl. Technol.*, **180**, 298–315 (2011).
3. T. M. COVER and J. A. THOMAS, *Elements of Information Theory, 2nd Ed.*, John Wiley & Sons, Inc., Hoboken, NJ (2006).
4. H. SINGH ET AL., "Nearest Neighbor Estimates of Entropy," *J. Mathcl. Managmt. Sci.*, **23**, 301–321 (2003).
5. G. E. WHITESIDES, "A Difficulty in Computing the K-Effective of the World," *Trans. Am. Nucl. Soc.*, **14**, 680 (1971).
6. R. N. BLOMQUIST ET AL., "Source Convergence in Criticality Safety Analyses," Tech. Rep. 5431, OECD/NEA (2006).