6-24-2010

Diagnostics of source distribution and particle population in Monte Carlo source iteration methods

Bryan Scott Chapman

Bryan S. Chapman
Candidate

Chemical and Nuclear Engineering
Department

This thesis is approved, and it is acceptable in quality and form for publication:

Approved by the Thesis Committee:

, Chairperson

Diagnostics of Source Distribution and Particle Population in Monte Carlo Source Iteration Methods

BY

Bryan Scott Chapman

B.S., Nuclear Engineering
University of New Mexico, 2008

THESIS

Submitted in Partial Fulfillment of the
Requirements for the Degree of

Master of Science
Nuclear Engineering

The University of New Mexico
Albuquerque, New Mexico

May, 2010

Dedication

I dedicate this work to my fiancé, Dion Lynn Jackson, who has encouraged me throughout this process. Her patience and support has been invaluable.

Acknowledgements

First, I would like to thank Dr. Taro Ueki, my advisor, for giving me the opportunity to pursue graduate research, which allowed me to expand my knowledge and pursue my educational goals. Secondly, I would like to thank my parents for their support and instilling within me the desire to attain a higher education.

Diagnostics of Source Distribution and Particle Population in Monte Carlo Source Iteration Methods

BY

Bryan Scott Chapman

ABSTRACT OF THESIS

Submitted in Partial Fulfillment of the
Requirements for the Degree of

Master of Science
Nuclear Engineering

The University of New Mexico
Albuquerque, New Mexico

May, 2010

Diagnostics of Source Distribution and Particle Population in Monte Carlo Source Iteration Methods

BY

Bryan Scott Chapman

B.S., Nuclear Engineering
University of New Mexico, 2008

M.S., Nuclear Engineering
University of New Mexico, 2010

Abstract

This thesis is concerned with the development of mesh-input-free diagnostics for the determination of the iteration at which the source distribution of a Monte Carlo simulation has reached a stationary state, as well as the sufficiency of particle population size for a given tally cell volume, so as to reduce bias and increase accuracy of estimations of physical properties. Such physical properties can include, but are not limited to, neutron effective multiplication, power distribution, neutron flux and various interaction rates. When the physical properties of a Monte Carlo simulation are accurately estimated, they can be used to predict the actual behavior of a nuclear system, only being limited to the assumptions used to create the model.

Five methods were used to describe the state of the source distribution. Four of the methods were established indicators of the source distribution's state that required the

input of a mesh, which divided the geometry into bins. These indicators are the Shannon entropy, Jensen measure, the progressive relative entropy and the posterior relative entropy. The fifth indicator of the source distribution's state was developed as to eliminate the need for the input of a mesh upon the geometry. This method will be identified as the regionwise average position indicator or *RAPI* and is calculated by taking the sum of the distances of the regionwise average particle positions in the model at each cycle from the corresponding regionwise average particle positions at the first cycle. In conjunction with the Shannon entropy, Jensen measure, progressive relative entropy and the *RAPI*, an on-the-fly step-refined judgment of the indicators of the source distribution's state will be employed to determine at which cycle or iteration the indicators have reached convergence, signifying that the simulated source distribution has begun to fluctuate around the true source distribution. This step-refined on-the-fly diagnostic of the source distribution was developed from the Wilcoxon rank sum in non-parametric statistics. The posterior relative entropy cycle of convergence is determined to be the cycle at which the posterior relative entropy becomes less than the average value of the posterior relative entropy over the second half of active cycles. The cycle of convergence was determined for three different models by the use of the above described methods. The resulting cycle of convergence obtained by the use of the indicators requiring a mesh input was compared against that obtained from the mesh-input-free indicator, *RAPI*, for each of the models. It was found that the *RAPI* was an excellent representation of the source distribution's state and more conservative than the posterior relative entropy diagnosis. The *RAPI* can be used to determine the cycle at which the

source distribution converged to an equilibrium fluctuation range of stationary state, thus eliminating the need for mesh-input for physical property estimation.

Applications of graph theory techniques to Monte Carlo methods were also investigated as a means of meshless convergence indication, but drawbacks for such an application led to a particle population diagnostic investigation. This was done because meshless particle population diagnosis for the power distribution has yet to be done in Monte Carlo source iteration methods. In power distribution calculations, tally cells are used to estimate the power distribution in a model. To approach this problem, the concept of Euclidian minimum spanning trees (EMST) was applied to the source distribution to develop a meshless diagnosis of the particle population. One source particle effect is the characteristic volume of one particle and is defined to be the cubic of the average edge length of an EMST. Then using this characteristic volume, weak and strong requirements of the particle population size were defined for minimum tally cell volume. This diagnostic was compared against a verified population diagnostic, which requires a mesh input, termed as PD-MESH in this thesis. These diagnostic methods were used in the analysis of a pressurized water reactor initial full core simulation. The comparison of the EMST-based population diagnosis to PD-MESH showed that it can be used to determine if a population size is of sufficient size for power distribution calculations, eliminating the need for mesh-based diagnosis.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Monte Carlo (MC) Codes solve criticality problems or simulate nuclear systems by the source iteration method. At the start of this method, or the first cycle, a batch of particles is distributed within the fissile regions of the geometry. These particles and their interactions are tracked until their track terminates or they leak out of the system. The location of the fission events of the particles, determine the spatial distribution of particles in subsequent cycles. The particles interactions are tracked in the next cycle, and this process is repeated until the source distribution converges or begins to fluctuate around the true fundamental mode distribution. When the source distribution has converged, the interactions reaction rates of the particles can be begin to be tallied. These tallies are used to estimate the physical properties of the nuclear model. If the source distribution has not converged before tallying begins or the number of particles used in the simulation is insufficient to adequately model the problem, bias and error are introduced into the physical property estimation. To reduce the bias and error of the tallies, diagnostics must be developed. The first diagnostic is one that can determine if the

source distribution has converged, and the second diagnostic is one that can decide if the number of particles that are being used for Monte Carlo simulations is of sufficient size.

The number of cycles that the source distribution takes to converge is affected by, but not limited to, the dominance ratio (DR) of the problem that is being considered, i.e. the ratio of the first higher mode eigenvalue to the fundamental mode eigenvalue (DR=k_1/k_0). If the dominance ratio is high or close to unity, the source distribution may take thousands of cycles to converge. If the DR is low, the source distribution may converge in less than one-hundred cycles. The convergence of the source distribution can also be affected by the heterogeneity of the system and the number of particles used for the modeling.

To determine when the source distribution has converged to the true fundamental mode, so that more accurate tallying can be performed, Ueki and Brown proposed mesh-based convergence indicators whose use has been demonstrated [1-3]. The indicators proposed were progressive relative entropy, posterior relative entropy and Shannon entropy. Unfortunately these indicators depend upon mesh-input/binning by the user. If the binning is done incorrectly, i.e., not adequately refined, too refined or not placed in the correct locations, these indicators can give false signs of convergence. The posterior relative entropy also depends on the user to specify when the tallying begins or the number of inactive cycles. If the user defines the number of inactive cycles to be too small and the tallying begins before the source has converged (the source does not converge in the middle of active cycles), the posterior relative entropy can also give false indication of convergence. These issues, as well as the theory of the entropy convergence indicators will be discussed in Chapter 2.

To avoid the problems that may arise with user input meshing, a meshless convergence indicator needs to be developed. For this work, a meshless convergence indicator was developed from the idea of automatic meshing. In the automatic meshing, "centers" are created by taking the average positions of the particles that are generated in the geometry. Every particle generated in any geometry is generated within fissile regions of the geometry only, and each particle position has an *x, y* and *z* component. By taking the average position of the particles in a three-dimensional space, in the *x, y* and *z* dimensions, a center coordinate will be found. This center will divide the geometry into eight cells; top-upper-right, top-upper-left, top-lower-right, top-lower-left, bottom-upper-right, bottom-upper-left, bottom-lower-right, and bottom-lower-left. This procedure can be then applied to the particles in these eight cells and find the average position of the particles within these cells. This will create eight more centers or local origins for a total of nine. The automatic meshing can be refined as far as desired. This can be done by taking another averaging of the particle positions. From the eight cells centers, sixty-four cells are created. By taking the average of the particle position in the sixty-four cells, sixty-four more centers or origins are obtained for a total of seventy-three origins. And by taking this process even further, from the sixty-four origins that are determined, five hundred twelve cells are created. By taking the average of the particle positions in these five hundred twelve cells another five hundred twelve centers or origins can be realized. The number of origins that can be determined is dependent on the number of divisions that are performed. This dependency is exhibited by the relationship, $T = \sum_{i=0}^{L-1} 8^i$, where T is the number of origins found and L is the level of discretization or number of divisions to be performed. For example if L is equal to 2, T will be equal to 9 and if L is equal to 3,

T will be equal to 73. From this technique, a method free of predefined meshing for convergence indication can be developed by taking the sum of the distance of the calculated centers at each iteration from the corresponding calculated centers at the first iteration. This indicator of the source state will be known as the regionwise average position indicator (*RAPI*) and will be further discussed in Chapter 3.

In conjunction with the convergence indicator an on-the-fly judgment of the indicator will be needed to create an on-the-fly convergence diagnostic. An on-the-fly convergence diagnostic would track the indicator through the cycles, checking at each cycle to see if the convergence indicator has reached a state of stationarity. When the convergence indicator has reached stationarity, the source distribution matches the true source distribution within small fluctuation and the estimation of physical properties or tallying could begin. Ueki proposed using a Wilcoxon signed rank test on the convergence indicator for determination of the cycle when the source distribution has reached stationarity [3]. The details of this test will be discussed in Chapter 5.

The second diagnostic of concern within this thesis was to determine if the number of particles in a simulation is of sufficient size; a meshless particle population diagnostic will accomplish this task. Euclidian minimum spanning trees (EMST) were investigated as meshless means of drawing diagnostic information from the source particles in Monte Carlo source iterations, and EMST is a concept adopted from graph theory. EMST's were first investigated as a means of convergence indication of the source distribution. By finding the total cost of an EMST at every cycle, a convergence indicator can be developed. But due to limitations found using EMST's in this way, the

main focus for the utilization of EMST's in this thesis is for meshless particle population diagnosis.

EMSTs have been used to compute the various entropies in multidimensional signal processing and image registration [4-6], can be applied to pattern recognition, have been used for multidimensional data analysis [7], and are capable of characterizing the order and disorder of a set of particles in some physics experiments [8]. Because of these applications, it should be possible for EMSTs to be applied to the source particles in Monte Carlo source iterations to draw some diagnostic information. This technique will be used to find an EMST with the vertices being the particle positions within the Monte Carlo simulation geometry.

The population diagnosis will determine what the resolution of the tally cell needs to be for power distribution investigations. It is important to draw some criterion for minimum tally cell volume for power distribution for a given population size, i.e., a given number of histories per cycle. This criterion will be developed from the average edge length of an EMST. It is important because if the tally cell size is incorrect an unreliable power distribution can be inferred from the simulation. The EMST diagnostic results will be compared to previous work by Ueki on population diagnosis with a diagnostic bin mesh, denoted PD-MESH [9]. The use of EMSTs will provide a meshless diagnostic for determining tally cell resolution, which will eliminate the need for prior experience with mesh input to obtain practical and applicable results. The theory behind the EMST method, the constraints of its implementation for convergence indication, and an alternative application for population diagnostics will be discussed in Chapter 4.

In summary, the objective of this thesis is to demonstrate and implement the aforementioned automated indicator, *RAPI*, coupled with an on-the-fly judgment of the indicator in Monte Carlo simulations as well as to investigate the feasibility of population diagnostics derived from EMST's. In Chapter 2 entropy or mesh-based convergence indication will be discussed, and the *RAPI* theory and method will be discussed in Chapter 3. The theory behind the EMST method, the constraints of its implementation for convergence indication, and alternative applications for population diagnostics will be discussed in Chapter 4. The step refined on-the-fly judgment indicator or Wilcoxon rank sum and the theory behind it will be discussed in Chapter 5. Numerical results of the implementation of the *RAPI* method and its comparison against mesh-based convergence determination will be presented and discussed in Chapter 6. The numerical results for the EMST based population diagnostics and its use for defining a criterion for a tally mesh resolution for power distribution of a given population size will be presented and discussed in Chapter 7. Finally, suggestions for future work and conclusions will be presented in Chapters 8 and 9 respectively.

Chapter 2

Entropy Convergence Indicators

In Monte Carlo calculations, the cycle at which the source distribution reaches a stationary state, or matches the true fundamental mode, needs to be determined so that accurate tallying of the physical properties of a Monte Carlo simulation can done. To determine when the source distribution reaches a convergent state an indicator of the source distribution's state is needed. Brown and Ueki proposed and demonstrated the use of convergence indicators that utilized different entropies of the system to determine when the source distribution has reached convergence [1-3]. These convergence indicators are mesh-based and were developed from information theory. Since the entropy convergence indicators were proven to be able to determine when the source distribution has reached convergence, they will be used as comparison for the mesh-input-free convergence indicator proposed in this thesis. The comparison of the mesh-input-free convergence indicator to the entropy convergence indicators will determine if the mesh-input-free indicator can describe the state of the source distribution and if it can be used to determine the cycle at which the source distribution has reached stationarity.

In information theory, entropy is a measure of uncertainty associated with a random variable. In Monte Carlo calculations the source distribution is random. The entropy or measure of uncertainty of the source distribution over the whole geometry at each cycle can be calculated and there are different ways of defining the entropy of a system. By comparing the entropy from one cycle to the next, the change in uncertainty associated with the source distribution can be tracked. The entropy of the system can be used as a tool to indicate source convergence. When the entropy or uncertainty begins to fluctuate around an equilibrium value or converges, the particle source distribution has reached stationarity and fluctuates around the true source distribution.

In Monte Carlo calculations the geometry is divided into bins. Bins are sub volumes of the entire volume, and their size and location are currently predetermined by the user. The initial source distribution is also defined by the user. The initial source distribution is usually randomly distributed within part or whole of the fissile regions of the geometry. By this distribution, the bins that divide the geometry are populated with particles. The initial distribution determines the birth sites of particles in the first iteration. In a cycle or iteration the particle interactions within the simulation are recorded and the particle distribution in the next iteration is defined by nuclear fission interactions. This process is repeated for as many cycles as desired. The part of the source distribution located within these bins can be defined as $S^{(j)}(i)$, where j is the iteration or cycle number and i denotes the i^{th} bin out of B total bins. The source distribution is normalized to one. The properties of the source distribution are explained in equation (2.1).

$$S^{(j)}(i), i = 1,2,...,B; \quad \sum_{i=1}^{B} S^{(j)}(i) = 1 \tag{2.1}$$

Because $S^{(j)}(i)$ is normalized to one, it can be interpreted as a probability. This account is found in equation (2.2).

$$S^{(j)}(i) = \text{"probability of particle starting its history at bin } i\text{"} \tag{2.2}$$

In information theory the information content of an event needs to be measured. In this analysis, event (E_i), is a particle starting its history at bin i. The information content of such an event is calculated by equation (2.3).

$$I(E_i) = \log_2 S^{(j)}(i) = \text{"information content of event } E_i\text{"} \tag{2.3}$$

The source distribution in the first cycle (j=1) is randomly distributed as said before, but the distribution of particles in subsequent cycles is determined from previous cycle fission sites. Since the distribution is originally randomly distributed, the random nature of the distribution can be deferred to the subsequent cycles. Since the distribution is random we can use the definition for entropy that Appelbaum describes [10]. Given a random variable X, not knowing which of its values $x_1, x_2, ..., x_n$ will occur, one really doesn't know how much information $I(x = x_1), I(x = x_2), ..., I(x = x_n)$ will be received but we can regard the information content of the random variable as denoted by **I(X)** [11].

2.1 Shannon Entropy

The mean or expected information content of this random variable is called its entropy, and to obtain the entropy, equation (2.4) can be used:

$$H(X) = E(I(X)) = -\sum_{j=1}^{n} p_j \log_2(p_j) \tag{2.4}$$

where p_j is the probability of the event $\{x = x_j\}$. This entropy is identified as the Shannon Entropy. From the concept of entropy or measure of uncertainty, three more variations can be developed and they are known as Jensen measure, progressive relative entropy and posterior relative entropy. Shannon Entropy is the simplest of the four entropy calculations. By applying the concept of Shannon entropy to the source distribution we arrive at equation (2.5).

$$H(S^{(j)}) = -\sum_{i=1}^{B} S^{(j)}(i) \log_2(S^{(j)}(i)) \tag{2.5}$$

Shannon Entropy is considered to be a concave functional [11]. The simplest example of a concave function is a concave function of one variable and is displayed in Figure 2.1.

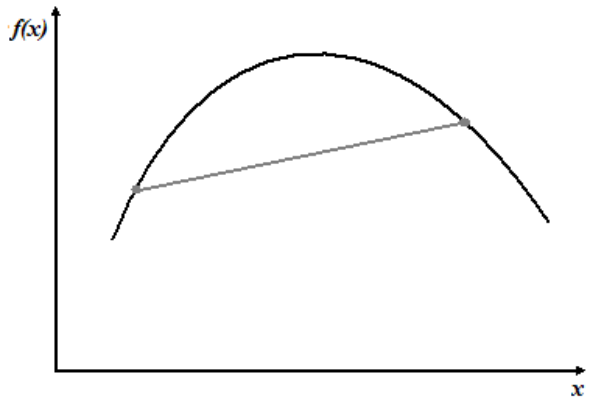


Figure 2.1: Concave Function of One Variable

A function is said to be concave when an arc connecting two points on the function has values that are all less than the values of the function, which can be seen from the preceding graph. When the points in the arc all take values greater than the function, the function is said to be convex. These concepts can be generalized to a functional, i.e., functions of one function. Without creating a graph, the concavity of the functional can be determined. The functional F is considered to be concave if the conditions in equation (2.6) are met.

$$F\left(\lambda f_1+(1-\lambda) f_2\right) \geq \lambda F\left(f_1\right)+(1-\lambda) F\left(f_2\right); \quad 0 \leq \lambda \leq 1 \tag{2.6}$$

In this equation, f_1 and f_2 are two independent random distributions. The variables in equation (2.6) can be changed to suit our investigation. The functional can be reinterpreted to use the Shannon Entropy, and the random distributions can be replaced by the source distributions S and T, which is seen in equation (2.7).

$$H\left(\lambda S+(1-\lambda) T\right) \geq \lambda H\left(S\right)+(1-\lambda) H\left(T\right) \tag{2.7}$$

In equation (2.7) the implicit assumption is that $S = S^{(1)}$ and $T = S^{(j)}$. In other words the random distribution, *S*, is equal to the source distribution defined in equation(2.1), but for only the first cycle. The random distribution, *T*, is equal to the source distribution at cycle *j*. Because preference is given to neither *S* nor *T*, we set λ=0.5. This is done to conserve the concavity of the functional.

2.2 Jensen Measure

By rearranging the equality in equation (2.7) and using these definitions, the description of the second entropy involved in our analysis is arrived at: Jensen measure (Equation (2.8)).

$$\begin{aligned} J\left(S^{(j)}\right) &\equiv H\left(0.5S^{(1)} + 0.5S^{(j)}\right) - 0.5H\left(S^{(1)}\right) - 0.5H\left(S^{(j)}\right) \\ &= -0.5\sum_{i=1}^{B}\left(S^{(1)}(i) + S^{(j)}(i)\right)\log_2\left(0.5\times\left(S^{(1)}(i) + S^{(j)}(i)\right)\right) \\ &\quad + 0.5\sum_{i=1}^{B} S^{(1)}(i)\log_2\left(S^{(1)}(i)\right) + 0.5\sum_{i=1}^{B} S^{(j)}(i)\log_2\left(S^{(j)}(i)\right) \end{aligned} \tag{2.8}$$

Jensen measure is a metric derived from the concavity of Shannon entropy. Jensen measure is non-negative, if and only if $S^{(1)}(i) = S^{(j)}(i)$ for *i*=1,…,*B*. The definition for Jensen measure is similar to one defined for relative entropy.

2.3 Relative Entropy

"Relative entropy is a measure of distance between two distributions. In statistics it arises as an expected logarithm of the likelihood ratio" [11]. Equation (2.9) depicts such a definition,

$$D\left(p\|q\right)=\sum_{x}p\left(x\right)\log_{2}\left(\frac{p\left(x\right)}{q\left(x\right)}\right) \tag{2.9}$$

where *p(x)* is thought to be the true distribution and *q(x)* is the assumed distribution. This definition can also be interpreted as the penalty or cost of information content of assuming *q(x)* when the observed distribution is *p(x)*. Relative entropy satisfies the relationship $D\left(p\,\|\,q\right)\geq 0$ and is equal to zero if and only if *p(x)=q(x)* for all *x*. The definition of Shannon entropy can be used to simplify the expression in equation (2.9) and arrive at equation (2.10).

$$\begin{aligned} D\left(p\,\|\,q\right) &= -\sum_{x}p\left(x\right)\log_{2}q\left(x\right)-\left[\textit{Shannon entropy of }\ p\left(x\right)\right] \\ &= -\sum_{x}p\left(x\right)\log_{2}q\left(x\right)-\left[-\sum_{x}p(x)\log_{2}p(x)\right] \end{aligned} \tag{2.10}$$

We can apply the concept of relative entropy to Monte Carlo calculations for nuclear criticality and static reactor analysis. By setting the two random distributions in equation (2.9), $p(x)$ and $q(x)$, to the source distributions in the j^{th} cycle, $S^{(j)}$, and the 1st cycle, $S^{(1)}$, we arrive at equation(2.11).

$$D\left(S^{(j)}\,\|\,S^{(1)}\right)=\sum_{i=1}^{B}S^{(j)}\left(i\right)\log_{2}\left(\frac{S^{(j)}\left(i\right)}{S^{(1)}\left(i\right)}\right) \tag{2.11}$$

As the Monte Carlo calculation runs through the iterations, the relative entropy will show a general increasing trend due to the fact that $S^{(j)}$ is moving away from $S^{(1)}$, assuming that $S^{(1)}$ becomes less and less credible. The relative entropy is also lower bounded by the square of L^1 distance as well as greater than or equal to zero [11]. This relationship is found in equation (2.12).

$$D\left(S^{(j)} \,||\, S^{(1)}\right) \geq \frac{1}{2\log_e 2}\left(\sum_{i=1}^{B}\left|S^{(j)}(i) - S^{(1)}(i)\right|\right)^2 \geq 0 \tag{2.12}$$

The L^1 distance is the sum over all bins, of the absolute difference of the two distributions $S^{(j)}(i)$ and $S^{(1)}(i)$. Ueki found that a lower bound of the relative entropy exists, which increases through the cycle progression [4]. This property is found in equations (2.13) and (2.14).

$$D\left(\overline{S}^{j} \,||\, S^{(1)}\right) \geq D\left(f_j^* \,||\, S^{(1)}\right) \tag{2.13}$$

$$D\left(f_{j+1}^* \,||\, S^{(1)}\right) \geq D\left(f_j^* \,||\, S^{(1)}\right) \tag{2.14}$$

In equations (2.13) and (2.14), $\overline{S}^{j}$ is the expectation value of the source distribution in cycle *j*, i.e. $\overline{S}^{j}$ in *D* represents the ensemble average over infinitely many realizations of S^{j}; and $f_j^* \in C\left(\varepsilon^{(j)}\right)$, defined via $D\left(f_j^* \,||\, S^{(1)}\right) = \min_{f \in C\left(\varepsilon^{(j)}\right)}\left\{D\left(f \,||\, S^{(1)}\right)\right\}$,where $C\left(\varepsilon^{(j)}\right)$ is a set of probability distribution defined by $C\left(\varepsilon^{(j)}\right) = \left\{f \in P; D\left(f \,||\, \overline{S}^{(\infty)}\right) \leq \varepsilon^{(j)}, \varepsilon^{(j)} > 0\right\}$ in which $\varepsilon^{(j)} = D\left(\overline{S}^{(j)} \,||\, \overline{S}^{(\infty)}\right)$ and $\overline{S}^{(\infty)} = \lim_{j\to\infty} E\left[S^{(j)}\right]$. Ueki also showed that $\varepsilon^{(j)}$ monotonically decreases or $\varepsilon^{(1)} \geq \varepsilon^{(2)} \geq \ldots \geq \varepsilon^{(j)}$ [3].

2.3.1 Progressive Relative Entropy

By equations (2.13) and (2.14), $E\left[S^{(j)} \| S^{(1)} \right]$ is expected to be increasing. But by a practical reason the concept of the progressive relative entropy is developed. The progressive relative entropy (*PRE*) and is defined in equation (2.15).

$$\begin{aligned} PRE(j) &\equiv D\left(S^{(1)} \| 0.5\times\left(S^{(1)}+S^{(j)}\right)\right)+D\left(S^{(j)} \| 0.5\times\left(S^{(1)}+S^{(j)}\right)\right) \\ &= \sum_{i=1}^{B} S^{(1)}(i)\log_2\left(\frac{S^{(1)}(i)}{0.5\times\left(S^{(1)}(i)+S^{(j)}(i)\right)}\right)+\sum_{i=1}^{B} S^{(j)}(i)\log_2\left(\frac{S^{(j)}(i)}{0.5\times\left(S^{(1)}(i)+S^{(j)}(i)\right)}\right) \end{aligned} \tag{2.15}$$

The *PRE* is a symmetrization of the relative entropy, $D\left(S^{(j)} \| S^{(1)}\right)$. The development of the concept of progressive relative entropy is needed because of a problem that arises with the application of the relative entropy to the source distribution in Monte Carlo calculations. Referring back to equation (2.11), the problem with using relative entropy can be seen. If the source distribution in one of the bins in the first cycle is equal to zero, the relative entropy will become infinite. This condition can develop when the point source option in Monte Carlo is utilized. "To avoid such a singularity, the *PRE* of the source distribution at cycle *j* defined in (equation (2.15)) is utilized" [3]. *PRE* will also exhibit a general increasing trend because as the cycles progress, the binned source distribution in cycle *j*, $S^{(j)}(i)$, moves away from the binned source distribution in the first cycle, $S^{(1)}(i)$. "Thus, the distances of $S^{(j)}(i)$ and $S^{(1)}(i)$ to $0.5\times\left(S^{(j)}(i)+S^{(1)}(i)\right)$ will increase, which implies that both terms in the right side of equation (2.15) will exhibit a general increasing trend from the initial value of zero $\left(PRE(1)=2\times D\left(S^{(1)} \| 0.5\times\left(S^{(1)}+S^{(1)}\right)\right)=0\right)$ to their respective stationary levels" [3]. By

rearranging the equation for Jensen measure we can arrive at an equation that is a magnitude of the *PRE*. The next set of equations demonstrates this alteration.

$$
\begin{aligned}
J\left(S^{(j)}\right) &= -0.5\sum_{i=1}^{B}\left(S^{(1)}(i)+S^{(j)}(i)\right)\log_2\left(0.5\times\left(S^{(1)}(i)+S^{(j)}(i)\right)\right) \\
&\quad +0.5\sum_{i=1}^{B}S^{(1)}(i)\log_2 S^{(1)}(i)+0.5\sum_{i=1}^{B}S^{(j)}(i)\log_2 S^{(j)}(i) \\
&= -0.5\sum_{i=1}^{B}S^{(1)}(i)\log_2\left(0.5\times\left(S^{(1)}(i)+S^{(j)}(i)\right)\right)+0.5\sum_{i=1}^{B}S^{(1)}(i)\log_2 S^{(1)}(i) \\
&\quad -0.5\sum_{i=1}^{B}S^{(j)}(i)\log_2\left(0.5\times\left(S^{(1)}(i)+S^{(j)}(i)\right)\right)+0.5\sum_{i=1}^{B}S^{(j)}(i)\log_2 S^{(j)}(i) \\
&= 0.5\sum_{i=1}^{B}S^{(1)}(i)\log_2\left(\frac{S^{(1)}(i)}{0.5\times\left(S^{(1)}(i)+S^{(j)}(i)\right)}\right) \\
&\quad +0.5\sum_{i=1}^{B}S^{(j)}(i)\log_2\left(\frac{S^{(j)}(i)}{0.5\times\left(S^{(1)}(i)+S^{(j)}(i)\right)}\right) \\
&= 0.5\left[\begin{array}{l}\displaystyle\sum_{i=1}^{B}S^{(1)}(i)\log_2\left(\frac{S^{(1)}(i)}{0.5\times\left(S^{(1)}(i)+S^{(j)}(i)\right)}\right) \\ \displaystyle+\sum_{i=1}^{B}S^{(j)}(i)\log_2\left(\frac{S^{(j)}(i)}{0.5\times\left(S^{(1)}(i)+S^{(j)}(i)\right)}\right)\end{array}\right] \\
&= 0.5\left[D\left(S^{(1)}\,\|\,0.5\times\left(S^{(1)}+S^{(j)}\right)\right)+D\left(S^{(j)}\,\|\,0.5\times\left(S^{(1)}+S^{(j)}\right)\right)\right] \\
&= 0.5PRE(j)
\end{aligned}
\tag{2.16}
$$

From this rearrangement of the Jensen measure, it is shown that the Jensen measure is one half of the progressive relative entropy at cycle *j*.

2.3.2 Posterior Relative Entropy

The final relative entropy that was proposed was a "posterior defensive visual diagnostic" [12]. This diagnostic is known as the posterior relative entropy (*PosRE*). Posterior relative entropy is defined by equations (2.17), (2.18), (2.19) and (2.20).

$$PosRE(j) \equiv D\left(S^{(j)} \,\|\, \mu(i)\right) \tag{2.17}$$

$$\begin{aligned} \mu(i) = &\text{ average source distribution over second} \\ &\text{half of prior assumed active cycles} \\ = &\frac{\sum_{j=N_{1/2}+1}^{N_{\max}} S^{(j)}(i)}{\left(N_{1/2} - N_{\max}\right)} \end{aligned} \tag{2.18}$$

$$\begin{aligned} N_{1/2} = &\text{ Final cycle of first half of active cycles} \\ N_{\max} = &\text{ Total number of cycles} \end{aligned} \tag{2.19}$$

$$PosRE(j) = \sum_{i=1}^{B} S^{(j)}(i) \log_2\left(\frac{S^{(j)}(i)}{\mu(i)}\right) \tag{2.20}$$

One can check after the final cycle or iteration to verify the *PosRE* crosses the average of the *PosRE* over the second half of the active cycles before the first active cycle begins. If the value of the *PosRE* crosses the average of the *PosRE* over the second half of the active cycles after the beginning of the active cycles the *PosRE* is said to be invalid. The theoretical foundation for the *PosRE* was developed by Ueki from equation (2.21) [3].

$$D\left(E\left[S^{(n)}\right] \,\|\, \lim_{m\to\infty} E\left[S^{(m)}\right]\right) \geq D\left(E\left[S^{(n+1)}\right] \,\|\, \lim_{m\to\infty} E\left[S^{(m)}\right]\right) \tag{2.21}$$

Where $E\left[S^{(n)}\right]$ is the ensemble average over infinitely many realizations of $S^{(n)}$ and $\lim_{m\to\infty} E\left[S^{(m)}\right]$ is the stationary probability distribution.

It was Brown and Ueki who proposed and developed the use of Shannon entropy and relative entropy as indicators of convergence [2]. With the preceding definitions of

entropies it can be determined at which cycle the source distribution converges or fluctuates around the true source distribution. By taking the values calculated at each cycle for each of the entropies and comparing them to the values calculated in the previous cycles it can be determined if the entropies have reached stationarity. This is known as an on-the-fly diagnostic and can be performed by a Wilcoxon signed rank test. A step refined on-the-fly judgment was proposed by Ueki [3] and will be discussed further in chapter 5.

The determination of posterior relative entropy is different from the other entropy calculations because it depends on user defined values, i.e. the number of inactive cycles and the number of active cycles. *PosRE* can be manipulated by changing the number of active and inactive cycles. However inconsistencies can arise in the use of *PosRE* if the total number of cycles in the Monte Carlo run is less than it takes for the model to reach a state of stationarity. This inconsistency can be seen when we look at the analysis of a homogeneous cube with vacuum boundary conditions. The simulation was run with 10,000 particles per cycle. Because of the number of particles in this simulation the source distribution takes about 1400-1500 cycles to reach a stationary state. This can be seen from *RAPI* behavior in Figure 2.2, which is developed and explained in Chapter 3.

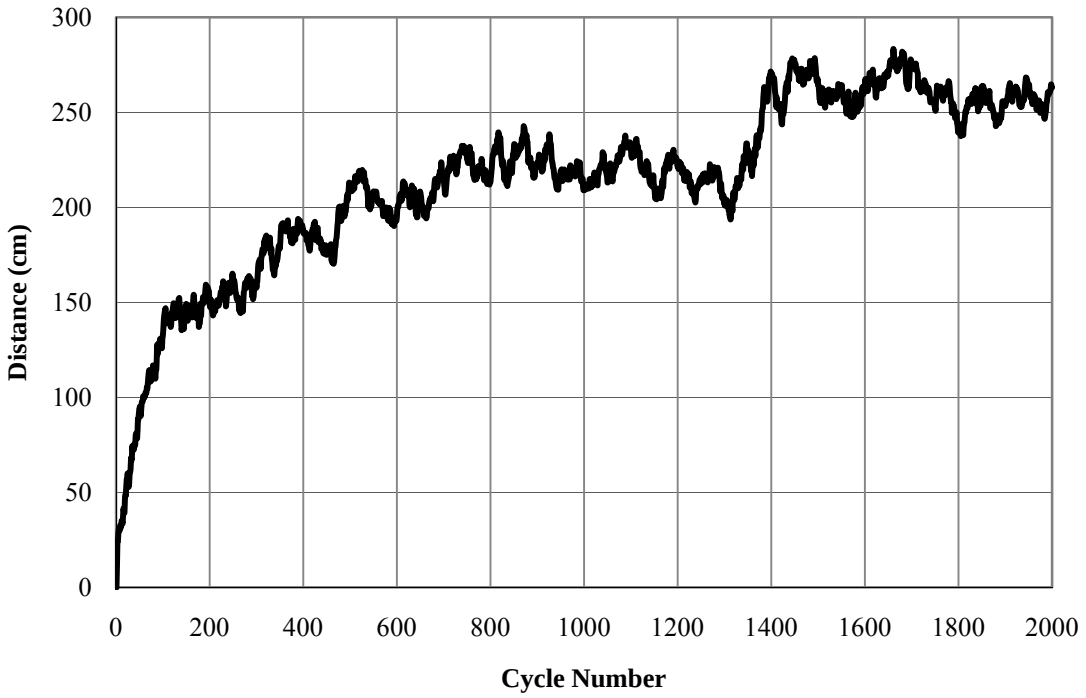


Figure 2.2: *RAPI* of Homogeneous Cube with 10,000 particles per cycle

This model is shown to converge around 1500 cycles. If a run of this simulation is done with the total number of iterations set to less than 1500 cycles, the calculation of the posterior relative entropy returns values inconsistent with what theory says posterior relative entropy should behave like. *PosRE* should always decrease from its initial value as is illustrated by equation (2.21) [3]. A model of the homogenous cube problem was run for 1400 cycles with 400 inactive cycles, and the results for the calculation of the posterior relative entropy is displayed in Figure 2.3. The value of the *PosRE* over the cycles or iterations of the run increases for approximately 80 cycles then begins to decrease.

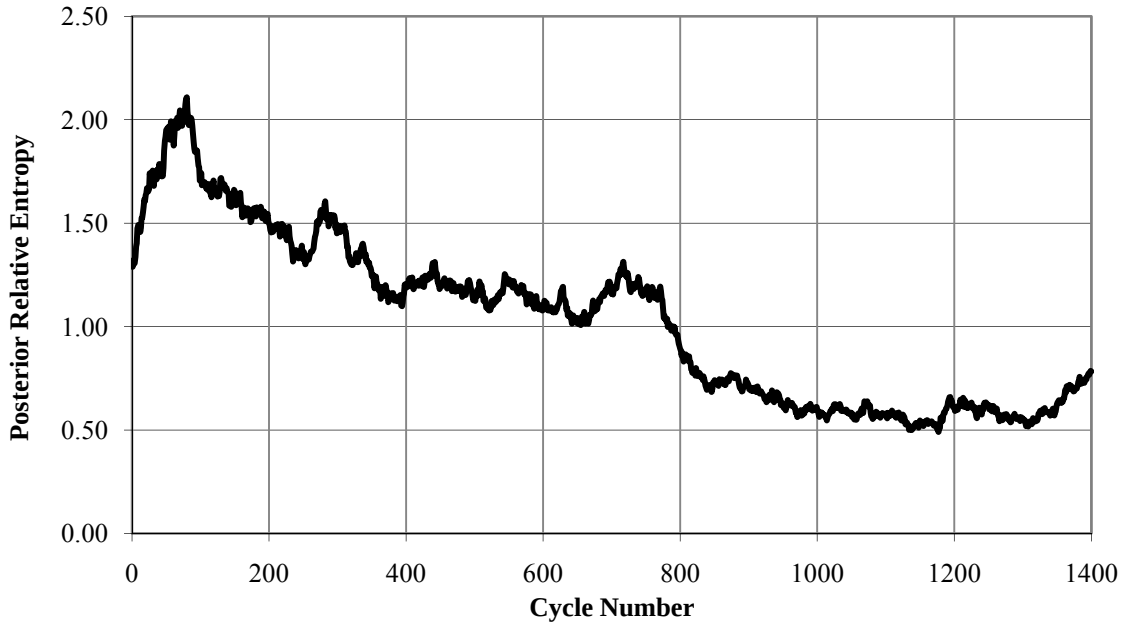


Figure 2.3: Posterior Relative Entropy of Homogeneous Cube with 10,000 Particles per cycle

This inconsistency indicates that the predefined number of inactive cycles was too small because the utility of the *PosRE* entails the implicit assumption that convergence occurs before the beginning of the second half of the active cycles. To be able to use *PosRE* in the proper manner, the number of inactive cycles needs be set to the number of cycles that it takes for the source distribution to reach a stationary state. In the case of the homogeneous cube the number of inactive cycles needs to be set to 1500 and the number of active cycles be 1000 for a total number of cycles of 2500. It is for this reason that posterior diagnostic measures are not an "ultimate" judgment; the stationary state needs to be determined during the iterated process. This can be done with an on-the-fly judgment like the Wilcoxon rank sum. Problems can also arise with the calculation of the entropies, when the model being simulated is undersampled or the number of particles for each cycle is too low for the run to reach a convergent state.

Chapter 3

Regionwise Average Position Indicator of Convergence

In Monte Carlo (MC) techniques it is very useful to know at which cycle the source distribution has become stationary or converges. When the particle distribution converges, it reaches a state of equilibrium and is said to match the true source distribution of the problem. It is at this cycle that the physical properties of the system can begin to be tallied, and the results of the tallies will more accurately represent the physical parameter and give better insight to the behavior of the system being simulated. To accomplish the task of determining the cycle at which the source distribution has reached the true fundamental mode, a convergence indicator needs to be developed. The convergence indicator needs to be representative of the source distribution's state throughout the iterational process. The indicator also needs to be free of requirement of user input mesh. This indicator will be presented as the regionwise average position indicator (*RAPI*) and was developed from the concept of automatic meshing.

In MC neutron transport calculations a predefined number or batch of particles is generated within the boundaries of a geometry with different distribution characterizations. Tracking the interactions of one batch of particles through the specified geometry and materials is called a cycle. This tracking process is repeated over many cycles or iterations. Each cycle has the same number of particles and the starting position in the first cycle of the first batch of particles is determined from the specified distribution. In each subsequent cycle the starting position of the batch of particles is determined from the fission sites of the previous batch of particles. Statistics of the physical properties of a problem can be accumulated by recording the interactions of the particles at each cycle or iteration in a simulation. To record these interactions accurately a mesh must be laid upon the geometry. The mesh divides the geometry into bins. Currently when using the MC process of neutron transport for the calculation of the effective multiplication of a nuclear system or other physical properties of a system, a user of the computer program must input coordinates. These coordinates are used to set up a mesh on the geometry, which in turn creates bins that discretize the geometry. A user defined binning scheme can be very limited because the binning scheme needs to be problem specific. The binning scheme is problem specific because it is the user who determines where the discretization will be most useful. The mesh, and thus the bins, need to be placed in the geometry where the most accurate and valuable information about the system can be obtained. This can be seen if we examine the different ways as to which nuclear systems can be described. Nuclear systems can range from lumps of fissionable material to nuclear reactor cores, with regions within the geometry ranging from fissile to non-fissile material. Not only can nuclear systems have different

geometries and material-neutron interaction probabilities, but they can also have different source distributions, including but not limited to point sources, uniform sources, and line sources. The requirement of preset meshing for various source distributions requires that the user intuitively knows how the particle distributions will behave in all possible types of geometry, material and initial source distribution possible.

The need for knowledge of particle distribution evolution from cycle to cycle in any geometry and initial source distribution combinations and its impact on mesh coordinate assignment can be eliminated by automating the mesh assignment and bin creation. The automation of bin creation can be done by taking the average of the particle positions in any geometry and initial source distribution and using this averaging to define the mesh coordinates. If the average particle position is used to determine how the mesh is defined, bins will be created in the regions of the geometry where the particle density is the highest. It is in these regions of the geometry that the binning definition will have the greatest impact on the determination of the physical properties of the system as well as obtaining information that leads to understanding about how the system may behave.

Every particle generated in a geometry is generated only within fissile regions of the geometry, and each particle position has an x, y and z component. In one-dimensional problems, one dimension is defined to be finite while the other two are defined to be "infinite". For one-dimensional geometry, only one the finite component will be used in performing calculations, whether it is the x, y or z component. In two-dimensional problems, two dimensions are finite while one is infinite, and in three-dimensional problems all three dimensions are finite. It is only in the finite dimensions that we wish

to implement the automatic meshing. To begin the automatic meshing, the average of the particle positions can be found, and this calculation is described in the equation (3.1).

$$x_{avg-1} = \frac{1}{N}\sum_{k=1}^{N} x_k \quad y_{avg-1} = \frac{1}{N}\sum_{k=1}^{N} y_k \quad z_{avg-1} = \frac{1}{N}\sum_{k=1}^{N} z_k \tag{3.1}$$

In these equations N is the total number of particles generated in the geometry and k is an index. For example if the problem is a slab that is finite in the x dimension and infinite in the y and z dimensions, the average in the x dimension or x_{avg} will only be calculated. In a two-dimensional geometry, only the average in two dimensions will be taken, and in three all three will be calculated. With these averages, a coordinate that will be the first center point has been obtained. This point becomes the intersection of the bin boundaries in the meshing. In a one-dimensional geometry two bins will be created by the division of the geometry by the first center point; four bins will be created in a two-dimensional geometry; and eight bins will be created in a three-dimensional geometry. We can further subdivide these bins, in any geometry, by taking the average position of the particles within each of the bins. This is done with equation (3.2),

$$x_{avg-i} = \frac{1}{N_{i-1}}\sum_{k=1}^{N_{i-1}} x_{k-(i-1)} \quad y_{avg-i} = \frac{1}{N_{i-1}}\sum_{k=1}^{N_{i-1}} y_{k-(i-1)} \quad z_{avg-i} = \frac{1}{N_{i-1}}\sum_{k=1}^{N_{i-1}} z_{k-(i-1)}$$

$$\begin{aligned} N_1 + N_2 &= N; && \text{one-dimension} \\ N_1 + N_2 + N_3 + N_4 &= N; && \text{two-dimensions} \\ N_1 + N_2 + N_3 + N_4 + N_5 + N_6 + N_7 + N_8 &= N; && \text{three-dimensions} \end{aligned} \tag{3.2}$$

which is similar to (3.1) with N, the number of particles in the entire geometry, changed where i is the regionwise average position coordinate, i-1 is the bin that is under consideration and N_{i-1} is the number of particles within the bin denoted i-1. With this

equation, there are an increased number of bins created in the geometry by automatic meshing. In this calculation the sum of the particles in the bins created by the center point is equal to the total number of particles. An example and explanation of automatic meshing for geometries in one, two and three dimensions will be discussed next.

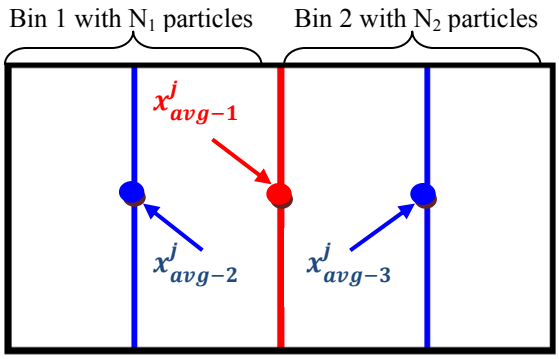


Figure 3.1: 1D Automatic Meshing

The first example is that of a one-dimensional geometry. For this example the geometry is infinite in the *y* and *z* directions. The *x* direction is finite, meaning that the exterior boundaries are known. The red line in the middle of Figure 3.1 represents the first bin boundary intersection that is created from the calculation $x_{avg-1} = \frac{1}{N}\sum_{k=1}^{N} x_k$. In general the coordinate for the x-direction bin center will be identified as x^j_{avg-i} where *i* identifies the center point, *i*-1 identifies which bin under consideration, and *j* identifies which cycle or iteration the calculation is being performed for. In the case of the first bin boundary intersection, *i* is equal to one. When the dimension in question in a one-dimensional investigation is either the *y* or *z* direction there will be similar values calculated for *y* or *z* and will be represented by y^j_{avg-i} and z^j_{avg-i}. The initial calculation of the average particle position, represented by the red line in Figure 3.1, creates two bins. From the two bins, two center coordinates will be determined by another averaging.

These coordinates in the *x* direction are represented by the blue lines. They are calculated by taking the average of the positions of the particles located in each of the spaces or bins, separated by the first bin boundary intersection. These two calculations, done with equation (3.2), create the secondary centers x^j_{avg-2} and x^j_{avg-3}. With these two secondary centers there is a total of three average coordinates and a total of 4 bins. By making x^j_{avg-2} and x^j_{avg-3} new bin boundary intersections, one can further compute the regionwise average particle position in each of the 4 bins leading to a total of 8 bins and 7 average particle positions. This meshing can be continued making the meshing finer or the bin size smaller, concentrating a higher number of bins in the areas of higher particle density.

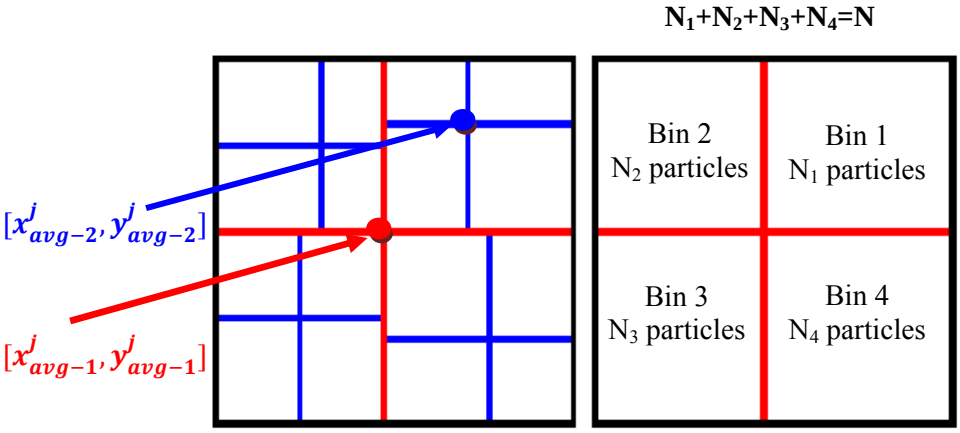


Figure 3.2: 2D Automatic Meshing

The next example is shown in Figure 3.2 and is that of a two-dimensional square. For explanation purposes, the *x* and *y* directions will be finite and the *z* direction will be infinite, although there could be any combination of two finite dimensions with the remaining being infinite. The red center point in the left square represents the average of the particle positions within the entire geometry with respect to the *x* and *y* directions and becomes the intersection of the 4 bin boundaries indicated by the red lines. This will be the first bin boundaries intersection and average coordinate. This coordinate divides the

geometry into four bins. We then find the average particle position in each of the bins or regions to find four more centers or bin boundary intersections. At two levels of averaging in a two-dimensional geometry, there are a total of five average coordinates and 16 bins. As with the one-dimensional scheme, the automatic meshing can be made finer and finer; increasing the amount of bins for information analysis, and these bins will be in the areas of highest particle density.

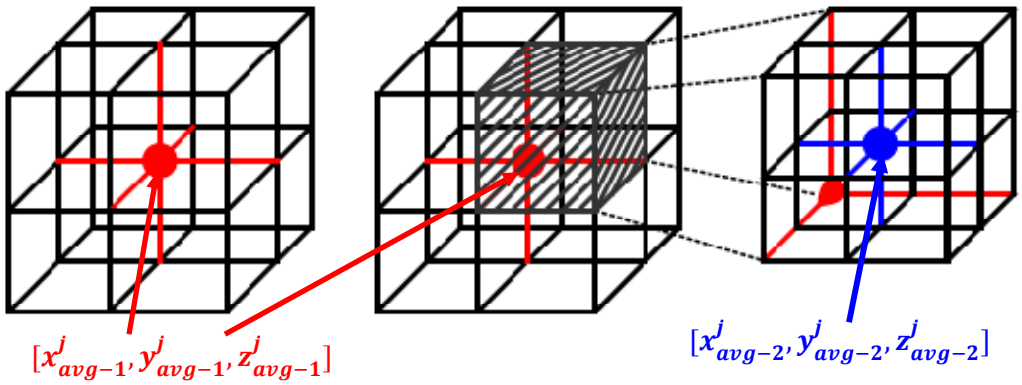


Figure 3.3: 3D automatic meshing

Finally there is the case of a three-dimensional geometry or cube, and this case is depicted in Figure 3.3. Each particle has an *x*, *y*, and *z* component of its position within the geometry. By taking the average of the *x*, *y*, and *z* components respectively, an average particle position or center coordinate that is related to the particle density can be found. This center coordinate becomes the intersection of 8 bin boundaries, whose intersections connected to the center is shown in red in Figure 3.3. This center coordinate will divide the geometry into eight bins; top-upper-right, top-upper-left, top-lower-right, top-lower-left, bottom-upper-right, bottom-upper-left, bottom-lower-right, and bottom-lower-left. As can be seen in the figure, one of the eight bins, top-upper-right, created from the first center coordinate is selected and shaded. Within the top-upper-right bin the

average particle coordinate is found again as with the first center coordinate creating a secondary center. This mesh coordinate or center is represented by the intersection of the blue lines. This process is repeated for the seven other bins. After the second set of averaging is completed a total of nine centers or mesh coordinates are found and these centers create a total of 64 bins. Once again the automatic meshing and bin size can be made finer and finer depending on the needed amount of detail by increasing the number of divisions.

The number of bins and the number of centers created by the automatic meshing for any dimensional analysis is dependent upon the following set of equations.

$$B = \left(2^D\right)^L \qquad T = \sum_{i=0}^{L-1} \left(2^D\right)^i \tag{3.3}$$

In these equations, B is the number of bins, T is the number of centers, D is the number of dimensions the geometry occupies, and L is the level of divisions or number of averages performed.

The automated meshing and bin creation technique can serve as an indicator of convergence. For example, in a three-dimensional space, this can be accomplished by taking the sum of the distances between the coordinates of the nine centers at each cycle from the corresponding nine centers or coordinates of the first cycle. The sum of the distances between corresponding centers can be found using equations (3.4), (3.5) and (3.6).

$$D_x^j = \sum_{i=1}^{T} d_i^j = \sum_{i=1}^{T} \sqrt{\left(x_{avg-i}^j - x_{avg-i}^1\right)^2} \qquad (T = 3) \tag{3.4}$$

$$D_{xy}^{j} = \sum_{i=1}^{T} d_i^j = \sum_{i=1}^{T} \sqrt{\left(x_{avg-i}^{j} - x_{avg-i}^{1}\right)^2 + \left(y_{avg-i}^{j} - y_{avg-i}^{1}\right)^2} \qquad (T = 5) \tag{3.5}$$

$$D_{xyz}^{j} = \sum_{i=1}^{T} d_i^j = \sum_{i=1}^{T} \sqrt{\left(x_{avg-i}^{j} - x_{avg-i}^{1}\right)^2 + \left(y_{avg-i}^{j} - y_{avg-i}^{1}\right)^2 + \left(z_{avg-i}^{j} - z_{avg-i}^{1}\right)^2} \qquad (T = 9) \tag{3.6}$$

The three equations are used for one dimension, two dimensions, and three dimensions respectively. The *j* in the superscripts is indicative of cycle number, the *i* in the subscripts represents which of the nine centers is under question, and *T* represents the total number of centers to be summed over. Equation (3.4) can be used for any dimension and equation (3.5) can be used for any combination of two dimensions. For a three-dimensional geometry, by plotting the distance of the nine centers from the initial nine centers over all cycles, the determination of the cycle of convergence can be made. This indicator of convergence will be designated the regionwise average position indicator or *RAPI*. The *RAPI* can be used as convergence indicator of the source distribution because the centers describe the overall and regionwise average source distribution location within geometry, as well as tracks that change in the sum of the distances of each center at each cycle to that of the corresponding centers in the first cycle. When the sum of the distances converges, fluctuating around a central value, the source distribution can be said to be converged since *RAPI* values are determined from particle positions. An example of how *RAPI* behaves in a Monte Carlo calculation's cycle progression is found in Figure 3.4.

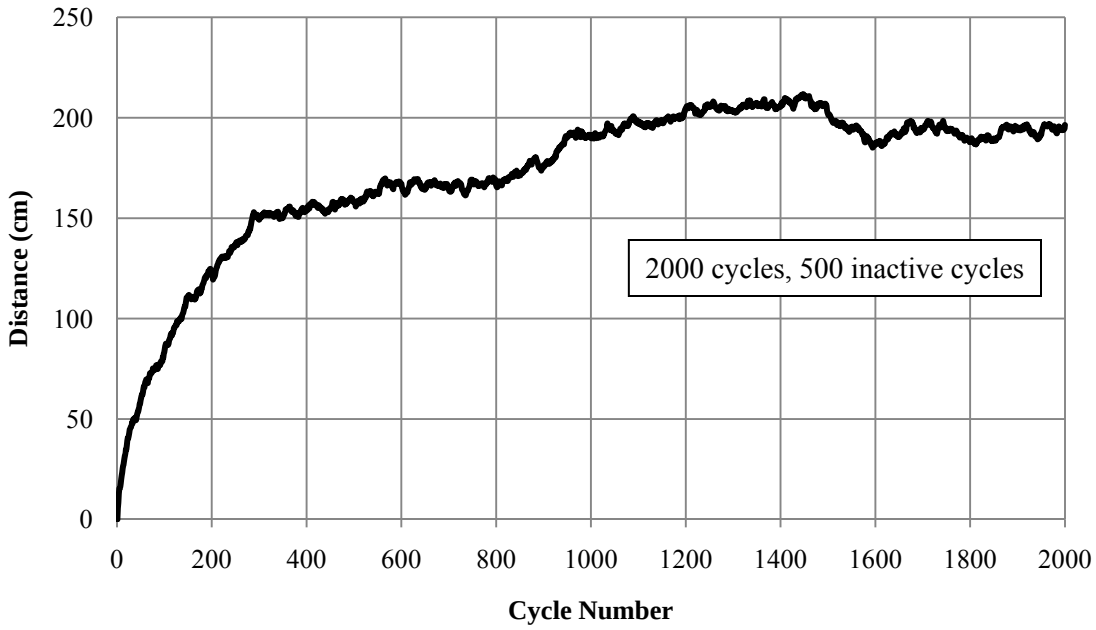


Figure 3.4: *RAPI* for Homogeneous Cube Case with 100,000 Particles per Cycle

The homogeneous cube problem had side length of 200 cm. To create this figure a simulation was run with 100,000 particles, for 2000 active cycles and 500 inactive cycles. Because the homogeneous cube is a symmetric problem, the distance of the first center point or first average coordinate at each cycle or iteration from that of the first average coordinate at the initial cycle or iteration should exhibit a quasi stationarity from the beginning. This indicator is calculated by eliminating the sum in equation (3.6) and arriving at equation (3.7).

$$D^j = \sqrt{\left(x^j_{avg-1} - x^1_{avg-1}\right)^2 + \left(y^j_{avg-1} - y^1_{avg-1}\right)^2 + \left(z^j_{avg-1} - z^1_{avg-1}\right)^2} \tag{3.7}$$

If this fact turns out to be true then the amount of computation that would need to be done for symmetric problems could be reduced. Figure 3.5 is created by plotting the distance of the single center position at each cycle from the single center position at the first cycle against the number of cycles.

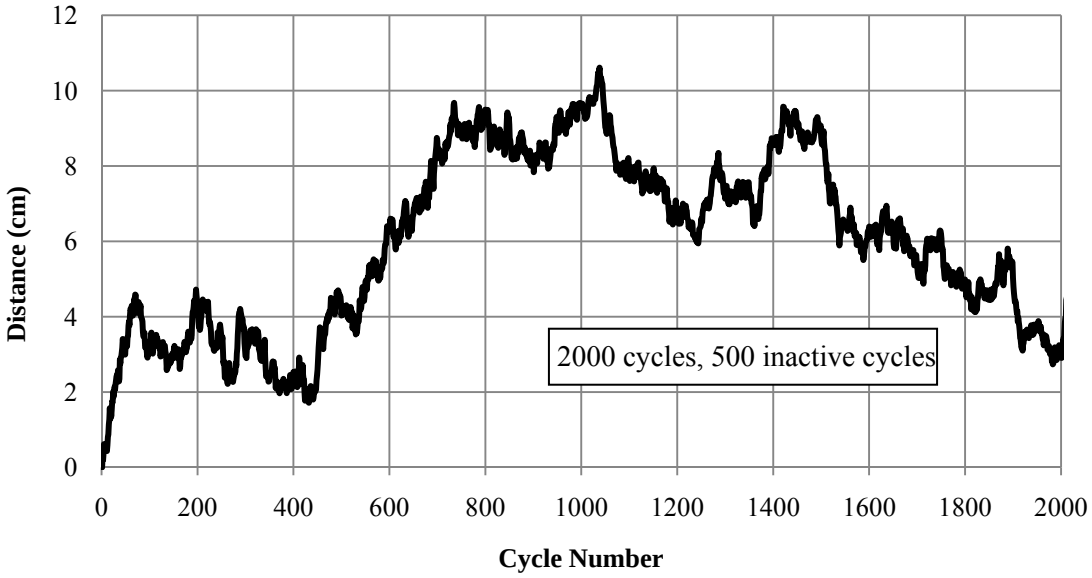


Figure 3.5: Single Center Distance for Homogeneous Cube Case with 100,000 Particles per Cycle

What is found is that the single center distance remains near that of first value, within about 10 cm, throughout all the iterations or cycles. The single center distance fluctuates around the first value, being in a quasi stationary state from the very beginning. The single center distance may be used to indicate convergence, but by comparing the graphs in Figure 3.4 and Figure 3.5, it can be seen that the sum of the distances is a better indicator of convergence of the source distribution in a symmetric system. In Figure 3.4 the *RAPI* does not reach stationarity immediately whereas the single center distance in Figure 3.5 is in a stationary state from the beginning. The sum of the distances indicator or *RAPI* takes into account the source distributions variation within the different regions of the entire geometry by combining the center's distances into an overall indicator of the source distribution.

We can also see that the use of the *RAPI* for convergence determination may be promising by comparing it to the Shannon entropy, a mesh based convergence indicator

of the source distribution proposed by Ueki [3]. The Shannon entropy for the homogeneous case is in Figure 3.6.

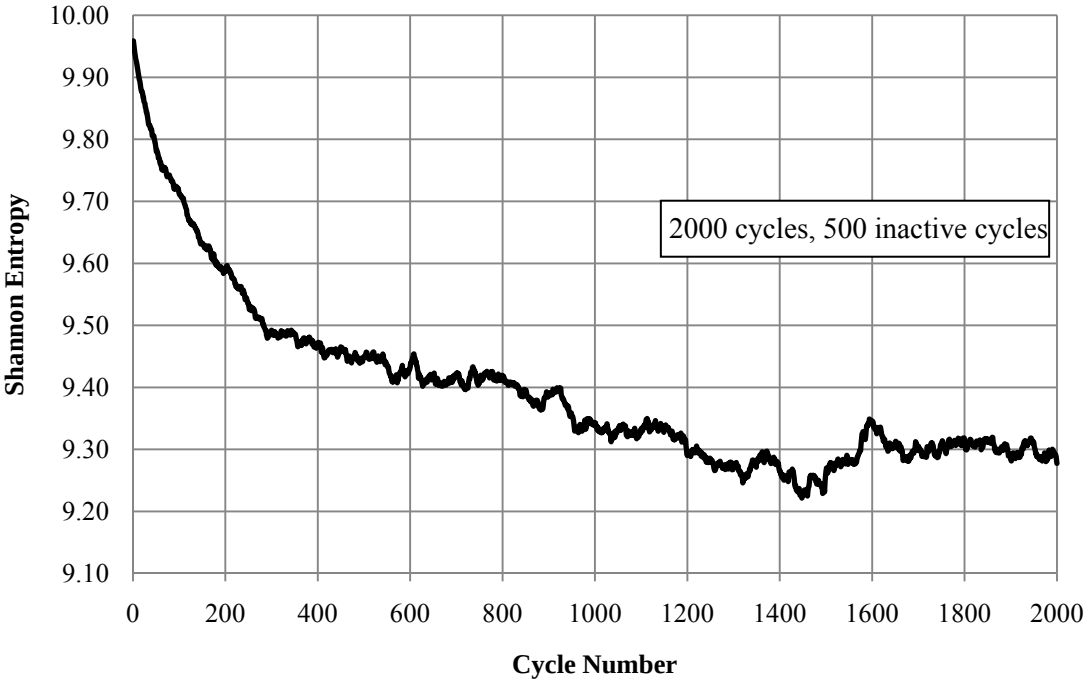


Figure 3.6: Shannon Entropy for Homogeneous Cube Case with 100,000 Particles per Cycle

The trend of the *RAPI* is opposite that of the Shannon entropy, increasing instead of decreasing but the fluctuations in both of the indicators occur at similar points in the number of cycles. So from this, *RAPI* looks to be a promising mesh-input-free convergence indicator. The *RAPI* method will be used to determine source distribution convergence for different scenarios using two computational packages, in a MC research code and in MCNP [13]. This method will then be compared against the four entropies associated with each simulation. By comparing the *RAPI* method plots against these entropies, which are verified indicators of convergence, it can be seen if the *RAPI* method can be used as a convergence indicator. The numerical results of the investigation will be discussed in Chapter 6.

The usefulness or accuracy of the *RAPI* method may be affected by the dominance ratio of the nuclear system. The *RAPI* method may also be affected by the heterogeneity of the system. The effects that these system characteristics have on the *RAPI* method will be discussed in Chapter 6, being inferred from the numerical results. Another question that needs to be looked at is if there is a difference between the research code and MCNP calculations, and if these differences cause a difference in the convergence determination. This method needs to be rigorously tested and compared to proven methods to validate its usefulness.

With the use of automatic meshing and bin creation, the need for knowledge of particle distribution behavior gained from years of experience as well as the need to input the mesh coordinates can be eliminated. This method can be useful in automatic determination of the cycle at which the convergence of the particle distribution occurs and thus initiating physical property information retrieval. This will ultimately increase the accuracy of results and the usefulness of the modeling being performed.

Chapter 4

Euclidian Minimum Spanning Tree Applications

In Monte Carlo simulations, it is useful to know at which cycle the source distribution has converged, as well as if the particle population is of sufficient size for a given tally cell or bin size, to accurately model a nuclear criticality problem. If the source distribution has converged and the problem/simulation is adequately populated, the error and bias associated with tallying of physical properties of the problem will be reduced. To determine when the source distribution has converged or if the problem is adequately populated, a diagnostic is needed. Currently, such diagnostics are mesh-based. Problems can arise with mesh-based diagnostics because the meshing/binning is defined by the user and the user can make mistakes in the location and resolution of the binning scheme. The concept of Euclidian minimum spanning trees (EMST) was investigated as mesh-input-free means of drawing diagnostic information from the source particles in Monte Carlo source iterations. Because Euclidian minimum spanning trees have been used in a number of applications with reasonable success [4-8], their use in Monte Carlo diagnostic work seemed practical. This section will explore applications of graph theory techniques

to diagnostics of Monte Carlo methods, to reduce error and bias in the tallying of physical properties and eliminate the need for user input mesh in diagnostic work.

4.1 Graph Theory

Graph theory is the study of mathematical structures called graphs that model relationships between objects in a specified collection. A graph consists of two finite sets, a set of vertices and a set of edges. An example of such a graph can be found in Figure 4.1.

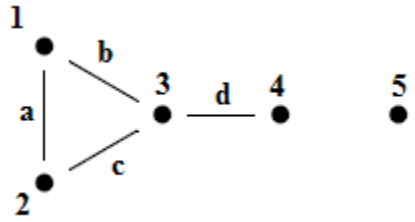


Figure 4.1: A graph with vertices as numbers and edges as letters

A graph is connected when every two of its vertices are connected [14]. This means that from any vertex in the graph, any other vertex in the graph can be reached, or there exists edges between vertices so that every other vertex is connected through these edges to every other vertex. An example of a connected graph is shown in Figure 4.2.

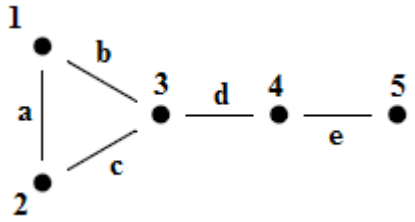


Figure 4.2: Connected Graph

A weighted graph is a graph whose edges have weight. A spanning tree of a graph is a spanning subgraph of that graph that is a tree [14]. Here, a spanning subgraph of a graph

is a graph that has exactly the same set of vertices as the parent graph but has the same or fewer edges. A tree is a graph that is acyclic and has *n* vertices and *n*-1 edges. Acyclic means that there are no cycles in the graph (cycles is used here as a reference to a circular path, not a cycle or iteration as in Monte Carlo source iterations). An example of a cycle can be found in Figure 4.3. A spanning subgraph of the graph in Figure 4.2 can be found in Figure 4.4.

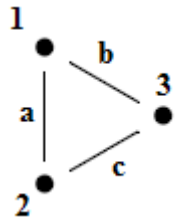


Figure 4.3: Three Cycle Graph

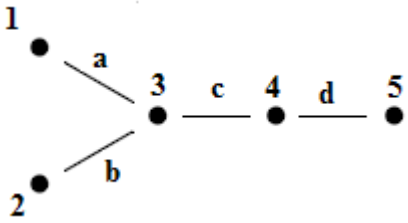


Figure 4.4: Spanning Tree

A minimum spanning tree is the least total weight spanning tree of a weighted graph. There may be many spanning trees of a weighted graph, but the minimum spanning tree has the least total weight possible for the graph in question. There exist different algorithms to find the minimum spanning tree of a graph, but Prim's algorithm will be used for the purpose of the investigation described in this thesis.

4.2 Prim's Algorithm

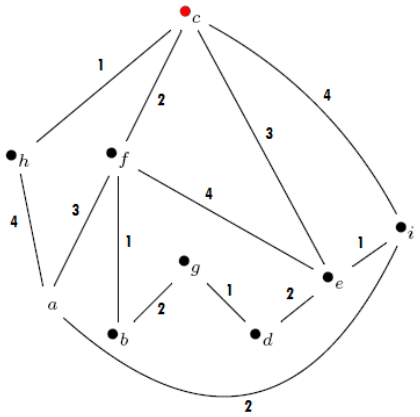
Prim's Algorithm works by taking a weighted connected graph and finds a minimum spanning tree. Prim's algorithm starts by choosing a vertex, any vertex in the graph say v_1. The next step in the algorithm is to choose an edge, $e_1=v_1v_2$, which is incident to vertex v_1 with the least weight and which is not a loop. Edge e_1 is connected to another vertex, v_2. Next the edge with the least weight that is incident to v_1 or v_2 is chosen to be part of the minimum spanning tree but with the other end of the edge not incident on v_1 or v_2, "i.e., we choose $e_2=v_iv_3$ where $i \in \{1,2\}$ but $v_3 \neq v_2, v_1$" [14]. This process of choosing edges of smallest weight, one whose end is a vertex previously chosen and the other end becoming involved for the first time, until there has been n-1 edges included in the minimum spanning tree (assuming the graph has n vertices). At completion of this process all of the vertices have been involved in the construction of the subgraph and it is proved to be a minimum spanning tree [14].

The following are two examples of Prim's algorithm: the first starts with one vertex in a graph, and the other by starts with a different vertex in the same graph. This will be done to demonstrate that Prim's algorithm will return the same minimum spanning tree for the same graph by starting with any vertex.

Example 1

<table>
<tr>
<td>

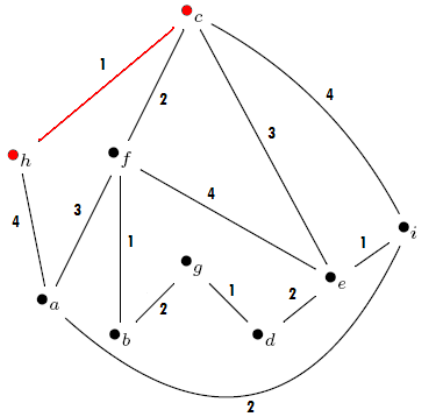
Step One



We begin Prim's by randomly choosing vertex *c*.

</td>
<td>

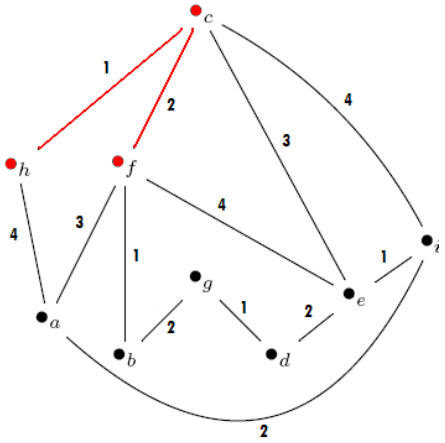
Step Two



We first choose the edge with weight one connected to *c* because it has the least weight.

</td>
</tr>
<tr>
<td>

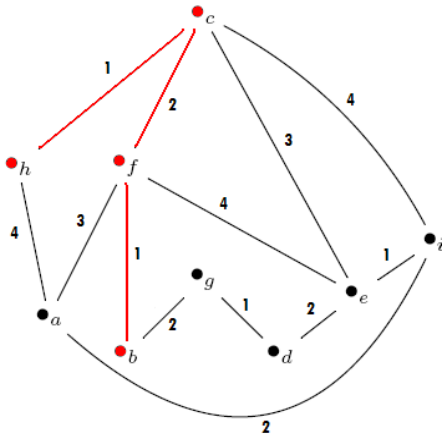
Step Three



We next choose the edge connected to *c* with weight two because it has the least weight of edges connected to vertices *c* and *h*.

</td>
<td>

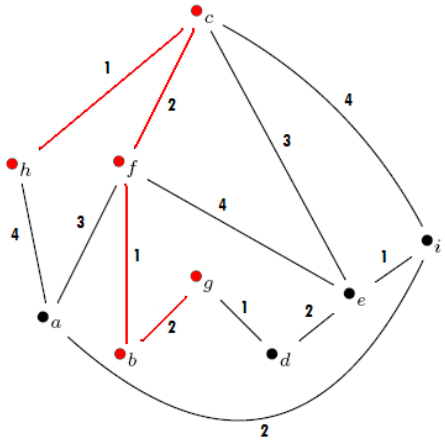
Step Four



We next choose the edge of weight one connected to vertex *f* because it has the least weight of all edges incident on previously used vertices.

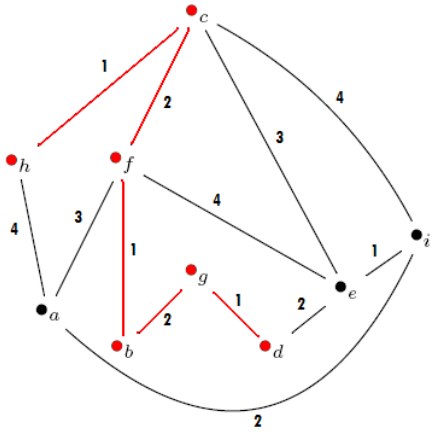
</td>
</tr>
</table>

Step Five



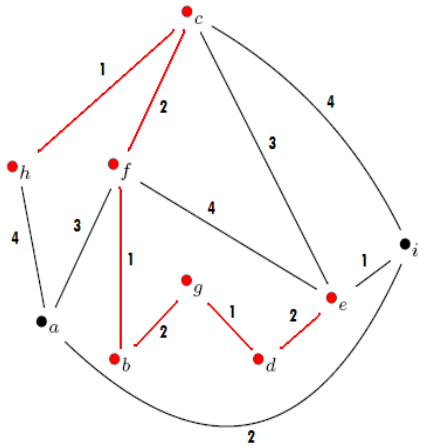
The next edge chosen is *bg* with a weight of two.

Step Six



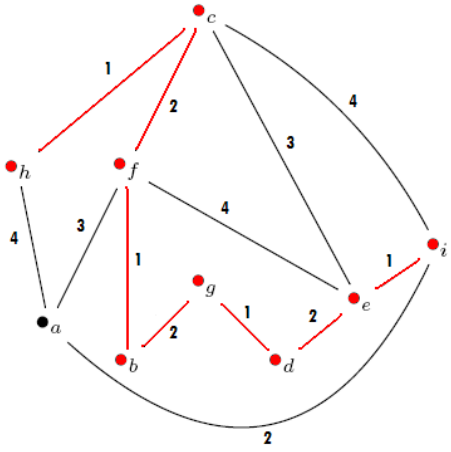
The next edge included in the minimum spanning tree is *gd* with weight one.

Step Seven

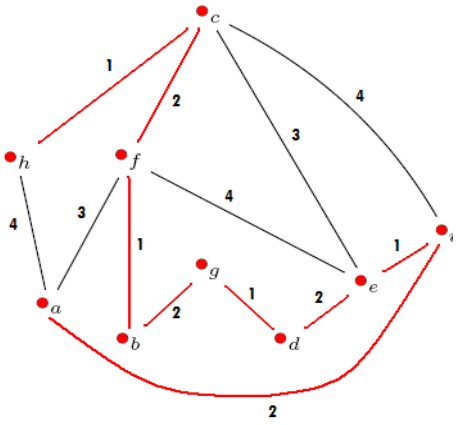
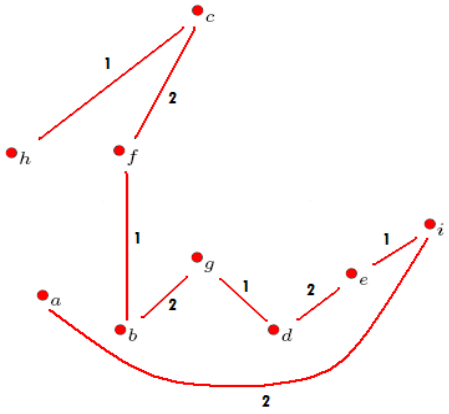


The next edge is *de* of weight two.

Step Eight



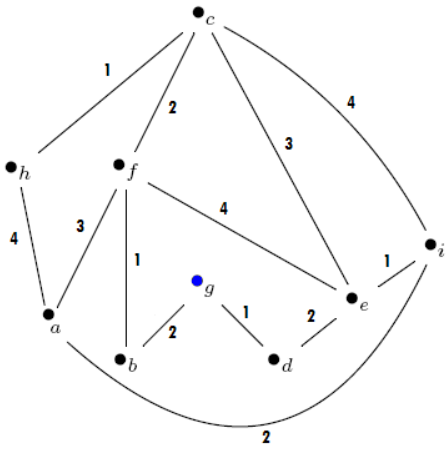
The next edge is *ei* of weight one.

Step Nine	Final MST
	
The final step to include all edges in the minimum spanning tree is to include edge *ia* of weight two.	This is the minimum spanning tree starting at point *c* with a total weight of twelve.

After demonstrating the first example of finding a minimum spanning tree, Prim's algorithm will be implemented a second time but by choosing a different starting vertex. This is done to illustrate that the total weight of the minimum spanning tree is not changed by the choice of starting vertex.

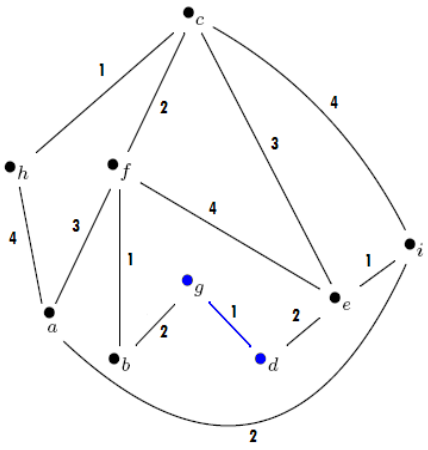
Example 2

Step One



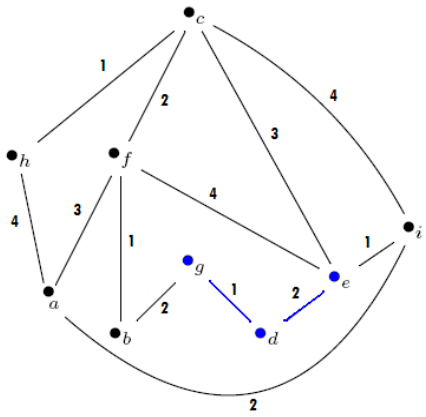
We begin Prim's by randomly choosing *g* as the starting vertex to create the minimum spanning tree.

Step Two



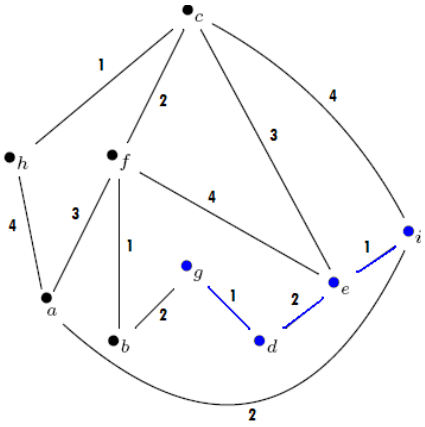
We next choose the edge of weight one connected to vertex *g* because it has the least weight of all edges.

Step Three

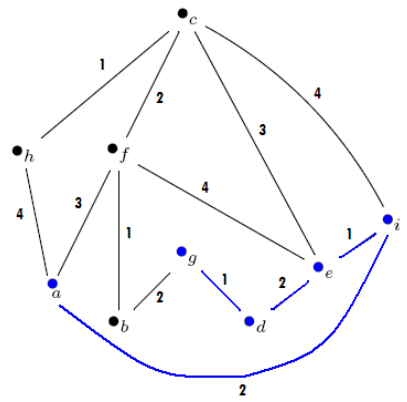
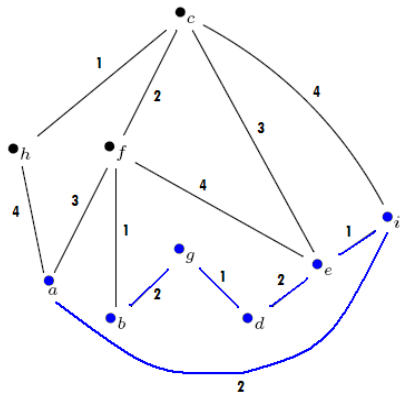
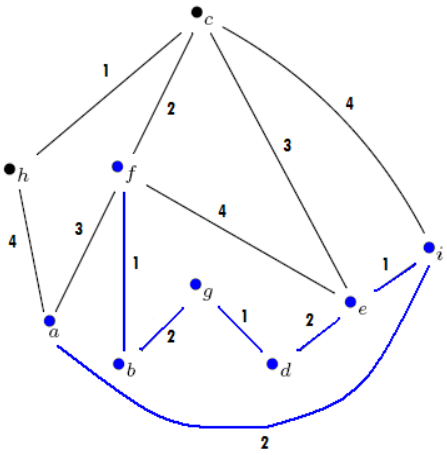
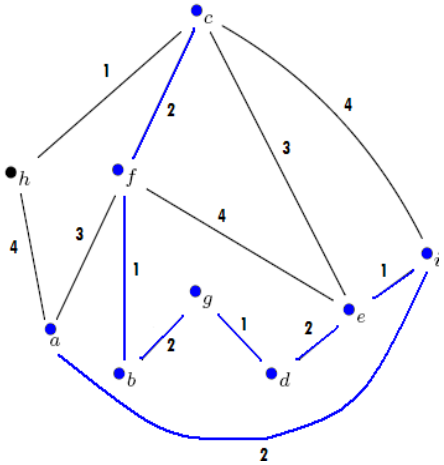


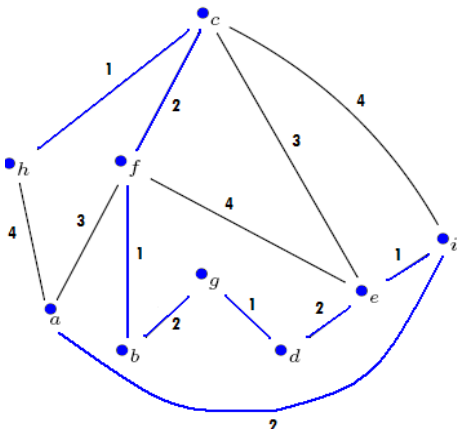
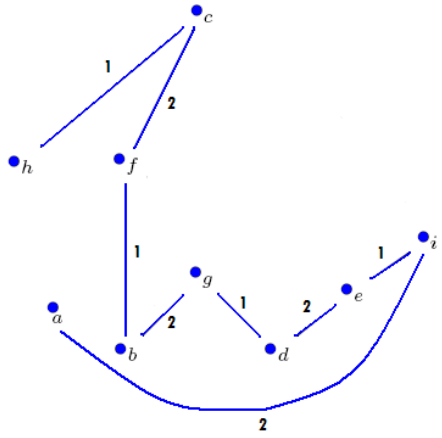
Although there are two edges that have weight two at this point it doesn't matter which is chosen and this will be seen shortly, so edge *de* is.

Step Four



The next edge with least weight is *ei* so this one is chosen to be in the minimum spanning tree.

Step Five	Step Six
	
We once again have two edges of weight two to choose from and as before either edge will do so edge *ia* is chosen.	The next edge that needs to be chosen is edge *gb*, it can now be seen even though it was looked over in previous steps it is now being used.
Step Seven	**Step Eight**
	
The next edge used is *bf* with a weight of one.	Then next edge to be chosen is edge *fc* of weight two.

Step Nine	Final MST
	
And the final edge that needs to be included in the minimum spanning tree is edge *ch* of weight one.	And the final minimum spanning tree found using Prim's algorithm also has a total weight of 12.

So by performing Prim's Algorithm's twice on the same graph, using different starting vertices, it can be seen that wherever the algorithm is started, the algorithm will return minimum spanning trees with equal weight. Prim's algorithm yields the minimum spanning tree when implemented on a connected graph and obtains the minimum weight regardless of the initial vertex selection [14].

4.3 Euclidian Minimum Spanning Trees in Monte Carlo

Prim's algorithm can be applied to Monte Carlo techniques. A graph acted upon by this algorithm will be created using the particle positions at the beginning of each cycle as the vertices of the graph. The graph will be complete such that each of the vertices will be connected by an edge to every other vertex. The weight of each edge will be the Euclidian distance calculated by equation (3.8)

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}, \tag{3.8}$$

where (x_1, y_1, z_1) and (x_2, y_2, z_2) are the coordinates of vertices. Prim's algorithm will then be implemented to find the Euclidian minimum spanning tree (EMST) of the complete graph created from the particle positions. From Prim's algorithm, the total cost of the EMST can be obtained. The total cost of the EMST may be used to determine convergence. Because the EMST of the model is derived from particle positions within the volume of the simulation, the total cost of the EMST is indicative of the state of the source distribution in each cycle. By tracking the behavior of the total cost of the EMST from cycle to cycle, where "cycle" refers to the cycle in Monte Carlo source iteration, it can be determined when the total cost of the EMST begins to fluctuate around an equilibrium value, reaching convergence. When the total cost of the EMST reaches a state of stationarity, the source distribution can be said to be in a convergent state because the total cost of the EMST is dependent on the overall particle distribution of the system. Two-dimensional and three-dimensional examples of an EMST that were found using Prim's algorithm on a particle source distribution in homogeneous models are shown in Figure 4.5, Figure 4.6 and Figure 4.7.

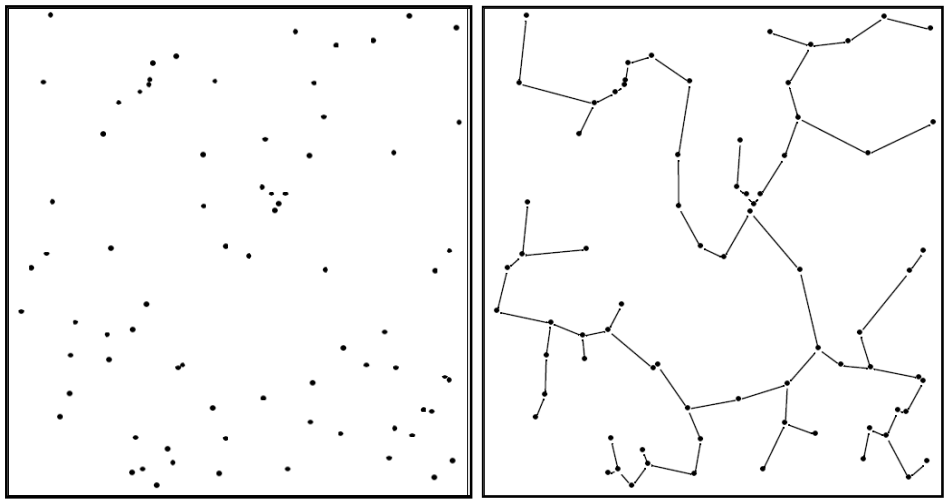


Figure 4.5: 2D 75 Point EMST With and Without Edges

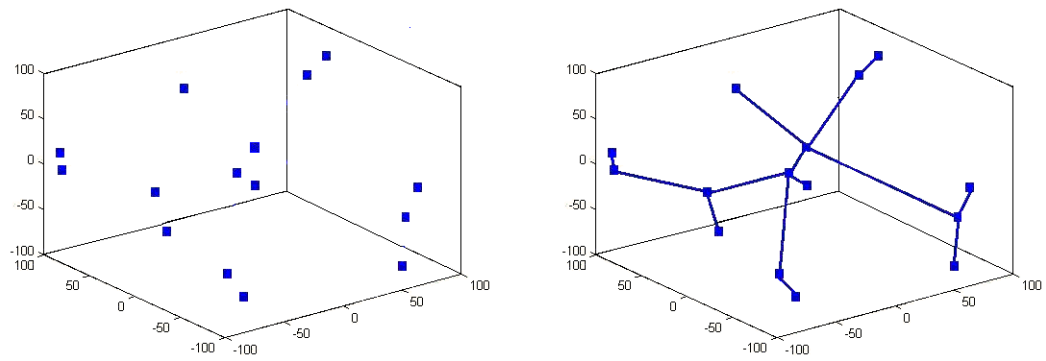


Figure 4.6: 3D 25 Point EMST With and Without Edges

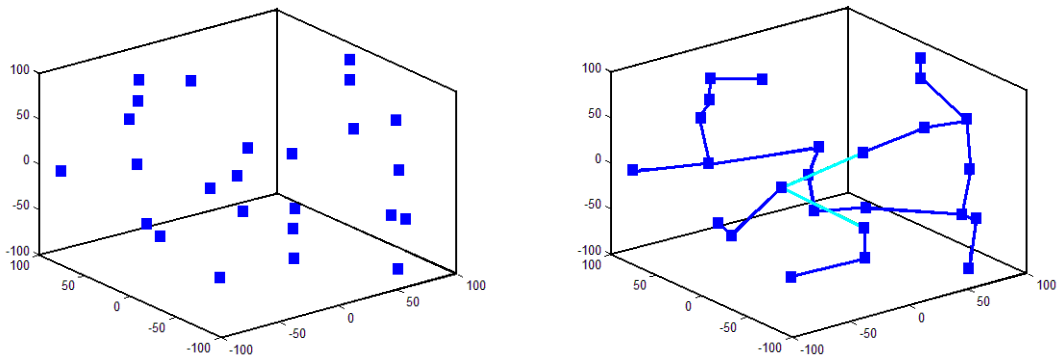


Figure 4.7: 3D 25 Point EMST With and Without Edges

The calculation of the total cost of the EMST can be done for as little or as many cycles as needed to determine convergence. In addition to use of the total cost of an EMST for convergence indication, the average edge length of an EMST may be used to evaluate the acceptable minimum tally cell volume in power distribution calculations. The use of EMST's and graph theory in Monte Carlo simulations can be very useful in eliminating the need for bin creation and thus bin-based determination of convergence and tally cell resolution. However, implementing the use of EMST's in Monte Carlo simulations has drawbacks and limitations.

One limitation arises because of the computational time or cost of implementing Prim's Algorithm on a graph created from a distribution of particles. The time it takes to

implement Prim's algorithm increases to the order N^2, where N is the number of particles. This can be seen in Figure 4.8.

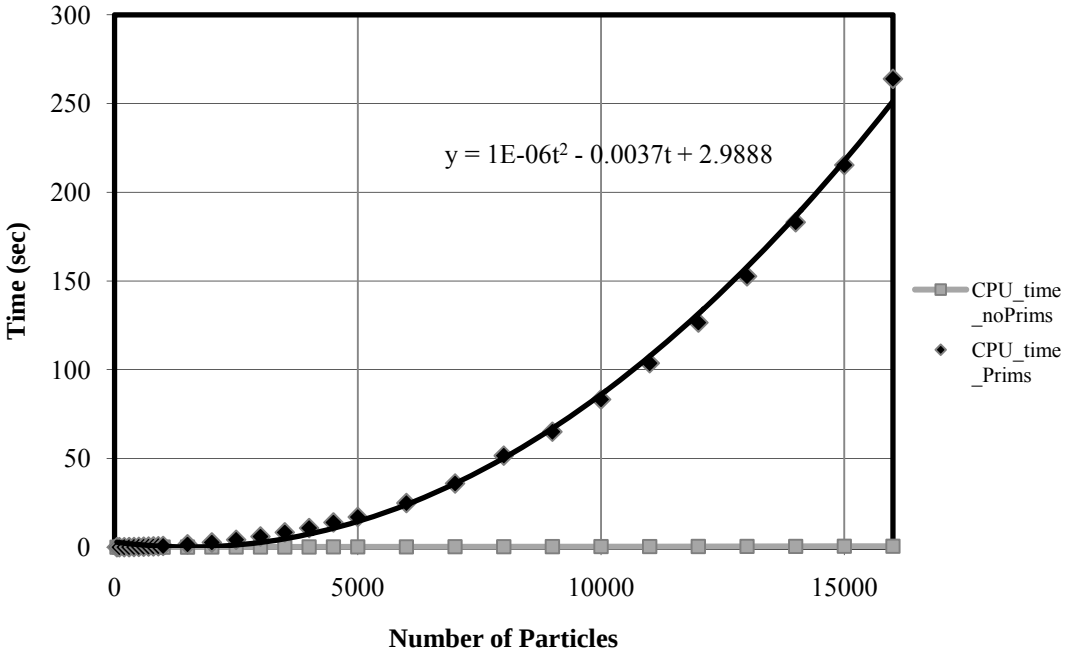


Figure 4.8: Computational Time Increase from Implementing Prim's Algorithm

In this figure the time needed to calculate the Euclidian minimum spanning tree of a system by using Prim's algorithm is for a single cycle. This is problematic because to adequately model a nuclear system with Monte Carlo methods, a large number of particles are needed; sometimes greater than 100,000 particles. Monte Carlo analysis of complex problems that require a large particle population can often take days without the addition of Prim's algorithm. Using the total cost of an EMST as a convergence indicator or finding the average edge length of an EMST for a problem with a large particle population would be counter-productive and even more time consuming if these calculations needed to done at every cycle. If Prim's algorithm were implemented in its current form, it would take far too long to perform the determination of the convergence of the source distribution, determine an average edge length, and to perform the

simulation of a model. Another limitation is the number of particles that can be used in the computation of the Euclidian minimum spanning tree. In the coding of Prim's algorithm a distance matrix, which describes the distance of every particle to every other particle, there exists a run-time memory shortage of dynamic allocation of edge matrix in Fortran 90 for large numbers of particles. Because of this run-time memory shortage, the number of particles that can be used in Prim's algorithm is limited to about 16,000 particles, which is why this is the upper limit of the number of particles in Figure 4.8.

4.4 Power Law Approximation of EMST Average Edge Length

Because of the run-time memory storage limits and large requirement of computational time for the implementation of Prim's algorithm for a large number of particles, an approximation that could extend the usefulness of EMST is needed. This approximation could eliminate the need for dynamic allocation of the edge matrix. An approximation of the average edge length past these limits for use as a particle population diagnostic was chosen to be investigated. The average edge length of an EMST at the first or final active cycle can be used as a criterion for determining minimum tally cell resolution.

A simple scaling law for the average edge length of an EMST with respect to the number of particles is proposed.

$$AEL(N) = \frac{L_{EMST}(N)}{N-1} \tag{3.9}$$

In this relationship $L_{EMST}(N)$ is the total weight of the EMST created from *N* particles. So the average edge length of an EMST with *N* particles, $AEL(N)$, is equal to the total weight of the EMST divided by the total number of edges, *N*-1. Let *Vol* be the volume of the domain. In a three-dimensional medium of uniform composition, the volume per particle at position $\vec{r}$, $V_N(\vec{r})$, is inversely proportional to *N* ($V_N(\vec{r}) = c(\vec{r}) \times Vol / N$), where $c(\vec{r})$ is the distribution function of the particles in terms of number density. From this relationship, it can be inferred that the expected average edge length in any sub domain, will be inversely proportional to $N^{1/3}$ in ideal cases. From this, the following power law can be imposed on $AEL(N)$.

$$AEL(N) \propto N^{-1/b} \tag{3.10}$$

From this proportionality, we can solve for what *b* may be equal to for any problem.

$$\begin{aligned} x &= aN^{-1/b} \\ y &= aN'^{-1/b} \end{aligned} \tag{3.11}$$

In equation (3.11) *x* and *y* represent the average edge lengths found for two different EMST's found with *N* and *N'* vertices respectively. By dividing the first equation by the second one can arrive at equation (3.12).

$$\frac{x}{y} = \left(\frac{N}{N'}\right)^{-1/b} \tag{3.12}$$

Then the log of both sides is taken and the equation in (3.13) is solved for *b*.

$$\ln\left(\frac{x}{y}\right) = -\frac{1}{b}\left(\frac{N}{N'}\right)$$
$$b = -\frac{\ln\left(N/N'\right)}{\ln\left(x/y\right)} \tag{3.13}$$

From the theory that developed equation (3.10), b is ideally equal to 3. The departure of b from 3 may depend on the heterogeneity of the system. For example in systems with fissile and non-fissile domains, particles can be born in only fissile regions. To create an EMST from the particle positions, edges must extend past non-fissile regions to connect particles in fissile regions. Extending across non-fissile regions increases the average edge length of the EMST thus inducing the departure of b from 3. Because 3 is a theoretical value, the value of b needs to be determined for different cases to be able to approximate the average edge length of an EMST for a large batch of particles. The average edge length can then be used to define a criterion to decide if the tally cell resolution of a problem is appropriate for power distribution estimation.

In Chapter 7, it will be determined if the power law truly can describe the average edge length of an EMST created by particle positions within a system for a large number of particles. The soundness of an EMST approach to population diagnostic will then be discussed. Then in Chapter 7, after verifying the reliability of the EMST approach, the criterion for the tally diagnostic will be developed, discussed, implemented and validated against other diagnostics.

Chapter 5

Step-Refined On-The-Fly Diagnostic of Convergence

In addition to the convergence indicators, an on-the-fly test needs to be performed during the process of iteration to determine at which cycle the convergence indicators have reached a stationary state. Ueki proposed and demonstrated the use of the Wilcoxon rank sum as a successful means of detecting convergence [3].

To utilize this method, the convergence indicators must be manipulated into a form that can be useful for analysis with this test. First we find the difference of the convergence indicator from one cycle to the next and for simplicity the regionwise average position indication (*RAPI*) value will be used for this explanation. *RAPI* is the sum of the distances of the regionwise average particle positions at each cycle to corresponding regionwise average particle positions in the first cycle. By regionwise correspondence, it is meant the correspondence between the average particle position in the top-upper-right region at cycle *j* and the average particle position in the top-upper right region at the 1st cycle. This operation will be performed on the other convergence

indicators with the purpose of comparison and validation of the *RAPI* method. This difference is found in equation (3.14), where j is the current cycle.

$$\Delta RAPI(j) = RAPI(j) - RAPI(j-1) \tag{3.14}$$

Differencing significantly reduces autocorrelation when the cycle is well into stationarity [3]. Due to the assumption of stationarity that the source distribution is in the equilibrium fluctuation range, the expected value of the $\Delta RAPI(j)$ during stationarity is zero and this relation is found in equation (3.15).

$$E\left[\Delta RAPI(j)\right] = 0 \text{ if cycle } j \text{ is in stationarity} \tag{3.15}$$

The expected value of $\Delta RAPI(j)$ is equal to zero because during stationarity the source distribution should ideally be constant. But due to statistical fluctuations in the source distribution during stationarity from cycle to cycle, $\Delta RAPI(j)$ will be fluctuating around zero when the source distribution has reached a convergent state. *RAPI* will have a generally increasing trend as the centers calculated in subsequent cycles move away from the centers determined in the first cycle, until the source distribution begins its fluctuation around the true source distribution. Because of the generally increasing trend before stationarity, the prevailing value of $\Delta RAPI(j)$ will be greater than zero before the source distribution reaches a convergent state.

$$\Delta RAPI(j) > 0 \text{ prevalent before cycle } j \text{ is in stationarity} \tag{3.16}$$

During stationarity an opposite trend will appear, and this relationship is found in equation (3.17).

$$\begin{pmatrix} \Delta RAPI(j) < 0 \\ \Delta RAPI(j) > 0 \end{pmatrix} \text{ equally likely to occur through cycle } j \text{ during stationarity} \quad (3.17)$$

Each of the positive and negative values of $\Delta RAPI(j)$ are equally likely to occur during stationarity. The Wilcoxon rank sum can be used to detect the appearance of the trend found in equation (3.17) [3].

To utilize the Wilcoxon rank sum, M samples of $\Delta RAPI$ from preceding cycles need to be taken separated by L cycles. The samples $\Delta RAPI(j), \Delta RAPI(j-L)$, $..., \Delta RAPI(j-(M-1)L)$ are then ordered from smallest to largest in magnitude according to their absolute value. This ordered set is then denoted by $\{Q(k)\}_{k=1}^{M}$ as specified in equations (3.18) and (3.19).

$$|Q(1)| \le |Q(2)| \le ... \le |Q(M)| \quad (3.18)$$

$$\begin{aligned} &\{\Delta RAPI(j), \Delta RAPI(j-L), ..., \Delta RAPI(j-(M-1)L)\} \\ &= \{Q(1), Q(2), ..., Q(M)\} \end{aligned} \quad (3.19)$$

With these values then let

$$Rank: \ R_k \equiv k \quad (3.20)$$

and

$$\text{Sign indicator: } V_k \equiv \begin{cases} 0 \ , \text{ if } Q(k) < 0. \\ 1 \ , \text{ if } Q(k) \ge 0. \end{cases} \quad (3.21)$$

The Wilcoxon positive signed rank sum [4] is then defined in equation (3.22).

$$W(j) \equiv \sum_{k=1}^{M} R_k V_k \tag{3.22}$$

This is the first portion of the Wilcoxon rank sum. Then, the negative sign indicator is defined in equation (3.23),

$$\text{Negative sign indicator: } V_k^- \equiv \begin{cases} 1\ , \text{ if } Q(k) < 0. \\ 0\ , \text{ if } Q(k) \geq 0. \end{cases} \tag{3.23}$$

The Wilcoxon negative signed rank sum is defined in equation (3.24) as follows.

$$W^-(j) \equiv \sum_{k=1}^{M} R_k V_k^- \tag{3.24}$$

The sum of Wilcoxon positive signed rank sum and the negative signed rank sum is constant:

$$W(j) + W^-(j) = \sum_{k=1}^{M} \left(V_k + V_k^-\right) R_k = \frac{M(M+1)}{2} \tag{3.25}$$

By combining the trends in equations (3.16) and (3.17) with the definitions from above, general trends of the Wilcoxon rank sums are obtained. These trends are as follows.

$$W(j) > W^-(j) \text{ prevalent for cycle } j \text{ prior to convergence} \tag{3.26}$$

and

$$\begin{gathered} W(j) > W^-(j) \text{ and } W(j) < W^-(j) \\ \text{are equally likely to occur for cycle } j \text{ during stationarity} \end{gathered} \tag{3.27}$$

The shift in trend of the Wilcoxon positive rank sum depicted in the above declarations, indicates the transition into stationarity. By taking into account equation (3.25), the trends in statements (3.26) and (3.27) can be reinterpreted as the following.

$$W(j) > \frac{M(M+1)}{4} \quad \text{for cycle } j \text{ prior to convergence} \tag{3.28}$$

and

$$W(j) > \frac{M(M+1)}{4} \quad \text{and} \quad W(j) < \frac{M(M+1)}{4} \tag{3.29}$$
$$\text{are equally likely to occur for cycle } j \text{ during stationarity}$$

The constant value $\frac{M(M+1)}{4}$ is described as the median of the Wilcoxon rank sum and will be identified as $W_{mid}(M)$.

5.1 Step Refined On-the-Fly Diagnostic

Derived from the trend transition from declaration (3.28) to declaration (3.29), Ueki proposed the following "step-refined on-the-fly diagnostics of the source distribution convergence" to determine when the source distribution has reached the fluctuation range of stationarity [3].

5.1.1 Step One

In the first step samples of *RAPI* are taken from twenty cycles preceding the current cycle j for $j \geq 21$. In this step L and M in equation (3.19) are chosen to be 3 and 7 respectively. After obtaining the values of the Wilcoxon positive rank sum for cycles $j \geq$

21, the moving average of $W(j-2)$, $W(j-1)$ and $W(j)$ of current cycle j needs to be determined. After obtaining the moving averages, it needs to be determined at which cycle the moving average crosses the median of the Wilcoxon rank sum ($W_{mid}(7)=14$). At the cycle the moving average crosses the median value for the first time, the declaration of the convergence cycle is made denoted as $j_S(1)$. Then let $j_C = j_S(1)$ which is the cycle at which the moving average crossing the median is checked.

5.1.2 Step Two

In step two, samples of *RAPI* are simultaneously being taken from 40 cycles preceding the current cycle j for $j \geq 41$. In this step L and M from equation (3.19) are taken to be 4 and 10 respectively. At cycle $j_S(1)+20$, check whether the moving average of $W(j-3)$, $W(j-2)$, $W(j-1)$ and $W(j)$ crossed the median of the ranked sum $\left(W_{mid}(10)=27.5\right)$ before $j_S(1)+20$. [The number of preceding cycles is increased by 20, thus the judgment is made 20 cycles after $j_S(1)$.] If this crossing has occurred prior to this cycle let $j_S(2)=j_S(1)+20$. In this case, j_C is not updated. If the crossing has not occurred at $j_S(1)+20$, continue to check the subsequent cycles to determine at which cycle the crossing occurs. Once the crossing has occurred, let $j_S(2)$ be equal to the cycle of the first crossing. Then set $j_C = j_S(2)$.

5.1.3 Step Three

In step three samples of *RAPI* are being concurrently taken from 60 cycles preceding the current cycle j for $j \geq 61$. In this step L and M from equation (3.19) are taken to be 5 and 12 respectively. At cycle $j_S(2)+20$, check whether the moving average of $W(j-4)$, $W(j-3)$, $W(j-2)$, $W(j-1)$ and $W(j)$ crossed the median of the ranked sum $\left(W_{mid}(12)=39\right)$ before $j_S(2)+20$. [The number of preceding cycles is increased by 20, thus the judgment is made 20 cycles after $j_S(2)$.] If this crossing has occurred prior to this cycle, let $j_S(3)=j_S(2)+20$, if this is the case j_C will not change as said before. If the crossing has not occurred at $j_S(2)+20$, continue to check the subsequent cycles to determine at which cycle the crossing occurs. Once the crossing has occurred, let $j_S(3)$ be equal to the cycle of the first crossing. Then set $j_C = j_S(3)$.

5.1.4 Step Four

In step four, samples of *RAPI* are simultaneously being taken from 80 cycles preceding the current cycle j for $j \geq 81$. In this step L and M from equation (3.19) are taken to be 5 and 16 respectively. At cycle $j_S(3)+20$, check whether the moving average of $W(j-4)$, $W(j-3)$, $W(j-2)$, $W(j-1)$ and $W(j)$ crossed the median of the ranked sum $\left(W_{mid}(16)=68\right)$ before $j_S(3)+20$. If this crossing has occurred prior to this cycle, let $j_S(4)=j_S(3)+20$. If the crossing has not occurred at $j_S(3)+20$, continue to check the subsequent cycles to determine at which cycle the crossing occurs.

Once the crossing has occurred, let $j_S(4)$ be equal to the cycle of the first crossing. Then set $j_C = j_S(4)$.

5.1.5 Step Five and Beyond

In steps five and later, samples of *RAPI* are simultaneously being taken from 20 X *A* cycles preceding the current cycle *j*, where *A* is the step number (*A*=5,6,..) and $j \geq 20$ X $A + 1$. In these steps *L* is always taken to be 5 and *M* is taken to be 4 X *A*. At cycle $j_S(A-1)+20$, check whether the moving average of $W(j-4)$, $W(j-3)$, $W(j-2)$, $W(j-1)$ and $W(j)$ crossed the median of the ranked sum $(W_{mid}(M))$ before $j_S(A-1)+20$. If this crossing has occurred prior to this cycle, let $j_S(A) = j_S(A-1)+20$. In this case j_C is not updated. If the crossing has not occurred at $j_S(A-1)+20$, continue to check the subsequent cycles to determine at which cycle the crossing occurs. Once the crossing has occurred, let $j_S(A)$ be equal to the cycle of the first crossing. Then set $j_C = j_S(A)$ (j_C is updated). Once again remember that $j_S(A)$ is in fact the current cycle in the cycle progression as with the previous steps.

5.1.6 Stopping Criteria

The steps will continue until the convergence cycle has been identified. Criteria for stopping of the step-refined on-the-fly diagnostic of the source distribution convergence need to be defined. The stopping criteria are as follows.

a. $\left[\begin{array}{c}\text{The number of the cycles preceding}\\ \text{the current cycle, from which}\\ \text{Wilcoxon rank sum is computed}\end{array}\right]$

$$> \left[\begin{array}{c}\text{66\% of the declared}\\ \text{convergence cycle}\\ \text{number}\end{array}\right].$$

Or,

$$20\times A > 0.66\times j_C$$

b. Same declared convergence cycle (j_C) at five consecutive steps

In criterion a, A is the step number and j_C is the declared convergence cycle. This criterion means that the diagnostics are stopped if the number of cycles immediately preceding the current cycle used to compute the Wilcoxon rank sum is greater than two-thirds of the number of cycles preceding the declared cycle of convergence, j_C. This criterion is in place for rapid and regular-convergence cases. The second criterion is for cases where the convergence tends to be sluggish. Once either of the criterion are met, the step-refined diagnostic analysis will halt, and the beginning of the active cycles will be declared as j_C.

5.2 Alternative Explanation of Step Refined On-the-fly Diagnostic

The step refined on-the-fly diagnostic will be explained once again in a different manner for the ease of understanding.

Take samples of $RAPI$ from cycles preceeding cycle j

The number of preceeding cycles is determined from the product of L and M

$L \times M$ = Number of Preceeding cycles

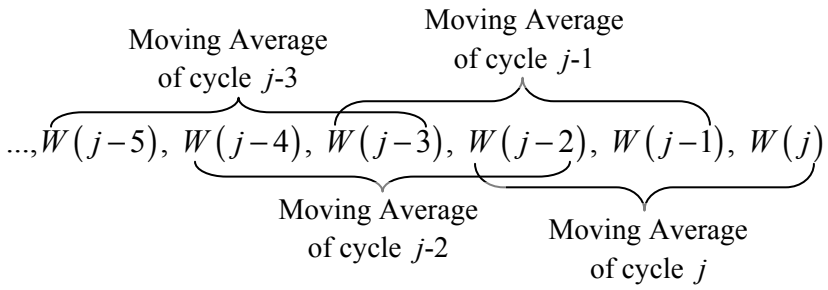
Samples are taken every L cycles, For example if $L = 3$

$$\underbrace{RAPI(j)}_{Sample}, RAPI(j-1), RAPI(j-2), \underbrace{RAPI(j-3)}_{Sample}, RAPI(j-4), \ldots$$

From this sampling Wilcoxon Rank Sum will be calculated for cycle j

For each step take the moving average of L Wilcoxon Rank Sums

For example with $L = 3$



Determine at which cycle does the moving average cross the median value $W_{mid}(M)$ of current step

The stepping is done as follows:

Step (i)	Cycle of first downward crossing	Cycle at which decision was made ($j_S(i)$)	Declared convergence cycle (j_C)
1	85	85	**85**
2	93	105	**85**
3	132	132	**132**

In step one the first crossing occurs at cycle 85, so $j_S(1)$ is set equal to 85 and the convergence cycle for the first step is equal to 85 as well. In the second step, it is determined if the crossing happens before $j_S(1) + 20$, and it does at the 93rd cycle. Since this is so, $j_S(2) = j_S(1) + 20$ and j_C remains the same for the second step. In the third step the first downward crossing happens after $j_S(1) + 20$, so $j_S(3)$ is set equal to the first downward crossing, as is j_C. The stepping continues until the stopping criteria are met.

Employing this diagnostic tool for the determination of the convergence cycle is superior to other methods in the following aspects. The step-refined nature of this process makes it superior because it increments the number of samples taken for the calculation of the Wilcoxon rank sum, ensuring that the declared cycle of convergence becomes more reliable as cycles progress. It is also on-the-fly, because the diagnostic takes samples from cycles previous to the current cycle and finds if the convergence criteria have been meet at the current cycle for a number of steps. If the criteria have not been met at the current cycle, the process continues stepping through the cycles until the criteria have been satisfied and a cycle of convergence is declared. Once either of the criteria is met, the diagnostic declares that the convergence of the source distribution is achieved and the process ceases to function and tallies can began to be recorded.

This step-refined on-the-fly diagnostic of the source distribution convergence shall be implemented using *RAPI* values, discussed in Chapter 3, to determine the convergence cycle of various simulations. The same diagnostic will then be performed with the entropy indicators, discussed in Chapter 4, to obtain the convergence cycle for the same simulations. The resulting convergence cycle obtained from the step-refined on-the-fly diagnostic utilizing both *RAPI* values and entropy values will be compared to validate the practicality of the *RAPI* method as well as the usefulness of the combination of the *RAPI* method and the step-refined on-the-fly diagnostic for convergence determination. The discussion of the numerical results and functionality of this diagnostic will be presented in Chapter 6.

Chapter 6

Numerical Results of Convergence Determination

The *RAPI* method can be used as an indicator of convergence of the source distribution, because the *RAPI* uses the particle positions of the source distribution to obtain its values. The cycle of convergence can then be determined by a step-refined on-the-fly diagnostic of the *RAPI* by Wilcoxon rank sum. The cycle of convergence that is determined using the values obtained from the *RAPI* needs to be validated to see if the *RAPI* is a good indicator of the convergence of the source distribution. This can be done by comparing the convergence cycle obtained from *RAPI* to the convergence cycle that is found by previous methods shown to be able to determine convergence. Previous work done by Ueki has shown that convergence indicators of monotonic trend, such as progressive relative entropy in combination with a step-refined on-the-fly diagnostic of the source distribution has been able to determine when the source distribution has reached a fluctuation around the true source distribution or a stationary state [3]. The determination of the convergence cycle was performed using values obtained by the *RAPI* method for three cases. These results were then compared against the convergence cycle determined

using the Shannon entropy, the Jensen measure, the progressive relative entropy and the posterior relative entropy for the same three cases.

The three cases for which convergence cycle determination was performed for was a homogeneous cube, Whitesides k_{eff} the world problem [15] and fuel storage vault of fresh PWR fuel assemblies [3]. As mentioned in chapter 3, the effectiveness of the *RAPI* may be affected by the dominance ratio and the heterogeneity of the problem, and these effects need to be investigated. The homogeneous cube was done first because it is a simple problem, homogeneous medium, with a high dominance ratio (DR=0.999) [16]. Investigating this problem will indicate the performance of the *RAPI* and how it is affected by a high dominance ratio, without having to worry about the effects on *RAPI* from the heterogeneity of the problem. The second problem investigated was Whitesides k_{eff} of the world problem, which is a 9x9x9 array of plutonium spheres surrounded on all sides by a thick water reflector; the center sphere is supercritical, the rest are subcritical spaced on 60-cm centers [15]. The difficulty in calculating the k_{eff} accurately arises because commonly used sampling may not account for the contribution of the supercritical sphere in the center to the calculation of the k-effective. In *A Difficulty in Computing the k-effective of the World*, Whitesides explains that "erroneous results for these types of problems are the result of the failure of the calculation to converge to the fundamental source mode" [15]. This problem has a lower dominance ratio than the homogeneous cube (DR=0.73) [1], and it has a high degree of heterogeneity. Using *RAPI* to determine the convergence cycle of this problem will test *RAPI* performance in heterogeneous problems while reducing the effects the dominance ratio has on *RAPI*. The final problem was a PWR fresh fuel vault. This problem is heterogeneous and has a

high dominance ratio. The dominance ratio was computed to be 0.9943, for the horizontal first and vertical fundamental mode, and 0.9934, for the horizontal fundamental and first vertical mode [3]. By examining this problem, *RAPI* performance will be tested for both heterogeneous and high dominance ratio effects. This final case will also be done to see if the *RAPI* can be used for real-world problems.

6.1 Homogeneous Cube

The first problem investigated was a homogeneous cube with dimensions -100 cm to 100 cm in x, y and z. The problem and the material properties are displayed in Figure 6.1.

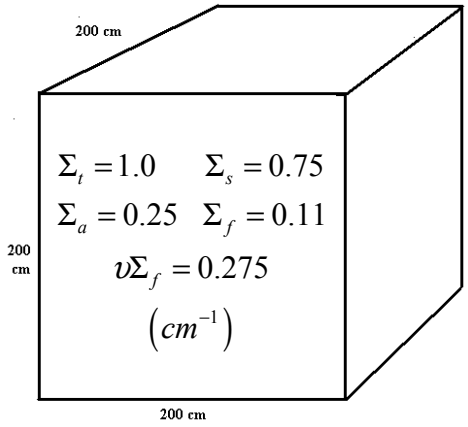


Figure 6.1: Homogeneous cube Simulation

There are one thousand bins used in this problem or 10 bins in the x, y and z direction. The initial source distribution is uniform throughout the whole geometry. The simulation was run with 10,000 particles for 1,400 cycles and 400 inactive cycles. Figure 6.2 shows the trend of the calculation of the *RAPI*. It appears that *RAPI* may converge around 850 cycles or iterations but the *RAPI* oscillates too greatly to be considered converged at this point. If we compare this graph to one of the entropies in the system, say the Jensen

measure (Figure 6.3), we can see that there is some inconsistency between the behaviors of the two. The Jensen measure seems to converge at about 900 iterations and does not oscillate as much as the *RAPI*.

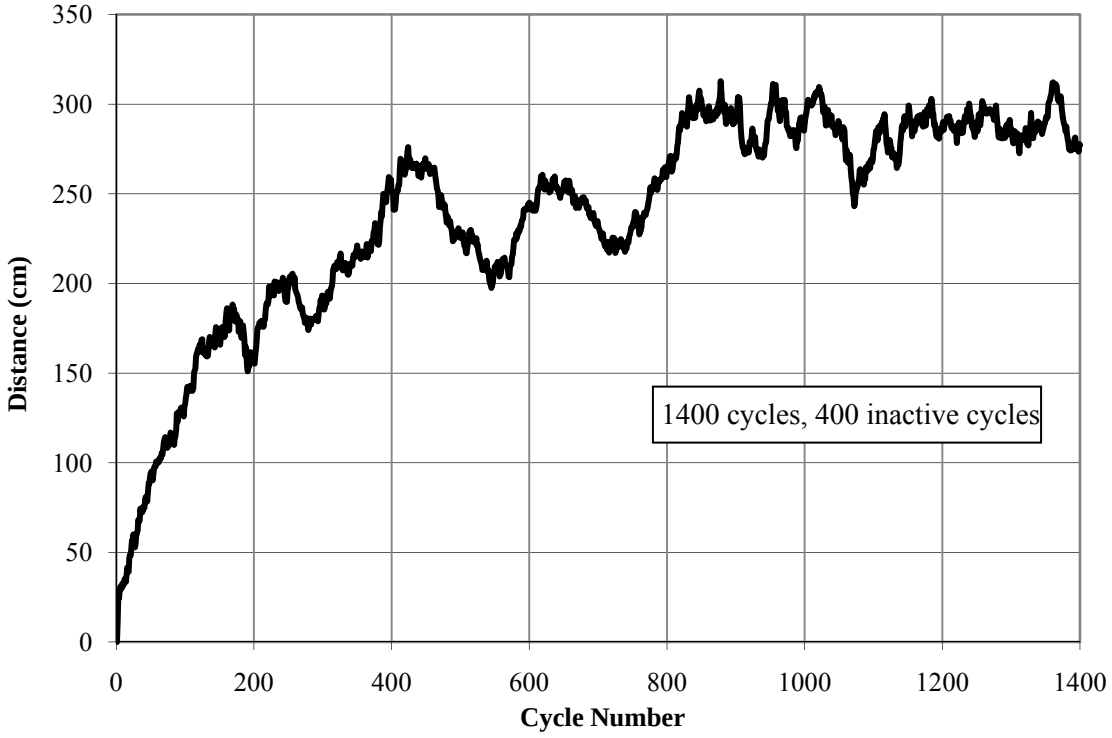


Figure 6.2: *RAPI* Distances for Homogeneous Cube with 10,000 Particles per Cycle

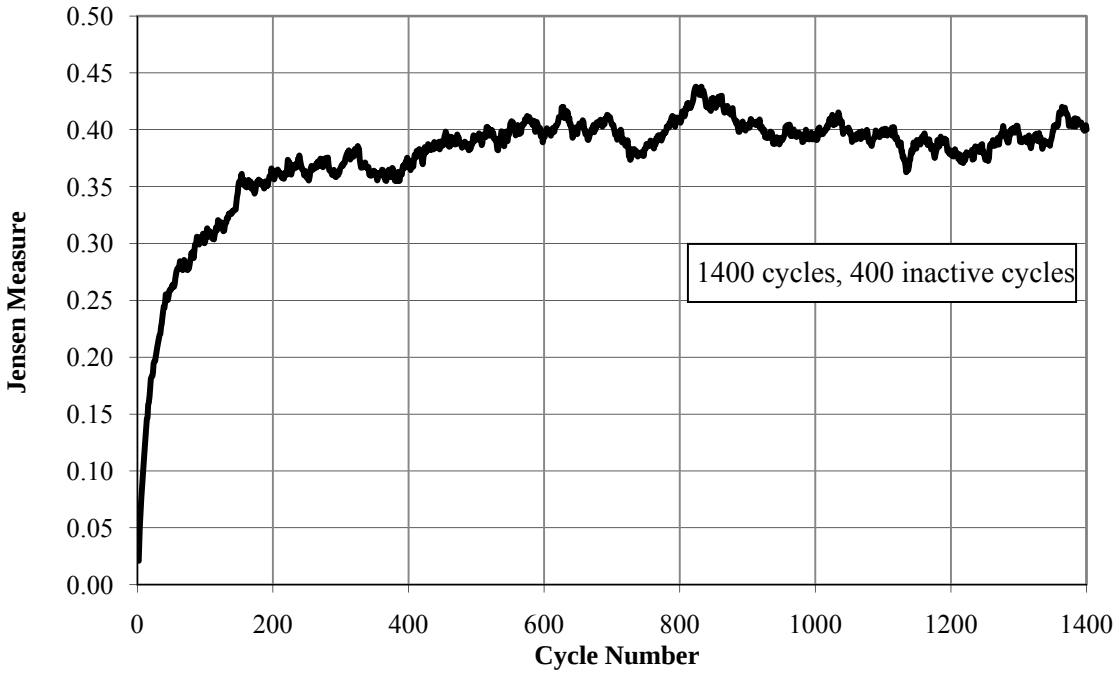


Figure 6.3: Jensen Measure for Homogeneous Cube with 10,000 Particles per Cycle

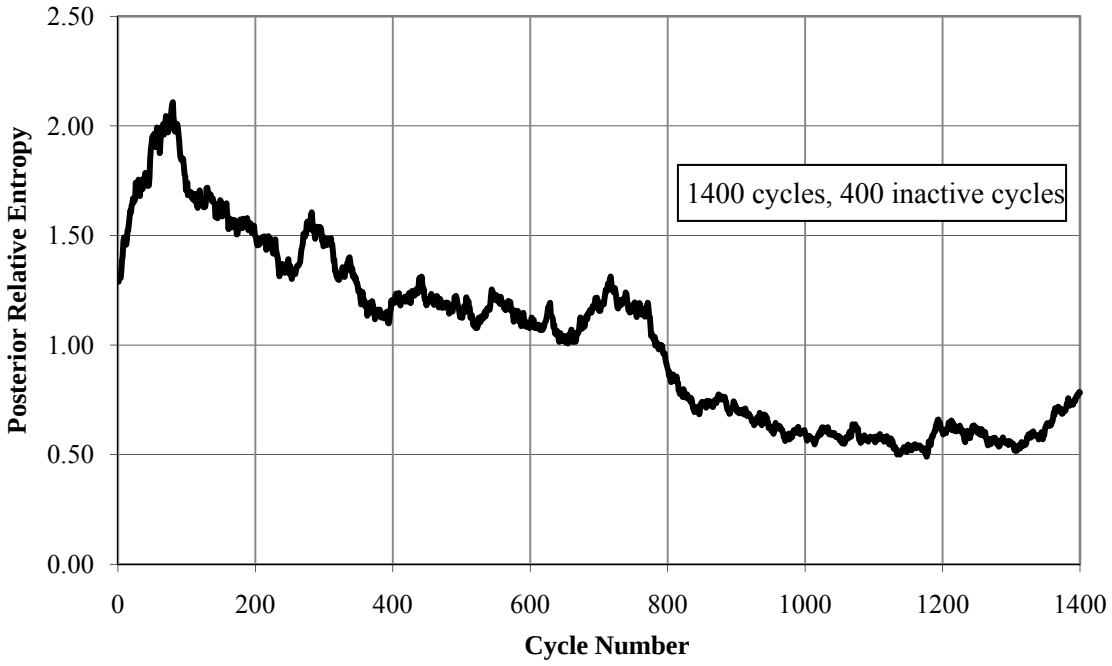


Figure 6.4: Posterior Relative Entropy for Homogeneous Cube with 10,000 Particles per Cycle

This inconsistency arises because the simulation is being undersampled. The error of undersampling occurs when there are an inadequate number of particles to

accurately model the behavior of a nuclear criticality problem. Not only is there inconsistency between *RAPI* and Jensen measure, there is inconsistency with the Shannon Entropy, progressive relative entropy, and the posterior relative entropy. The most significant irregularity in values obtained is with the posterior relative entropy. The posterior relative entropy behaved contrary to the way it should, which can be seen in Figure 6.4. In theory, the posterior relative entropy is supposed to have a monotonic (generally decreasing) trend, having its maximum value in the first iteration or cycle. In the beginning of the iterative process the posterior relative entropy increased in amount, peaked at the 79^{th} iteration, and then began to assume a general decreasing trend. Another observation is that the posterior relative entropy rapidly increases from at cycle 1,300 to 1,400, which is a sign that the convergence takes more than 1,400 cycles. Another sign that would support that convergence takes longer than 1,400 cycles is the great amount of oscillation of the *RAPI* found in Figure 6.2 from cycles 800-1,400 where there should be a plateau if the *RAPI* were in a convergent state.

Because of the inconsistency with the previous model, alterations needed to be made to the particle population to properly simulate the system, increase the total number of cycles and increase the number of inactive cycles. The number of cycles needed to be increased so the convergence behavior of the indicators of the source distribution past 1,400 cycles could be examined. The number of inactive cycles needs to be increased, because the posterior relative entropy is calculated by using the average of the source distribution over the second half of active cycles. If the source distribution has not converged by the second half of active cycles, i.e., too few inactive cycles, the trend found in Figure 6.4 may arise. By increasing the number of inactive cycles, the total

number of cycles, and the number of particles; the system should be better simulated, and the results of the convergence indicators will be more accurate. There should also be better agreement between the indicators. From this model it can possibly be inferred that the *RAPI* method may also be used as a particle population diagnostic, but this possibility needs to be further investigated and is beyond the scope of this thesis.

For the next simulation of the homogeneous cube case, the number of cycles was increased from 1,400 to 2,000. The number of inactive cycles was increased from 400 to 500 to see the effect on the posterior relative entropy calculations. Finally the number particles were increased from 10,000 to 100,000. The next figure displays how the sum of the distances or *RAPI* changed as a result. The amount of variation decreases but the *RAPI* does not converge until much later in the iterational process. This late convergence is due to the high dominance ratio of the problem. The dominance ratio of this problem is 0.999, close to unity, computed by discrete ordinate methods [16]. In cases with a high dominance ratio, the source distribution may slowly converge or may not converge to stationary state due to the autocorrelation of the MC stationary sources being strong and slowly decaying [12]. Using the *RAPI* method, the step-refined on-the-fly diagnostic of the source distribution, discussed in chapter 5, was performed for the redefined homogeneous cube base. Stopping criterion b was met at the 15^{th} step. The results of the diagnostic procedure are displayed in Table 6.1.

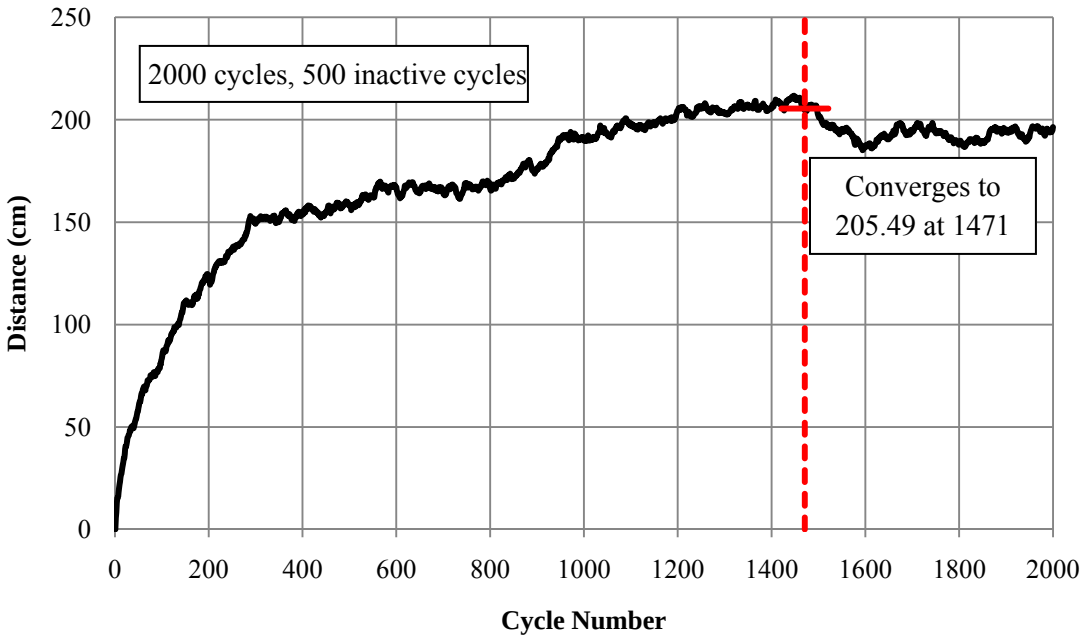


Figure 6.5: *RAPI* distances For Homogeneous Cube with 100,000 Particles per Cycle

From the definition of stopping criteria b, the step refined on-the-fly judgment process will be terminated once the same cycle of convergence has been declared for five consecutive steps. Utilizing the *RAPI* values gives a declaration of the cycle of convergence in the step 15 of 1471.

Step (i)	Cycle of first downward crossing	Cycle at which decision was made ($j_S(i)$)	Declared convergence cycle (j_C)
1	327	327	**327**
2	343	347	**327**
3	347	367	**327**
4	436	436	**436**
5	460	460	**460**
6	608	608	**608**
7	503	628	**608**
8	704	704	**704**
9	734	734	**734**
10	734	754	**734**
11	1471	1471	**1471**
12	1491	1491	**1471**
13	1500	1511	**1471**
14	1508	1531	**1471**
15	1501	1551	**1471**

Table 6.1: Determination of Convergence Cycle Using *RAPI* for Homogeneous Cube Case

The behavior of the *RAPI* through the cycles has a plateau at in the beginning of the cycles only to fluctuate again and then to converge at the 1471st cycle. The erratic behavior of the *RAPI* trend is due to the high dominance ratio. The on-the-fly judgment of the *RAPI* performed well, declaring convergence late in the iterations, when the *RAPI* appears to be in a stationary state. Once again a model with a dominance ratio close to one converges very slowly or may not converge at all. These results are indicative of the nature of problems with dominance ratios close to unity.

The *RAPI* step-refined on-the-fly results were then compared to the results obtained through the use of the four indicators of the source distribution proposed by Ueki [3]. The first comparison was against the results obtained using Jensen measure. Like the *RAPI*, the Jensen measure does plateau late in iterations, which can be seen in Figure 6.6. When the step-refined on-the-fly diagnostic is examined, stopping criteria b is not met until the 18th step (Table 6.2). The declared convergence cycle for the Jensen measure is found to be close to that found for the *RAPI* method, being 1568. Once again the late convergence of the Jensen measure is due to dominance ratio of the problem being close to unity causing the source distribution to converge slowly.

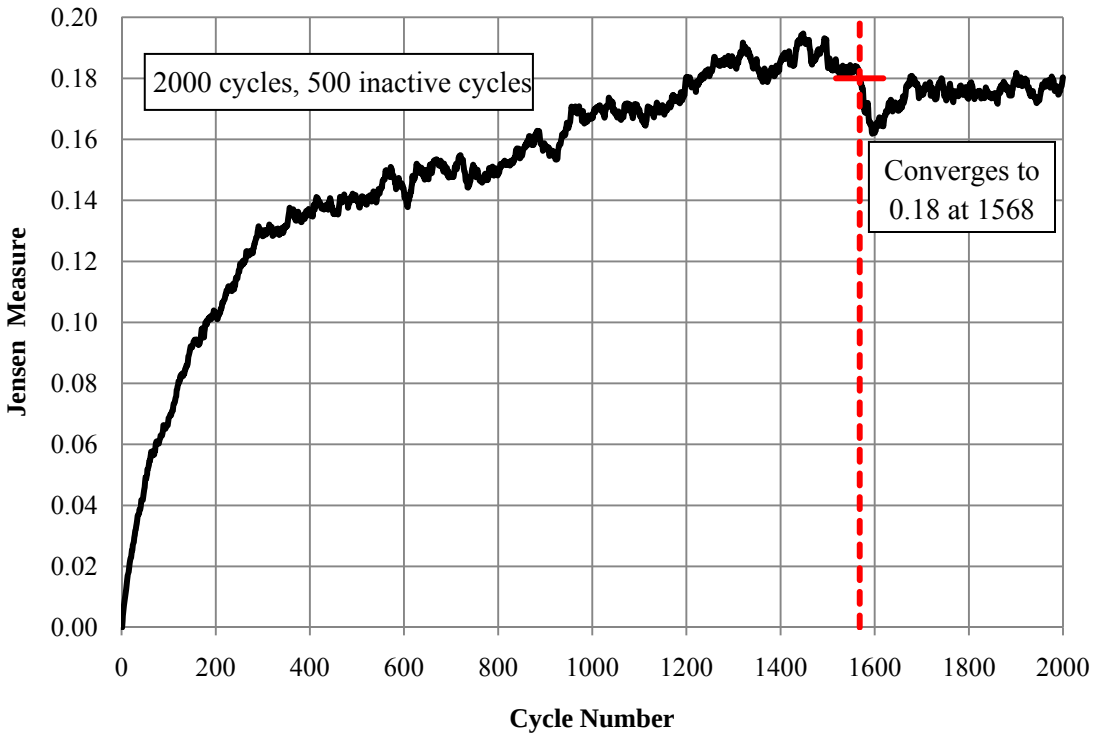


Figure 6.6: Jensen Measure for Homogeneous Cube with 100,000 Particles per Cycle

Step (i)	Cycle of first downward crossing	Cycle at which decision was made ($j_S(i)$)	Declared convergence cycle (j_C)
1	200	200	**200**
2	329	329	**329**
3	425	425	**425**
4	437	445	**425**
5	454	465	**425**
6	481	485	**425**
7	505	505	**505**
8	517	525	**505**
9	608	608	**608**
10	608	628	**608**
11	608	648	**608**
12	736	736	**736**
13	823	823	**823**
14	1568	1568	**1568**
15	1371	1588	**1568**
16	735	1608	**1568**
17	1576	1628	**1568**
18	1577	1648	**1568**

Table 6.2: Determination of Convergence Cycle Using Jensen Measure for Homogeneous Cube Case

The *RAPI* results were then compared to the results obtained using Shannon entropy as an indicator of the state of source distribution. The graph in Figure 6.7 of the Shannon entropy over all the iterations shows a similar trend that is found for the *RAPI*.

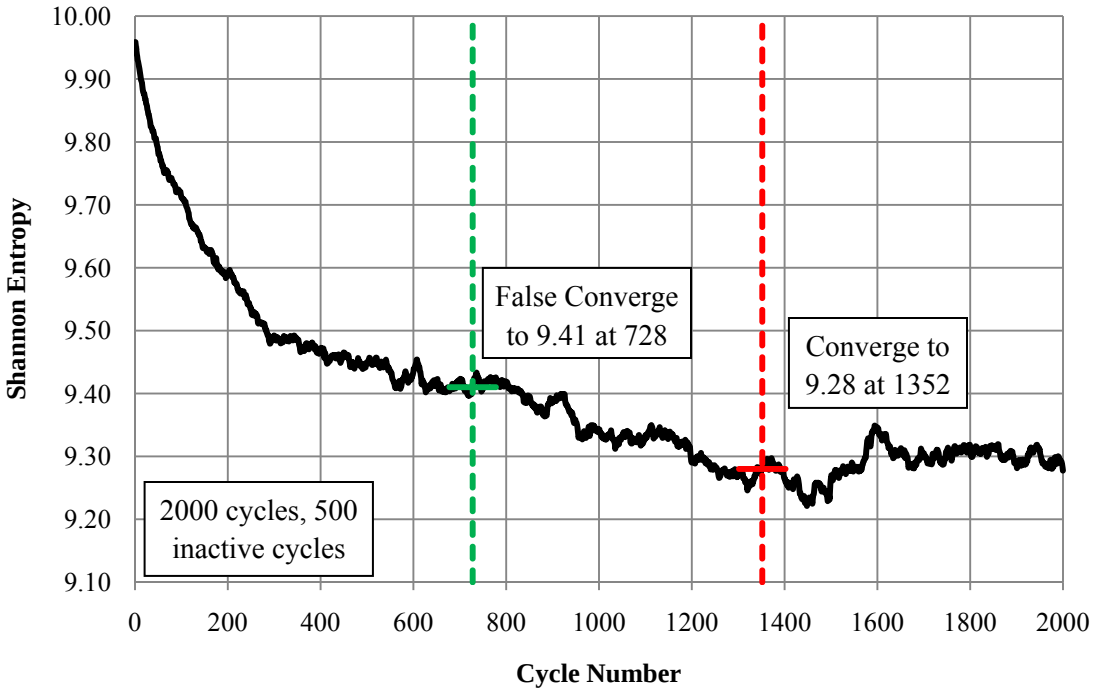


Figure 6.7: Shannon Entropy for Homogeneous Cube with 100,000 Particles per Cycle.

By looking at Table 6.3, the step-refined on-the-fly diagnostic of the Shannon Entropy gives a false declaration of the convergence cycle, meeting stopping criteria b in the 12th step. In the next step the declared convergence cycle is identified as 1352, which is closer in number to that obtained by use of the *RAPI* method, and this cycle is declared the convergence cycle in step 17. Since criteria b was already met, the step-refined diagnosis of the source distribution would have prematurely been terminated giving an incorrect convergence cycle. Comparing these results to those found using *RAPI*, one can conclude that using *RAPI* of the source distribution for convergence determination is better for a case with a high dominance ratio, but improvements or alterations to the step-

refined diagnostic may be needed. Yet again problems arise when investigating the source distribution of simulations with a high dominance ratio.

Step (i)	Cycle of first upward crossing	Cycle at which decision was made ($j_S(i)$)	Declared convergence cycle (j_C)
1	203	203	**203**
2	330	330	**330**
3	435	435	**435**
4	520	520	**520**
5	520	540	**520**
6	606	606	**606**
7	608	626	**606**
8	728	728	**728**
9	734	748	**728**
10	759	768	**728**
11	776	788	**728**
12	801	808	**728**
(13)	(1352)	(1352)	**(1352)**
(14)	(1359)	(1372)	**(1352)**
(15)	(1359)	(1392)	**(1352)**
(16)	(1355)	(1412)	**(1352)**
(17)	(1371)	(1432)	**(1352)**

Table 6.3: Determination of Convergence Cycle Using Shannon Entropy for Homogeneous Cube Case

The *RAPI* cycle determination was then compared to the results obtained through the use of the progressive relative entropy. From descriptions in Chapter 2, the progressive relative entropy is two times the Jensen measure. This can be seen by comparing Figure 6.8 and Figure 6.6. Using the progressive relative entropy (*PRE*) for the step-refined on-the-fly diagnostic of the source distribution yields the same results as the Jensen measure results, which can be seen by comparing Table 6.4 to Table 6.2. Once again stopping criteria b is met in step 18, and the declared convergence cycle is determined to be cycle 1568. The declared convergence cycle at the end of the diagnostic stepping provides similar results to the *RAPI* convergence determination. Because of the equality and similar results of the Jensen measure and the *PRE*, only *PRE* results will be discussed for successive cases.

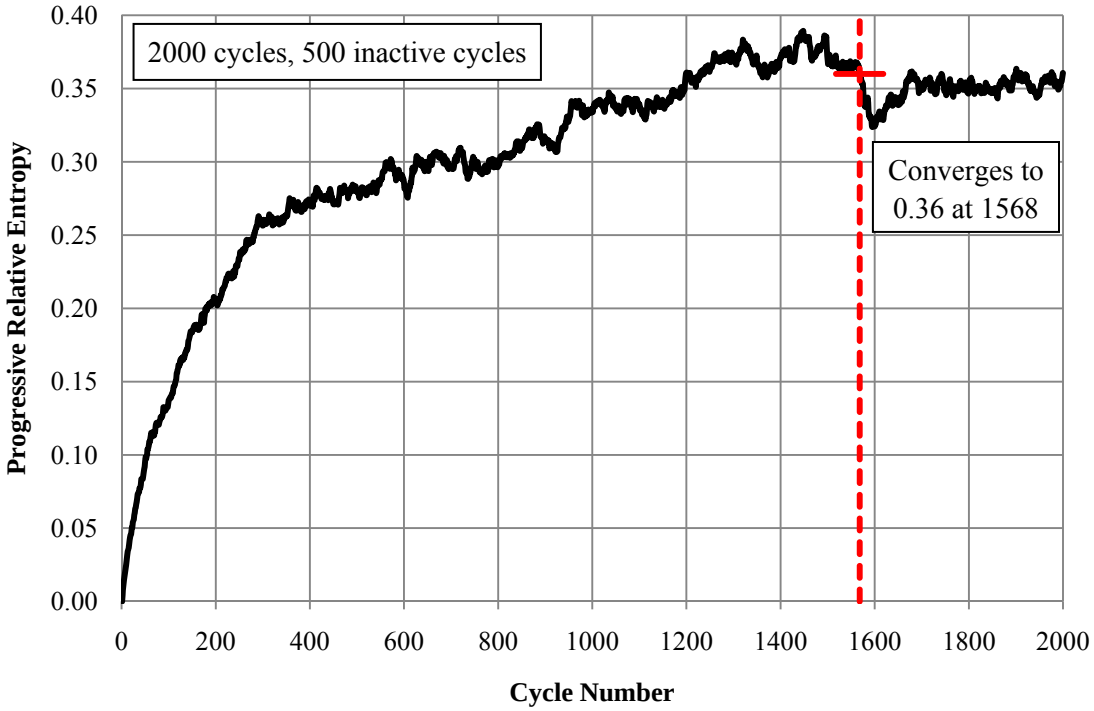


Figure 6.8: Progressive Relative Entropy for Homogeneous Cube with 100,000 Particles per Cycle.

Step (i)	Cycle of first downward crossing	Cycle at which decision was made ($j_S(i)$)	Declared convergence cycle (j_C)
1	200	200	**200**
2	329	329	**329**
3	425	425	**425**
4	437	445	**425**
5	454	465	**425**
6	481	485	**425**
7	505	505	**505**
8	517	525	**505**
9	608	608	**608**
10	608	628	**608**
11	608	648	**608**
12	736	736	**736**
13	823	823	**823**
14	1568	1568	**1568**
15	1371	1588	**1568**
16	735	1608	**1568**
17	1576	1628	**1568**
18	1577	1648	**1568**

Table 6.4: Determination of Convergence Cycle Using *PRE* for Homogeneous Cube Case

The final comparison was done against the results obtained for the convergence cycle determination using the posterior relative entropy (*PosRE*). The trend of the *PosRE*

found in Figure 6.9 is unlike that found for the *RAPI*. Its trend decreases instead of increases. The posterior relative entropy is said to converge when the posterior relative entropy reaches the average value of the posterior relative entropy over the last half of the active cycles. From Figure 6.9 the average value over the second half of active cycles is found to be 0.104 and the *PosRE* reaches this value at iteration 1362. This value is close to the result of 1471 obtained from the *RAPI*. This comparison shows that the *RAPI* can be used for source distribution convergence cycle determination of problems with a high dominance ratio, because it is close to the cycle determined using the *PosRE*.

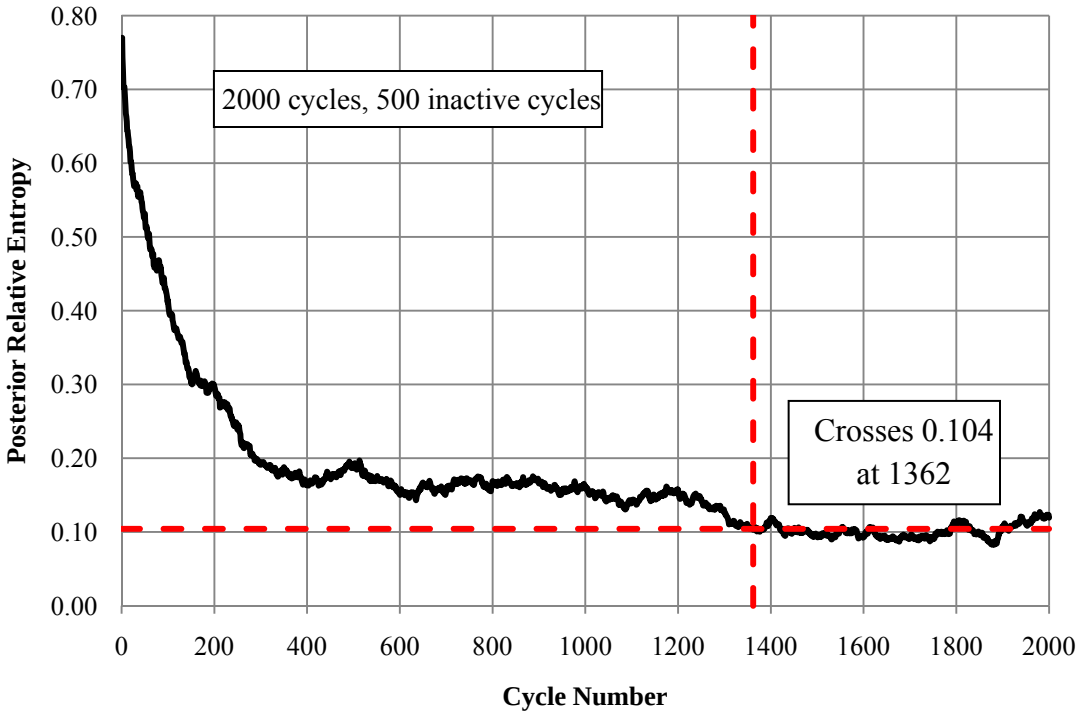


Figure 6.9: Posterior Relative Entropy for Homogeneous Cube with 100,000 Particles per Cycle

Increasing the number of particles and cycles improved the agreement between the *RAPI* trends and the entropy trends calculated for the homogenous cube case. Increasing the number of inactive cycles corrected the behavior of the posterior relative

entropy. These facts can be seen from the previous five figures. So with these results there is promise for using the *RAPI* method as a tool for determining convergence. The difficulty in determining the convergence using the entropies of the system in the homogenous case may be attributed to the high dominance ratio of the homogeneous cube simulation. Even due to this difficulty, the *RAPI* method performed well, meeting the stopping criteria without giving false confirmation of the cycle convergence unlike the Shannon entropy. By comparing Figure 6.5, Figure 6.8 and Figure 6.9, in addition to the step-refined diagnostic results in Table 6.1 and Table 6.4, *RAPI* can be as reliable as *PRE* and *PosRE* for use in on-the-fly diagnostics of simulations with a high-dominance ratio.

6.2 Whitesides k_{eff} of the World

The previous runs were done with a research code, but the next two runs were done with MCNP with the addition of the sum of the distances of the centers method or *RAPI*. The first of these two runs was for the convergence cycle calculation of Whitesides k-effective of the world problem. This problem is depicted in Figure 6.10, and it is a 9x9x9 array of plutonium spheres with radii of ~4 cm, spaced on 60 cm centers and reflected on all sides by a thick water reflector [15]. The center sphere is supercritical in the bare state and the remaining spheres are subcritical. This problem is heterogeneous and larger in dimension than the homogenous cube case. For the computation of the entropies of the system there is one bin per a sphere for a total of 729 bins. The initial source distribution is uniform throughout the fissile regions or the plutonium spheres.

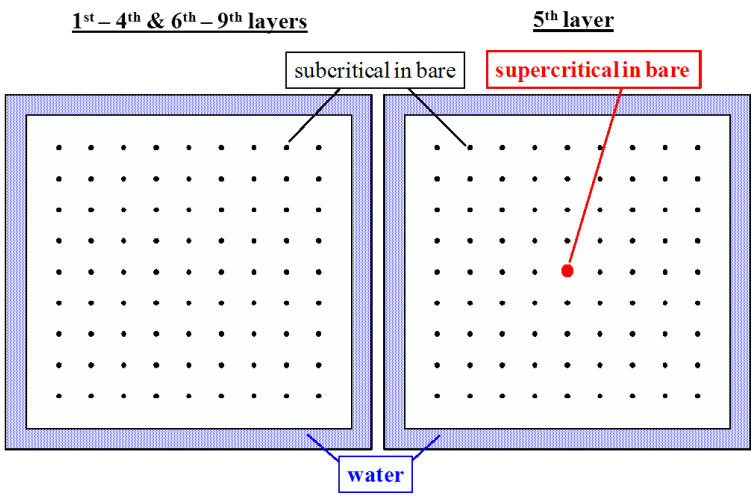


Figure 6.10: Whiteside k-effective of the World Problem [1]

The model was run with 100,000 particles, 2000 cycles and 100 inactive cycles. The dominance ratio of the problem is 0.73 [1]. The results of this run were very promising with quick convergence as well as with high similarity of trends between the *RAPI* method and the entropy values used for convergence indication. The following four figures demonstrate this. Figure 6.11 shows that the *RAPI* distance calculation converges very quickly. As with the previous problem, the convergence cycle was determined with the use of the step-refined on-the-fly diagnostic tool described in Chapter 5. The results of the implementation of this diagnostic derived from the *RAPI* values are shown in Table 6.5. After 7 steps the diagnostic declares the convergence cycle as 183. After 7 steps stopping criterion a was satisfied, so the stepping of the diagnostic was terminated; i.e., $20\times\underbrace{7}_{A}=140>121=0.66\times\underbrace{183}_{j_C}$. The graph of the *RAPI* values over the total number of iterations strangely fluctuates after reaching a convergent state. Perhaps this trend appears because the large amount of void in the geometry, but no explanation for this phenomenon could be found to explicitly be the cause. The results obtained using *RAPI*

for convergence cycle determination was then compared to the entropy based convergence cycle determination to see if there was agreement.

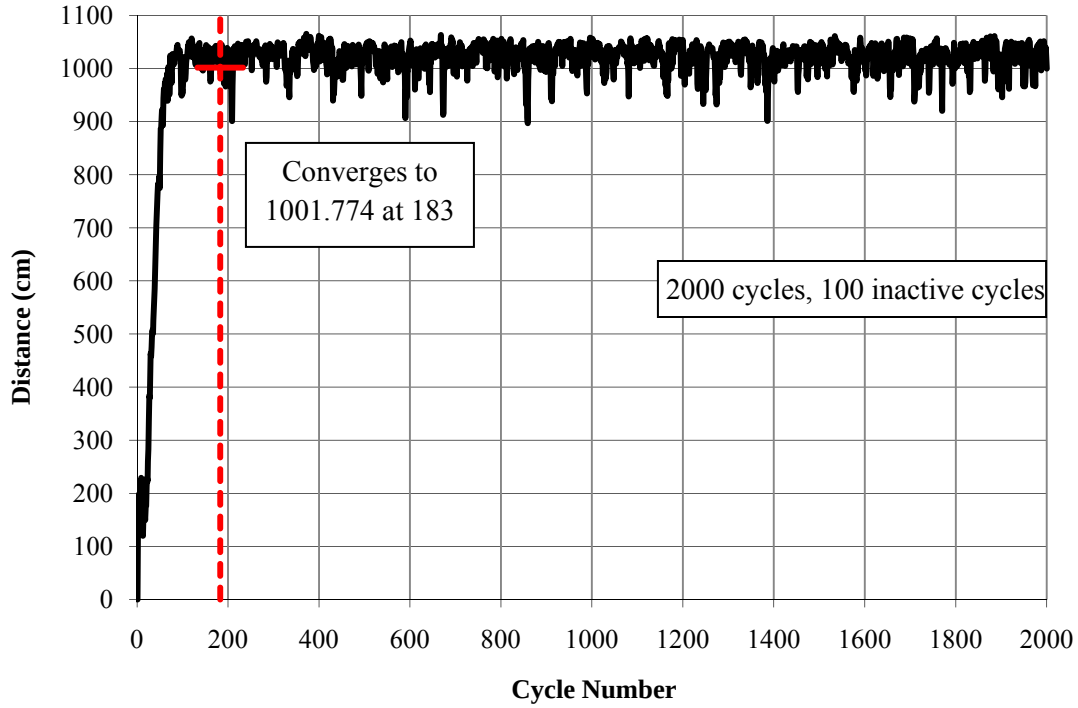


Figure 6.11: *RAPI* Distances for Whiteside's k_{eff} of the World Problem with 100,000 Particles per Cycle

Step (i)	Cycle of first downward crossing	Cycle at which decision was made ($j_S(i)$)	Declared convergence cycle (j_C)
1	98	98	**98**
2	98	118	**98**
3	162	162	**162**
4	183	183	**183**
5	188	203	**183**
6	188	223	**183**
7	206	243	**183**

Table 6.5: Determination of Convergence Cycle Using *RAPI* for K_{eff} of the World Case

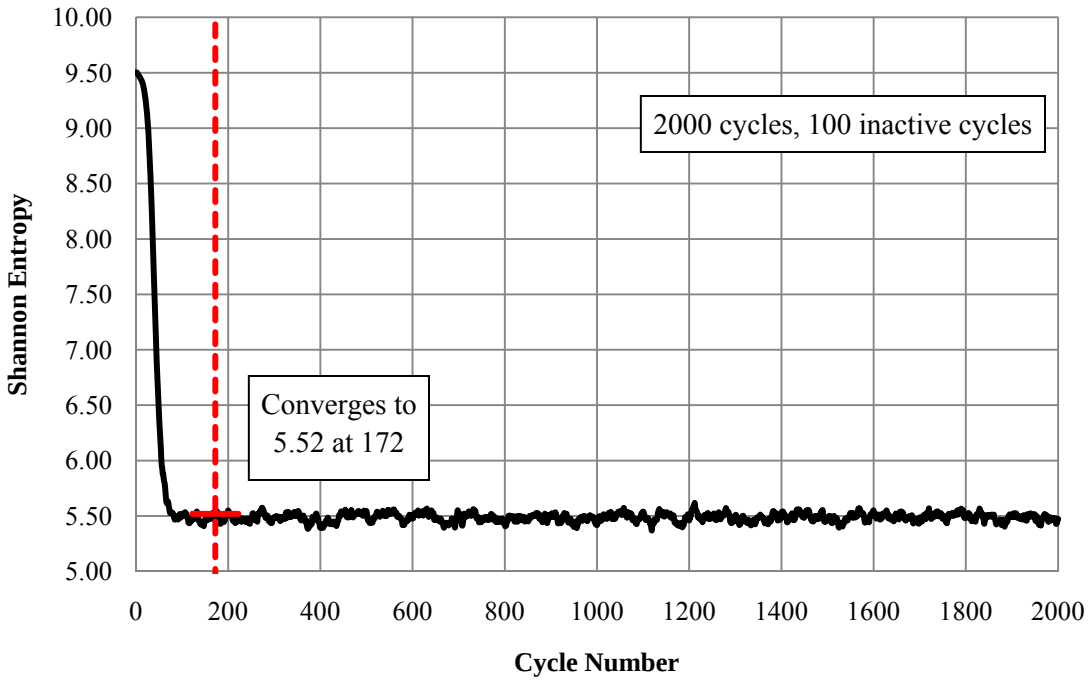


Figure 6.12: Shannon Entropy for Whiteside's k_{eff} of the World problem with 100,000 Particles per Cycle

The Shannon entropy results were the first that the *RAPI* results were compared against. The behavior of the Shannon entropy showed the opposite behavior of the *RAPI* descending as opposed to ascending. But the Shannon entropy reached a fluctuating state in a relatively small number of cycles. The step-refined on-the-fly diagnostic of the source distribution was performed using the Shannon entropy indicator of the source distribution. The results can be found in Table 6.6. The declared convergence cycle was found to be 172 at the sixth step when stopping criteria a was met. Comparing the two declared convergence cycles showed a very small difference. The convergence cycle determined by *RAPI* method was found later in the stepping process but since the declared cycles are so close, the *RAPI* can be as reliable as the Shannon entropy in on-the-fly diagnosis in determining source distribution convergence. The Shannon entropy

determines convergence in fewer steps than *RAPI*, but *RAPI* does not need a user input binning definition.

Step (i)	Cycle of first upward crossing	Cycle at which decision was made ($j_S(i)$)	Declared convergence cycle (j_C)
1	103	103	**103**
2	124	133	**103**
3	172	172	**172**
4	164	192	**172**
5	174	212	**172**
6	198	232	**172**

Table 6.6: Determination of Convergence Cycle Using Shannon Entropy for K_{eff} of the World Case

The *RAPI* results for Whitesides k_{eff} of the world case was then compared to the resulting convergence cycle found using the *PRE* of the system. The graph of the *PRE* for this case can be found in Figure 6.13, and its trend is very similar to that found in Figure 6.11. The convergence cycle determined for the k_{eff} of the world problem using progressive relative entropy was found to be 104 using the step-refined on-the-fly diagnostic shown in Table 6.7. The cycle determined using the *PRE* took 4 steps to achieve the determination of the cycle convergence when criterion a was met, and this declaration is 79 cycles sooner. With the later convergence cycle found by using *RAPI*, physical property estimation may have fewer values with which to average over, but the averaging of these values will be accurately done. The *PRE* may be more efficient in determining the convergence cycle of Whitesides problem, but it still needs the user to define the binning structure of the system. This comparison validates the use of the *RAPI* method for determination of the convergence cycle of the source distribution, because the both were able to determine a convergence cycle of the system.

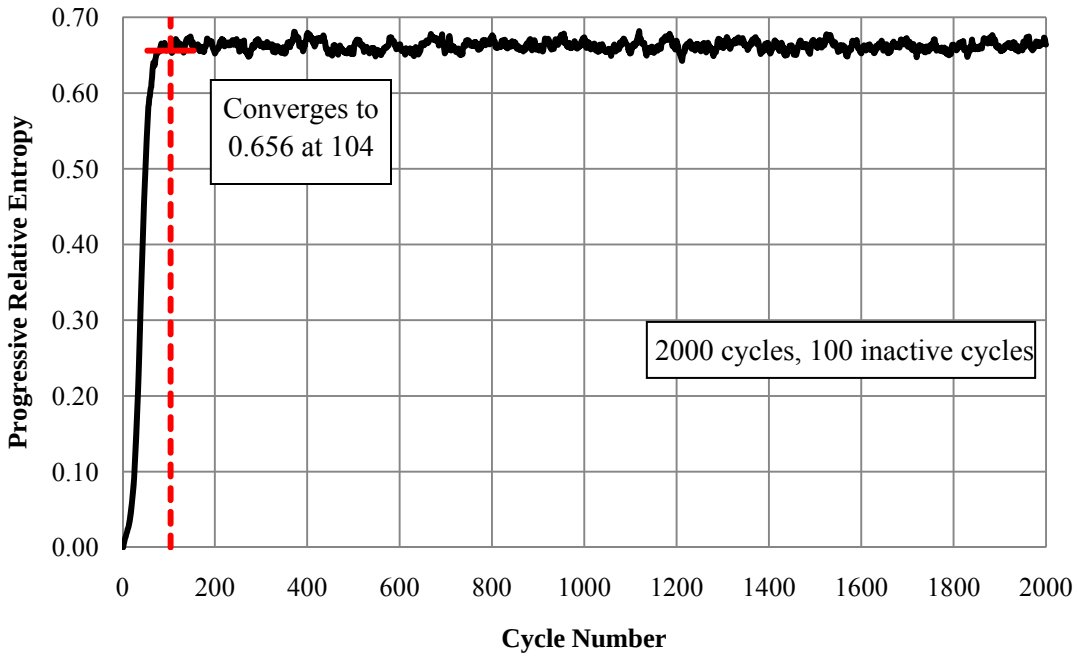


Figure 6.13: Progressive Relative Entropy for Whiteside's k_{eff} of the World problem with 100,000 Particles per Cycle

Step (i)	Cycle of first downward crossing	Cycle at which decision was made ($j_S(i)$)	Declared convergence cycle (j_C)
1	104	104	**104**
2	123	124	**104**
3	144	144	**104**
4	162	164	**104**

Table 6.7: Determination of Convergence Cycle Using *PRE* for K_{eff} of the World Case

The final comparison of the use of the *RAPI* for convergence determination as done against the *PosRE* results of convergence cycle determination. The trend of the *PosRE* found in Figure 6.14 has the same behavior as that of the Shannon entropy except it does not fluctuate as much once it reached stationary state. The convergence cycle was found by determining when the *PosRE* crossed the average of the *PosRE* over the second half of the active cycles. The average of the *PosRE* over the second half of the active cycles was found to be 0.0155, and the *PosRE* crosses at cycle number 78.

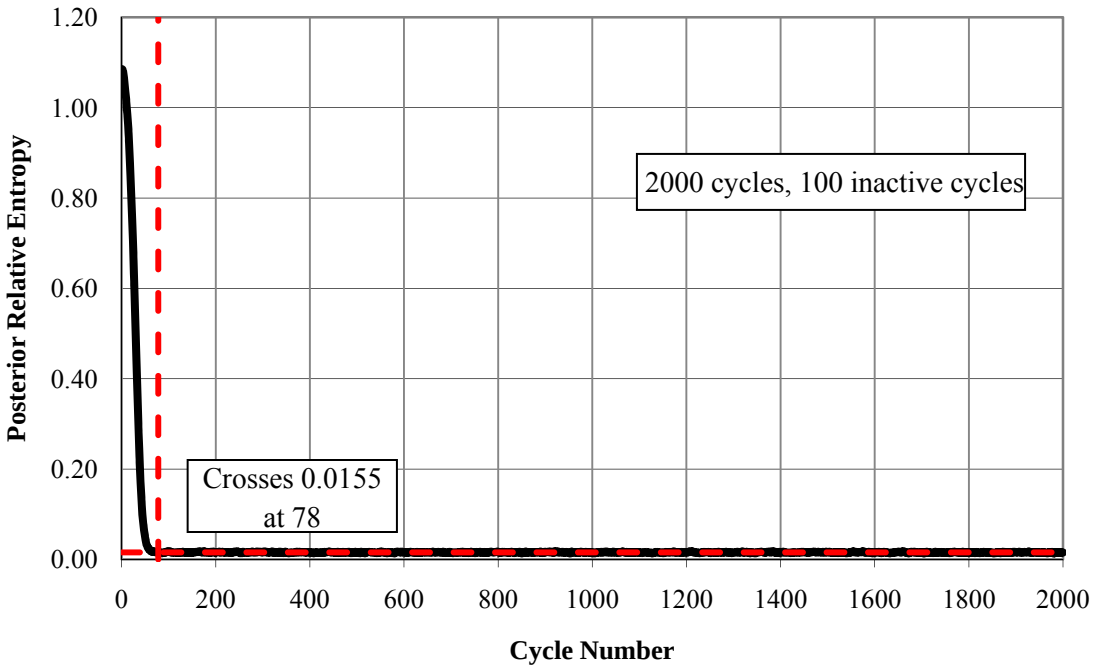


Figure 6.14: Posterior Relative Entropy for Whitesides K_{eff} of the World Problem with 100,000 Particles per Cycle

The convergence cycle using the *PosRE* found the system to converge quicker than did using *RAPI*. The convergence cycle declared through the use of *RAPI*, is a conservative declaration of convergence when compared against the cycle declared through the use of *PosRE*.

The comparison of the convergence determination using *RAPI* of the source distribution's state against the entropy indicators of the source distribution's state show that the *RAPI* method is a good tool for determining when physical property estimation or tallying can begin. The *RAPI* converged fairly quickly performing slower than the entropy indicators but accomplished the task. In using *RAPI* in addition to the entropy indicators, one will be able to effectively determine the convergence cycle of Whitesides k_{eff} of the world problem or problems with a dominance ratio akin to this problem. By comparing the results utilizing *RAPI* to those calculating convergence using the entropies

of the system, using *RAPI* proved to be effective in determining the convergence cycle of the source distribution without the need for a user defined mesh.

6.3 Fuel Storage Vault

The final case run in MCNP was a fuel storage vault of fresh PWR fuel assemblies "with the checkerboard placement of fuel bundle units and water-filled units" as depicted in Figure 6.15 [3]. The checkerboard arrangement is surrounded by concrete and water as shown. The initial source distribution of the problem is restricted to the upper left fuel bundle because the system is over moderated, and thus concrete is a better reflector than water. The bin definition for the problem used for the calculation of the system entropies is one bin per a fuel bundle horizontally and 12 vertically per a fuel bundle for a total number of bins of 432. The dominance ratio was computed to be 0.9943 for the horizontal first and vertical fundamental mode and 0.9934 for the horizontal fundamental and first vertical mode [3]. The convergence cycle determination was done for this simulation with 200,000 particles, 1,500 cycles and 500 inactive cycles. The result for the calculation of the *RAPI* of the source distribution's state through the iterational computation is found in Figure 6.16. Because of the dominance ratio that was computed, the source distribution has slow convergence and this fact can be seen in the slow convergence of the *RAPI*. The results of the step-refined on-the-fly diagnosis employing the use of the *RAPI* are illustrated in Table 6.8. The convergence cycle of the source distribution was determined to be 851 in the 16^{th} step of the diagnosis.

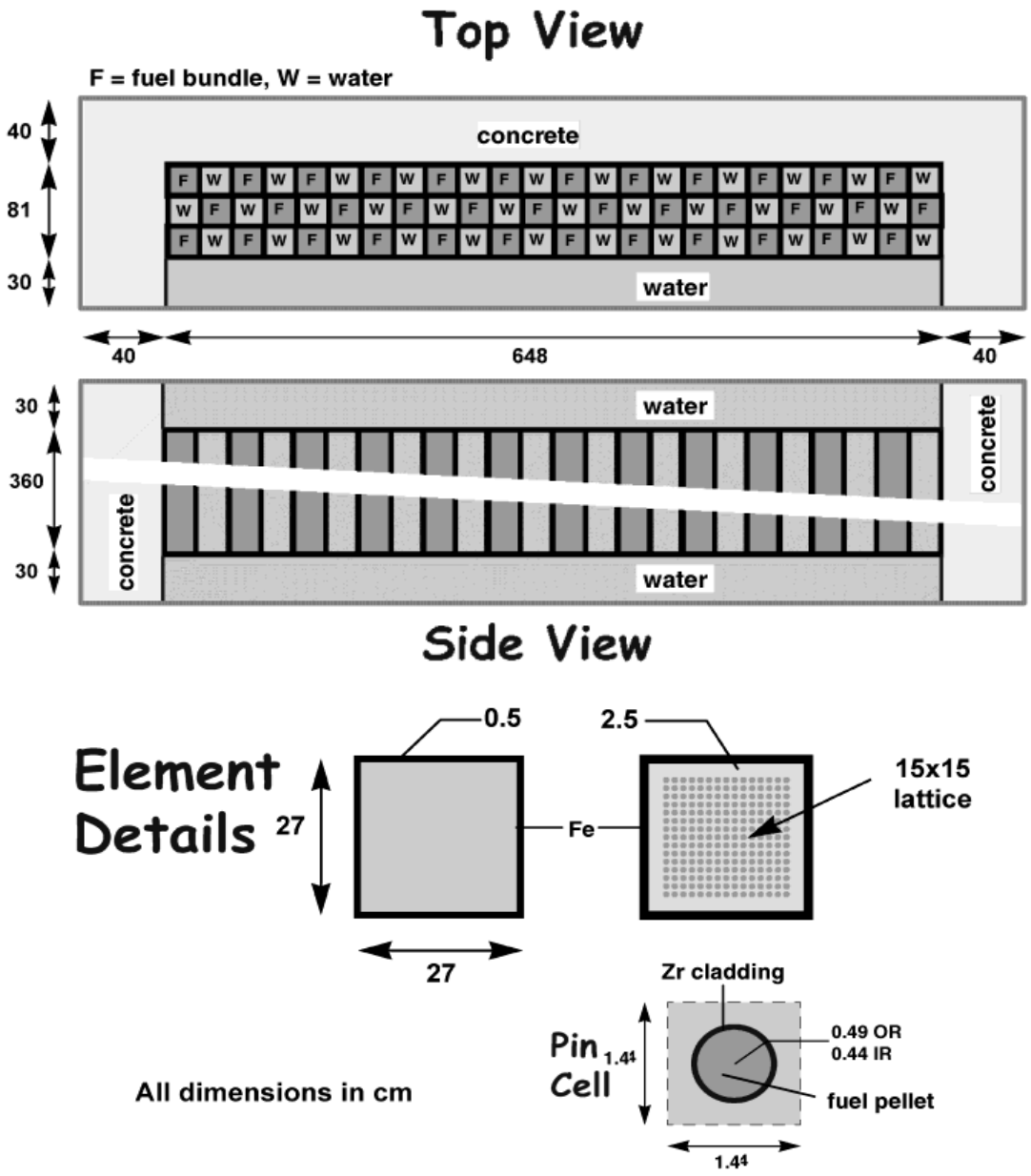


Figure 6.15: PWR Fresh Fuel Vault (FVF) [3]

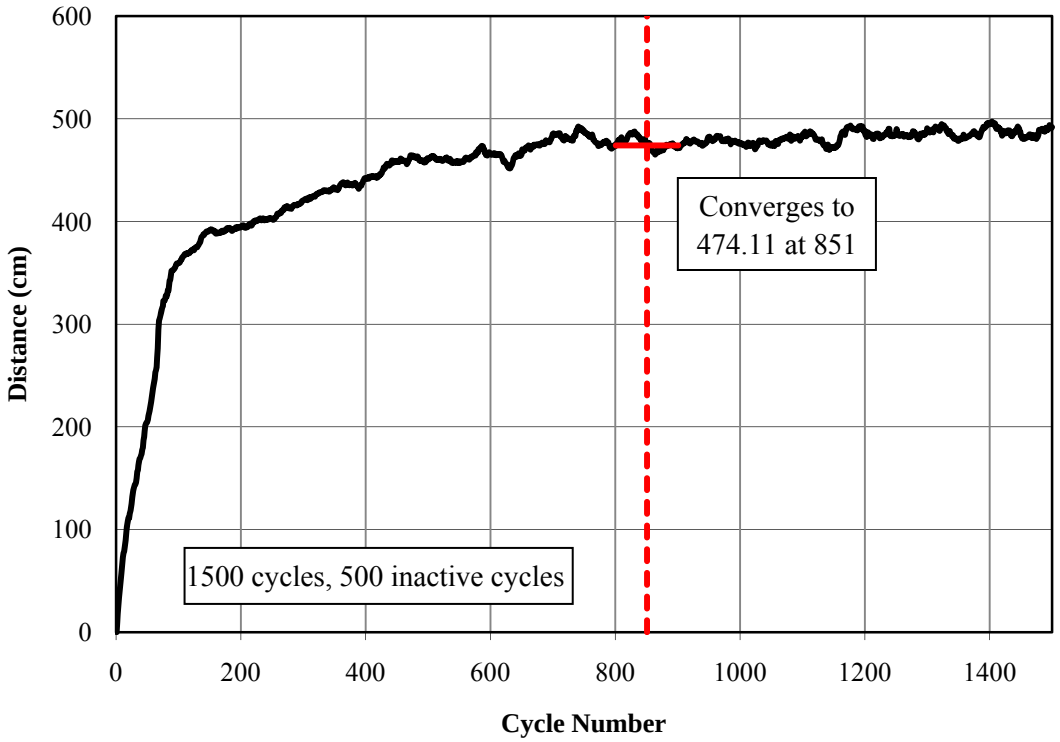


Figure 6.16: *RAPI* distances for Fuel Storage Vault Model with 200,000 Particles per Cycle

Step (i)	Cycle of first downward crossing	Cycle at which decision was made ($j_S(i)$)	Declared convergence cycle (j_C)
1	166	166	**166**
2	485	485	**485**
3	516	516	**516**
4	522	536	**516**
5	541	556	**516**
6	625	625	**625**
7	629	645	**625**
8	629	665	**625**
9	628	685	**625**
10	773	773	**773**
11	792	793	**773**
12	851	851	**851**
13	843	871	**851**
14	857	891	**851**
15	861	911	**851**
16	897	931	**851**

Table 6.8: Determination of Convergence Cycle Using *RAPI* for FVF case

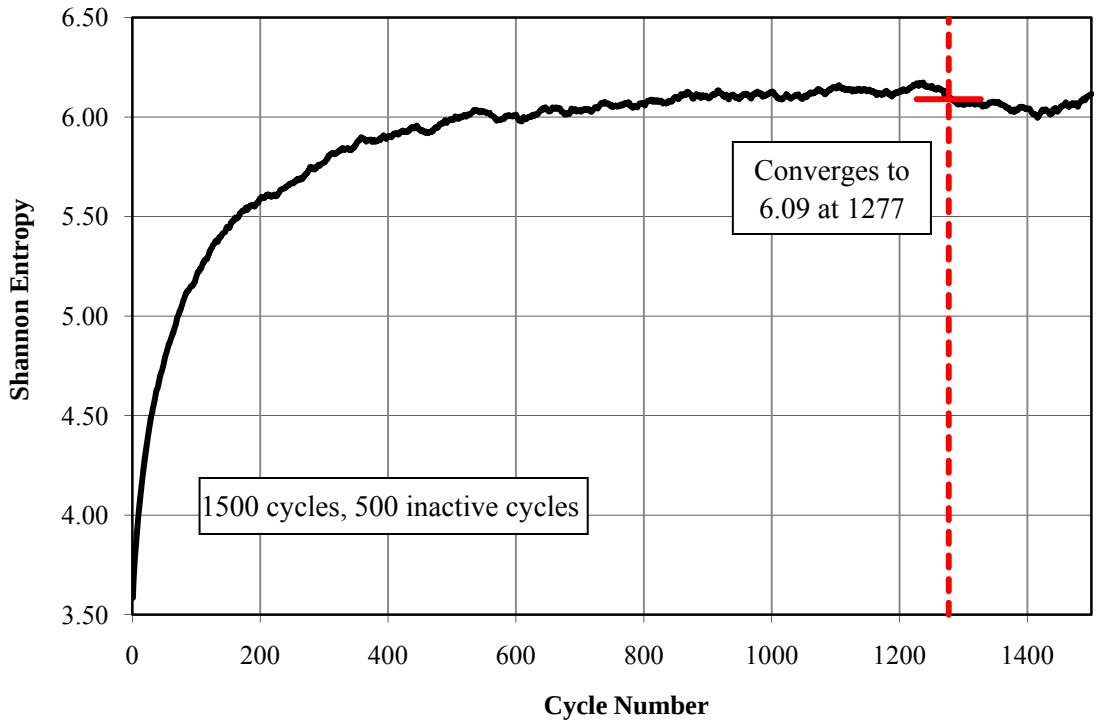


Figure 6.17: Shannon Entropy for Fuel Storage Vault Model with 200,000 Particles per Cycle

The convergence cycle was then determined by the step-refined on-the-fly diagnostic of the source distribution using the Shannon entropy. The convergence cycle of the source distribution was determined to be 1277 in the 15th step (Table 6.9). This value is much greater than that determined by using the *RAPI*. The reason for this result can be seen when examining the trend of the Shannon entropy found in Figure 6.17. The Shannon entropy does not enter into convergence until iteration 1277, dipping in value, seemingly entering a fluctuating state. The *RAPI* would allow the initiation of property tallying sooner than the Shannon entropy for the fresh fuel vault (FVF) problem. Like the homogeneous cube problem the FVF model has a high dominance ratio. The high dominance ratio could be the reason for the slow convergence of the Shannon entropy.

Step (i)	Cycle of first downward crossing	Cycle at which decision was made ($j_S(i)$)	Declared convergence cycle (j_C)
1	375	375	**375**
2	460	460	**460**
3	564	564	**564**
4	568	584	**564**
5	604	604	**604**
6	624	624	**604**
7	674	674	**674**
8	694	694	**674**
9	1028	1028	**1028**
10	1011	1048	**1028**
11	1277	1277	**1277**
12	1279	1297	**1277**
13	1283	1317	**1277**
14	1277	1337	**1277**
15	1277	1357	**1277**

Table 6.9: Determination of Convergence Cycle Using Shannon Entropy for FVF case

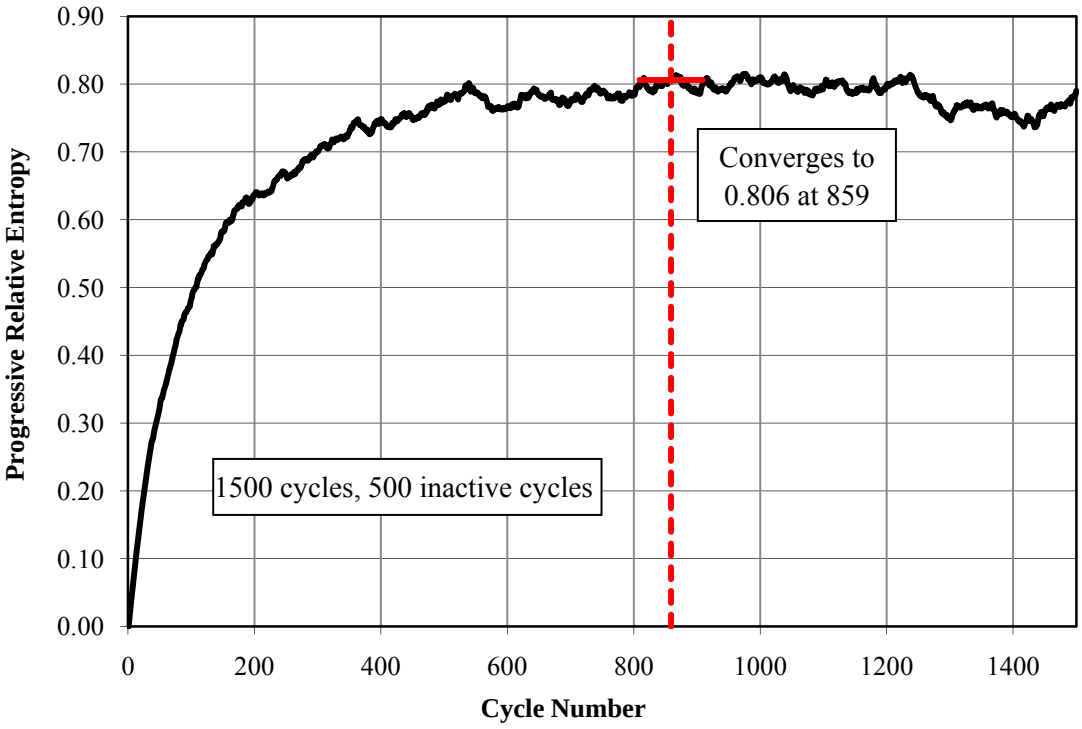


Figure 6.18: Progressive Relative Entropy for Fuel Storage Vault Problem with 200,000 Particles per Cycle

The convergence cycle of the source distribution for the FVF problem was then found using the step-refined diagnostic of the progressive relative entropy (*PRE*). The

PRE exhibits a similar trend to that of the Shannon entropy illustrated in Figure 6.18. The *PRE* indicator has slow convergence just as the Shannon entropy does, possibly due to the high dominance ratio of the simulation. The step-refined diagnostic was able to determine a convergence cycle by the 20th step when stopping criterion b was met. The convergence cycle was declared to be 859, which is close to the cycle determined by using the *RAPI*.

Step (i)	Cycle of first downward crossing	Cycle at which decision was made ($j_S(i)$)	Declared convergence cycle (j_C)
1	264	264	**264**
2	383	383	**383**
3	411	411	**411**
4	575	575	**575**
5	589	595	**575**
6	627	627	**627**
7	657	657	**657**
8	664	677	**657**
9	683	697	**657**
10	707	717	**657**
11	747	747	**747**
12	757	767	**747**
13	764	787	**747**
14	787	807	**747**
15	830	830	**830**
16	859	859	**859**
17	878	879	**859**
18	891	899	**859**
19	899	919	**859**
20	932	939	**859**

Table 6.10: Determination of Convergence Cycle Using *PRE* for FVF case

Finally the convergence cycle was determined from the posterior relative entropy of the PWR fresh fuel vault. The average of the posterior relative entropy over the second half of the active cycles is 0.0104. As previously mentioned, the posterior relative entropy should converge after it crosses the average over the second half of the active cycles. This crossing occurs at cycle 892 which is seen in Figure 6.19. This is

fairly close to the *RAPI* derived cycle determination, which validates the use of *RAPI* for determining when the source distribution has reached a stationary state.

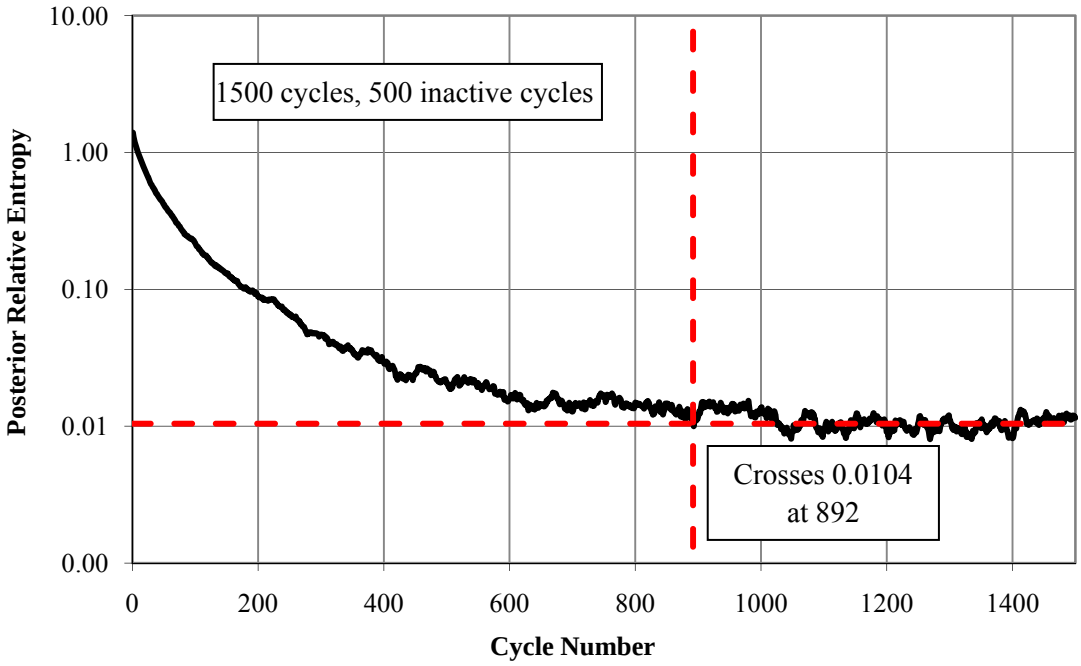


Figure 6.19: Posterior Relative Entropy for Fuel Storage Vault Problem with 200,000 Particles per Cycle

The convergence cycle determined through the use of the *RAPI* is the best option when evaluating the fresh fuel vault for PWR assemblies with the initial source distribution concentrated in the upper left fuel assembly. Because of the high dominance ratio of the problem, the posterior relative entropy of the system declared convergence later than the *RAPI*. The on-the-fly judgment of the progressive relative entropy also took more steps than did the judgment using *RAPI* values. Also due to the high dominance ratio, the Shannon entropy showed slow convergence not declaring a convergence cycle until late into the iterations. The analysis of this problem exhibits the usefulness of a mesh-input-free convergence indicator. Where the entropy indicators, which are defined by binning the source distribution, were greatly affected by the high

dominance ratio, the regionwise average position indicator of the source distribution converged sooner and would allow a greater number of tallies to be used in the physical property estimation.

From the last few models, it can be seen that the number of particles, size of the geometry, heterogeneity, and magnitude of the dominance ratio play an important role in the determination of the convergence cycle. The demonstrated performance of the regionwise average position indicator (*RAPI*) in the analysis and comparison of the last three cases proves that it is a versatile and effective indicator of the source distribution. The *RAPI* can be used to determine at which cycle or iteration the source has reached a stationary state. In this investigation it was found that the *RAPI* performs as well as or better than the bin dependent entropy indicators, because it can be used to determine the cycle of stationarity of the source for a range of models. *RAPI* is a better representation of the source distribution in the geometry as a whole while the entropies are piecewise approximations of the source distribution used to model the problem. The *RAPI* is also better because the need for user defined bins for tally initiation can be eliminated, thus simplifying and increasing the efficiency of any code it is included in. With the use of this tool, the level of experience with particle distribution behavior in Monte Carlo modeling needed to obtain valid results will be reduced. By using the regionwise average position indicator, the accuracy of results and the usefulness of the modeling being performed will be improved.

Chapter 7

Numerical Results for Population Diagnostics

Due to limitations found using EMST's in Monte Carlo, such as the computational time, which increases on the order of the number of particles squared, and data storage constraints, an approximation of the properties of an EMST past these limits for large batches of particles needed to be developed. As discussed in Chapter 4 such an approximation was developed for the average edge length of an EMST. In this approximation or power law relationship, the average edge length (*AEL*) was proportional to the inverse of the number of particles raised to the power of 1/*b* or $AEL(N) \propto N^{-1/b}$. From the theory used to develop this relationship, *b* is ideally equal to 3 but affected by the heterogeneity of the system. From this, *b* should be close to three for a homogeneous system where there is only fissile material. To test the validity of this statement the average edge length of an EMST for a varying number of particles was calculated for a homogenous cubic medium with and without reflecting boundary conditions. The cubic medium had side length of 20 cm with energy independent and isotropic scattering

macroscopic cross sections of $\Sigma_T = 1.0cm^{-1}$, $\Sigma_S = 0.75cm^{-1}$, $\Sigma_A = 0.25cm^{-1}$ and $\upsilon\Sigma_f = 0.275cm^{-1}$. The determination of *b* was done by fitting a trendline to the data obtained over the changing number of particles and is displayed in Figure 7.1.

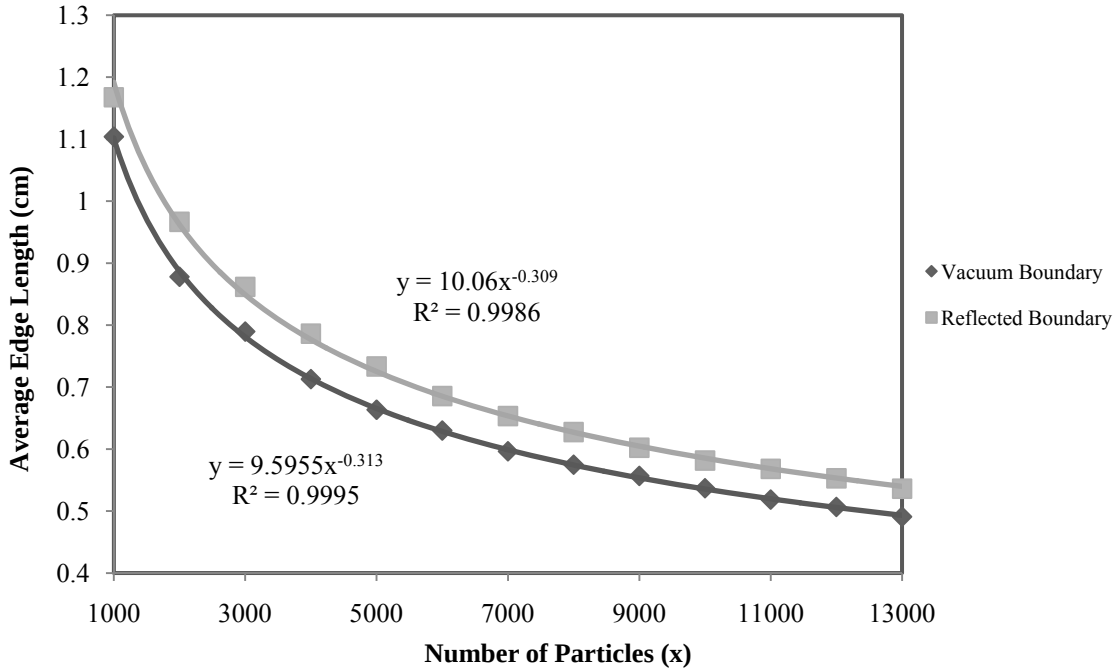


Figure 7.1: Determination of *b* for 20x20x20 Homogeneous Cube Case

From the equation for the trend line for the reflected boundary condition, *b* is equal to 3.24. For the vacuum boundary condition, *b* is found to be 3.19, both of which are close to three. So the assumption of *b* being close to 3 in homogeneous systems is found to be reasonable.

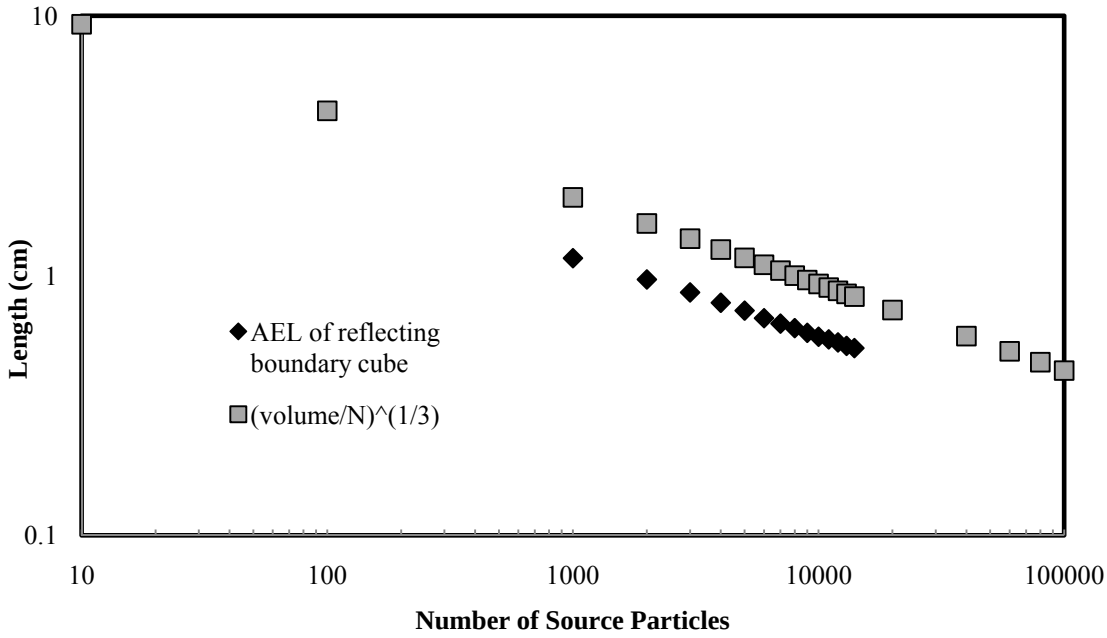


Figure 7.2: Comparison of AEL and Cubic Root of Volume per Particle for 20x20x20 Homogeneous Cube

Figure 7.2 compares the average edge length with the cubic root of volume per source particle. This comparison is done to illustrate the differences in the two techniques used to describe the effective neighborhood distance of a particle. It is observed that the *AEL* is much smaller than the later.

Prior to developing the diagnostic of the tally mesh resolution utilizing the average edge length of an EMST, the reliability of the EMST approach needs to be verified, i.e., which technique better captures an effective neighborhood distance or population sufficiency in nuclear criticality calculations. To do this, an EMST was constructed from the particle positions in the k_{eff} of the world problem of Whitesides [15]. Whitesides problem consists of bare subcritical Plutonium-239 spheres with a radius of 3.7819 cm placed at intervals of 60 cm in a 9x9x9 array except the central sphere. The central sphere is supercritical in a bare state and has a radius of 4.968 cm. The space between the spheres is void and the array is surrounded on all six sides by a

water reflector of 30 cm thickness, whose interior surface is 60 cm away from the center of the sphere on the edge of the array. A schematic of the problem is found in Figure 7.3.

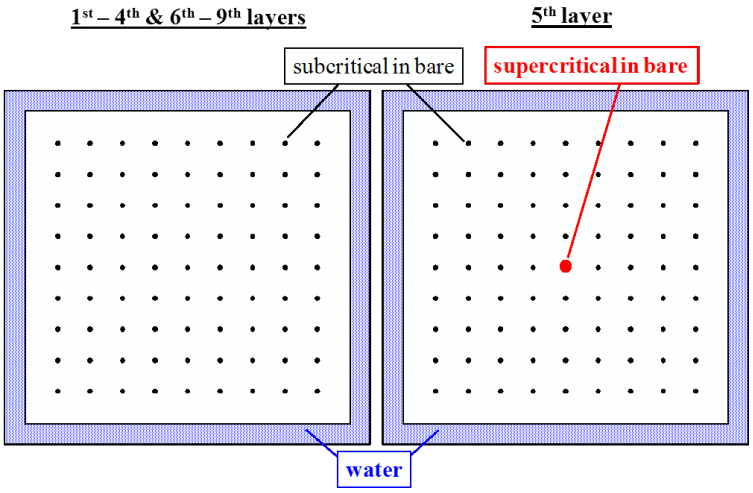


Figure 7.3: Whiteside k-effective of the World Problem [1]

In this problem, if the source distribution does not converge to the fundamental source mode, error can be introduced into computed results. Error or bias in results is also increased if the number of particles that are used to simulate the criticality problem is too small. Figure 7.4 shows the k_{eff} of the problem for various numbers of source particles per cycle computed by MCNP [13].

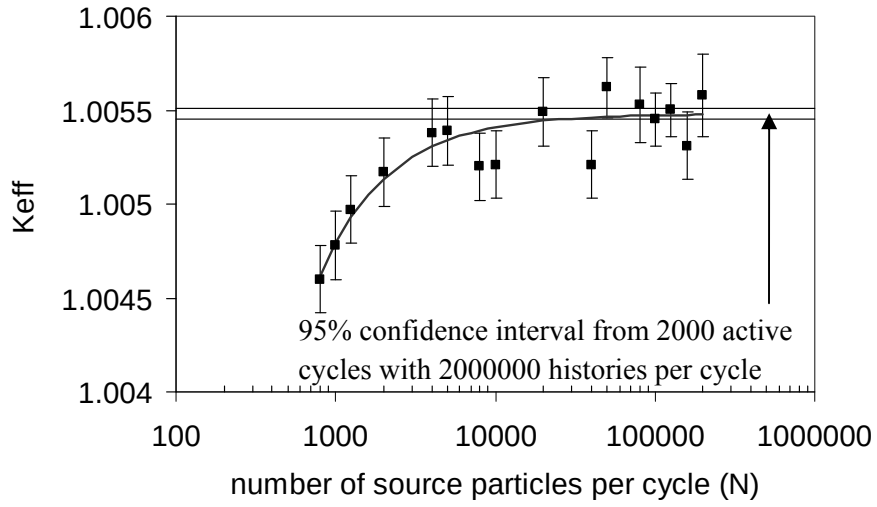


Figure 7.4: Confidence Interval (95%) of the k-effective of the World Problem [9].

The variance of the k_{eff} values are calculated by the batch method of a batch size of 50 cycles [1]. From the trendline in this figure "it is seen that if the number of source particles per cycle is smaller than 10,000, the k-effective estimate is certainly under the influence of bias and the third fractional digit may be affected; if the number of source particles per cycle is larger than 40,000, it is virtually free of bias" [17]. The k_{eff} estimate is under strong influence of bias before 10,000 particles, and the k_{eff} bias characteristic moves to gray or weaker influence between 10,000 and 40,000 particles.

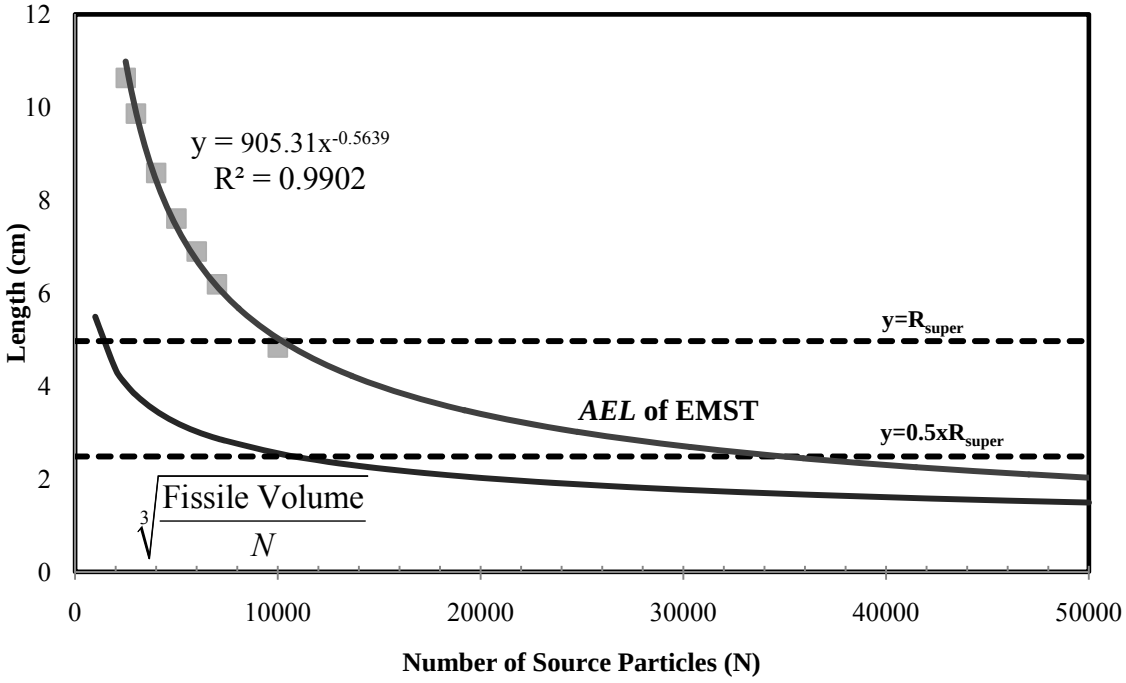


Figure 7.5: Average Edge Length of EMST Applied to Source Particles in k_{eff} of the World Problem

Figure 7.5 shows the cubic root of the fissile volume per source particle and the average edge length of an EMST for Whitesides problem for a varying number of source particles per cycle. Figure 7.5 also displays the radius of the supercritical sphere, $R_{super,}$ and ½ of the radius of the supercritical sphere, $0.5xR_{super,}$ as examples of characteristic geometric lengths for sufficiently sampled source particles. The *AEL* is extrapolated past

the limits of the number of particles allowed in the EMST calculation by the trendline equation displayed in the figure. The shift of the value of *AEL* from greater than R_{super} to less than R_{super} and from greater than $0.5xR_{super}$ and less than $0.5xR_{super}$ happens at 10,000 and 35,000 particles respectively. These transitions are consistent with transitions of the bias of k_{eff} found in Figure 7.4. An EMST created from fission site particles contain edges that connect spheres that are 60 cm apart at center; these edge lengths are an order of magnitude larger than R_{super}. The number of edges that connect the spheres is at least 9x9x9-1=728 if all the spheres contain at least one source particle. This is so because to create an EMST, all the source particles must be connected by an edge and in an EMST with *N* vertices there are *N*-1 edges. Because of this, the transition of *AEL* from greater than R_{super} to less than R_{super} represents a transition to the source particle neighboring distance in the scale of R_{super}. The transition of the AEL to values between R_{super} and $0.5xR_{super}$ happens around the same number of particles where bias characteristic of the k_{eff} changes from strong influence to gray influence. When examining the behavior of the cubic root of the fissile volume per a number of particles, the previously mentioned transition to values between the two R_{super} limits happens in the range of 1,000 to 10,000 particles. In comparison, the k_{eff} is under strong bias influence for this range of particle number. Beyond 10,000 particles the cubic root of the fission volume per source particle is less than the lower limit of $0.5xR_{super}$. The effective neighbor distance determined as the cubic root of fissile volume per source particle is smaller than the lower limit or characteristic geometric length where the k_{eff} bias characteristic on the third fractional digit cannot yet be ignored. As the values of cubic root of the fissile volume per a particle progresses towards 40,000 particles, above which the k_{eff} estimation is nearly free

of bias, there is little change in value. Due to the cubic root of the fissile volume per particle taking on values less than $R_{super,}$ where the bias strongly influences results, it is determined that it does not capture the effective neighborhood distance very well. Therefore, because of the transition of the *AEL* of the system consistent with the bias characteristic transition of the k_{eff}, the *AEL* is superior for population diagnostics to the cubic root of the fissile volume per source particle, in the presence of void.

From the previous discussion the *AEL* can be considered as a characteristic neighbor distance and a useful indictor of the adequacy of particle population. The *AEL* of an EMST can be then used to define the effective volume per source particle or the volume of one source particle effect for a set of N particles, represented by $EV(N)$, in equation (7.1).

$$EV(N) = (AEL(N))^3 = \left(\frac{L_{EMST}(N)}{N-1}\right)^3 \tag{7.1}$$

The minimum tally cell volume (MTCV) can then be suggested as

$$MTCV = 30 \times EV(N) \quad (\text{weak requirement}) \tag{7.2}$$

$$MTCV = 100 \times EV(N) \quad (\text{strong requirement}) \tag{7.3}$$

The MTCV requirements in equations (7.2) and (7.3) "refer to two conditions in the sense that ten source particles effects are to be ensured at tally cell locations producing one-third and 10% of average power density, respectively" [17]. Using these definitions will verify whether the cell volume of a given tally mesh is larger than the weak requirement and smaller than the strong requirement or larger than the strong requirement. If the tally cell volume is much larger than the strong requirement, then the particle population will

be regarded as sufficiently large. If the tally cell volume falls within the two limits, then the particle population will be considered acceptable in size. This diagnosis of the particle population does not require the input of mesh by the user.

The application of the meshless population diagnostic is compared against the existing population diagnostic with a diagnostic mesh identified as PD-MESH [9]. This diagnostic is described below. The diagnostic has B bins and $S^B(j)$ and $T^B(j)$ are distributions over B bins normalized to unity; $\sum_{j=1}^{B} S^B(j) = \sum_{j=1}^{B} T^B(j) = 1$, $S^B(j) \geq 0$ and $T^B(j) \geq 0$. The Shannon entropy of S^B is represented by equation (7.4).

$$H\left(S^B\right) \equiv -\sum_{j=1}^{B} S^B(j) \log_2\left(S^B(j)\right) \tag{7.4}$$

The relative entropy and chi-squared distance of S^B with respect to T^B are found in equations (7.5) and (7.6), respectively.

$$D\left(S^B \,||\, T^B\right) \equiv \sum_{j=1}^{B} S^B(j) \log_2\left(\frac{S^B(j)}{T^B(j)}\right) \tag{7.5}$$

$$\chi\left(S^B \,||\, T^B\right) \equiv \sum_{j=1}^{B} \frac{\left[S^B(j) - T^B(j)\right]^2}{S^B(j)} \tag{7.6}$$

These values satisfy equation (7.7).

$$\left| D\left(S^B \| T^B\right) - \frac{1}{2\log_e 2}\chi\left(S^B \| T^B\right) \right|$$
$$\leq \frac{1}{2\log_e 2}\left| \frac{1}{3}\sum_{j=1}^{B} \frac{\left[S^B(j) - T^B(j)\right]^3}{\left[S^B(j)\right]^2} - \frac{1}{4}\sum_{j=1}^{B} \frac{\left[S^B(j) - T^B(j)\right]^4}{\left[S^B(j)\right]^3} + \cdots \right| \tag{7.7}$$
$$= \text{ second order quantity of } \max_{1\leq j\leq B}\left| \frac{S^B(j) - T^B(j)}{S^B(j)} \right|^2$$

Let $S(\vec{r})$ be the fission source distribution at the continuous space coordinate $\vec{r}$ and $S(p,j)$ be defined as the Monte Carlo estimate of $\int_{bin\ j} S(\vec{r})dV$ at cycle p. The normalized source ratio is then defined in equations (7.8) and (7.9).

$$\mathrm{SR}(p) \equiv \frac{\left(\frac{S(p+1,1)}{S(p,1)}, \frac{S(p+1,2)}{S(p,2)}, \ldots, \frac{S(p+1,B)}{S(p,B)} \right)}{\sum_{j=1}^{B} \frac{S(p+1,j)}{S(p,j)}} \tag{7.8}$$

$$R^B \equiv \frac{1}{B}\overbrace{(1,1,\ldots,1)}^{B \text{ components}} \tag{7.9}$$

If the expectation value is denoted by $E[\bullet]$ we can find that the k_{eff} or effective neutron multiplication is in terms of the expectation of the Monte Carlo estimate of the integral of the fission distribution over bin j for cycle p and p+1, and this is found in equation (7.10).

$$\frac{E\left[S(p+1,j)\right]}{E\left[S(p,j)\right]} = 1 \tag{7.10}$$

$S(p+1,j)/S(p,j)$ will be constant over bins j=1,…,B if the particle population is satisfactorily large. From these descriptions and relations, Ueki proposed a more

stringent translation of the population diagnostic criteria in equation (7.7) as can be seen in equation (7.11).

$$D\chi \equiv \left| D\left(SR(p) \,||\, R^B\right) - \frac{1}{2\log_e 2}\chi\left(SR(p) \,||\, R^B\right) \right| < f \times \min\left[1, H\left(SR(p)\right)\right] \quad (7.11)$$

The factor f was recommended as to be 0.025. The choice of 0.025 is shown to be consistent with the k_{eff} eigenvalue calculation of the Whitesides' k_{eff}-of-the-world problem [9]. From this equation, the criterion for the Shannon entropy population diagnostic is arrived at, and it is this criterion that the effective volume of a source particle diagnostic will be compared against. To do this one must determine at what value of *N* does the left side of equation (7.11) become smaller than the right hand side of equation (7.11) for the diagnostic mesh set equal to the tally mesh in a reactor power distribution calculation. Next one examines at what value of *N* does 30x*EV(N)* (or 100x*EV(N)*) become smaller than the cell volume of the tally mesh. Then the two values of *N* obtained from this transition of the PD-MESH [9] value and the multiples of the effective volume of a source particle need to be compared to see if the two values of *N* are close to each other or not. "Except for the problems with a single extremely sharp peak of source distribution and/or the diagnostic checking with a small number of bins, Shannon entropy is larger than unity" [17]. As a result, the right side of equation (7.11) will reduce to *f* for three-dimensional full core reactor core simulations. Note if some of the bins in the Shannon entropy have zero particle presence, these bins will be neglected for calculational purposes.

The effective volume of a source particle diagnostic and the PD-MESH diagnostic were tested against the three-dimensional full core modeling of an 1100 MW_e pressurized water reactor depicted in Figure 7.6 [18]. The tally mesh declaration for power distribution determination was defined to be 4 cells per fuel bundle horizontally and 24 cells vertically. With this the tally cell volume (TCV) was $21.6^2 \times 366/(4\times 24) = 1779 cm^3$. The TCV will then be compared against $30\times EV(N)$ and $100\times EV(N)$. The diagnostic mesh for equation (7.11) produced B bins, where $B = 4\times 24\times 193 = 18,528$. As described in the previous paragraph the Shannon entropy was observed to be much greater than unity, so the criterion in equation (7.11) becomes

$$D\chi < f. \tag{7.12}$$

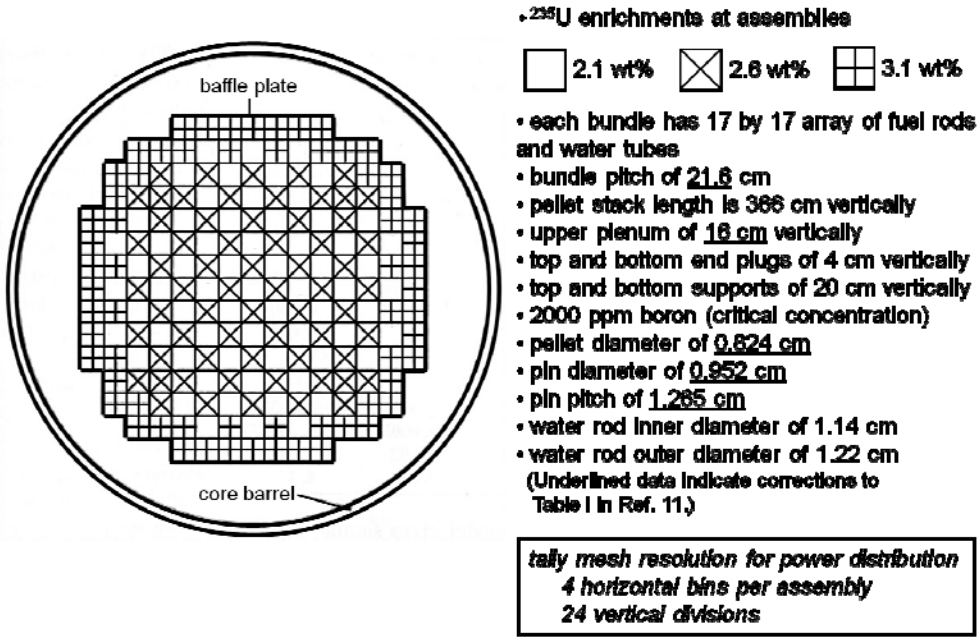


Figure 7.6: 1100 MW_e PWR Full Core Model at Beginning of Reactor Operation [18]

Because equation (7.11) reduces to equation (7.12), it needs to be examined when the left side of equation (7.12) becomes smaller than f. From earlier in this section f was set

equal to 0.025. The *AEL* trends derived from the EMST's of first collision sites and source particles is shown in Figure 7.7.

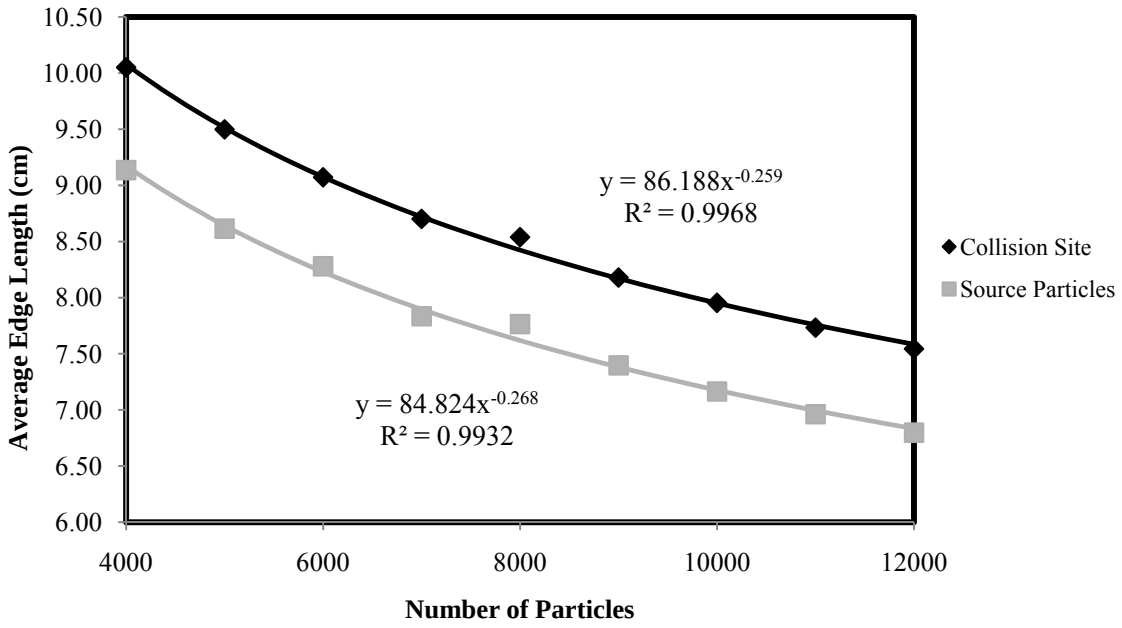


Figure 7.7: EMST AEL from Collision Sites and Fission Sites for PWR Initial Full Core Model

In addition to examining the *AEL* behavior for source particle positions, the first collision site *AEL* was examined. This was examined because the first collision site and source particle sites are on the opposite ends of a particle's first flight, and as the PWR model is reflected by water and the core is large in dimension, the escape of first flight particles is negligible. The performance of the power law fit is found to be exceptional, for this case. The fact that *b* is much closer in value to 3 for this simulation than in the k_{eff} of the world problem of Whitesides can be attributed to the adjacency of the fuel pins in the model.

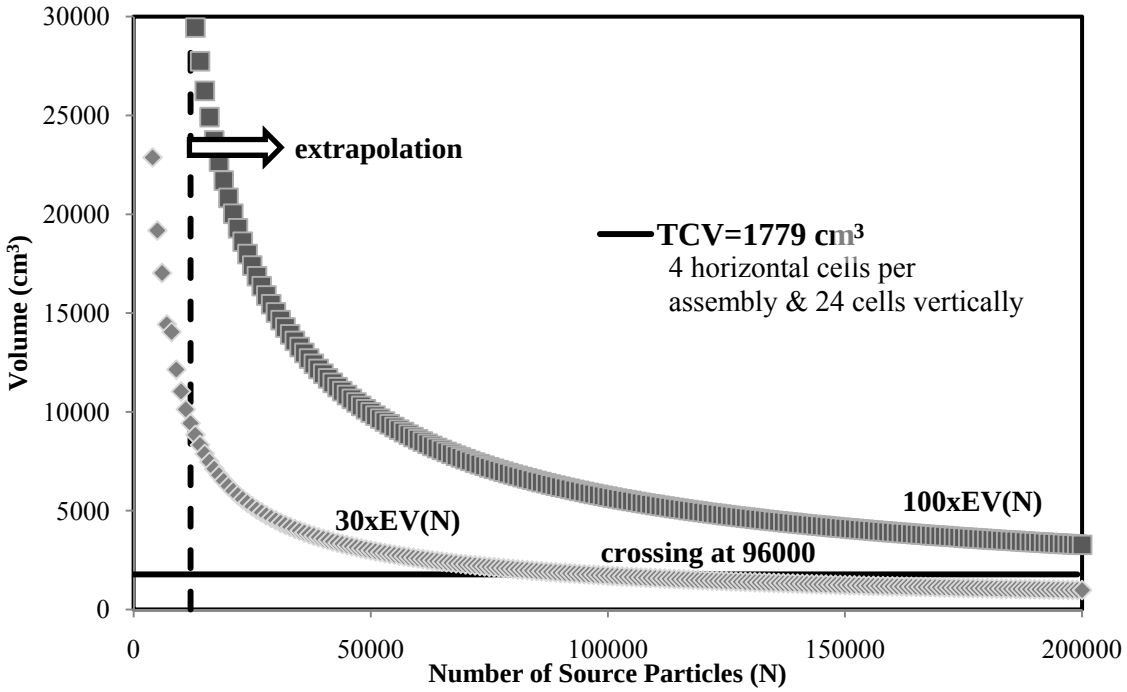


Figure 7.8: Effective Volume per Particle (*EV*) for Source Particles and Tally Cell Volume (*TCV*) for PWR Initial Full Core Model

In Figure 7.8 and Figure 7.9, *TCV* is compared against $30\times EV(N)$ and $100\times EV(N)$ for up to 200,000 source particle positions and 200,000 first collision sites. The extrapolation of the *AEL* past 12,000 in both figures is obtained from the fitting formulae in Figure 7.7. In Figure 7.8, $30\times EV(N)$ for source particle position becomes smaller than *TCV* at 96,000 particles. In Figure 7.10, $D\chi$ is has an average value of 0.0246, becoming smaller than *f*=0.025 for 100,000 histories per cycle (*N*=100,000). The observation obtained from the effective volume per source particle diagnostic for $30\times EV(N)$ crossing at 96,000 source particles is consistent with the downward crossing of $D\chi$, which verifies the ability of the ability of the effective volume per source particle diagnostic in determining an acceptably large particle population for power distribution determinations. It is also observed that $100\times EV(N)$ does not cross *TCV* before 200,000 particle per history

(*N*=200,000). Meaning that for the defined TCV, 200,000 particles may not be a strongly guaranteed large population for power distribution determination.

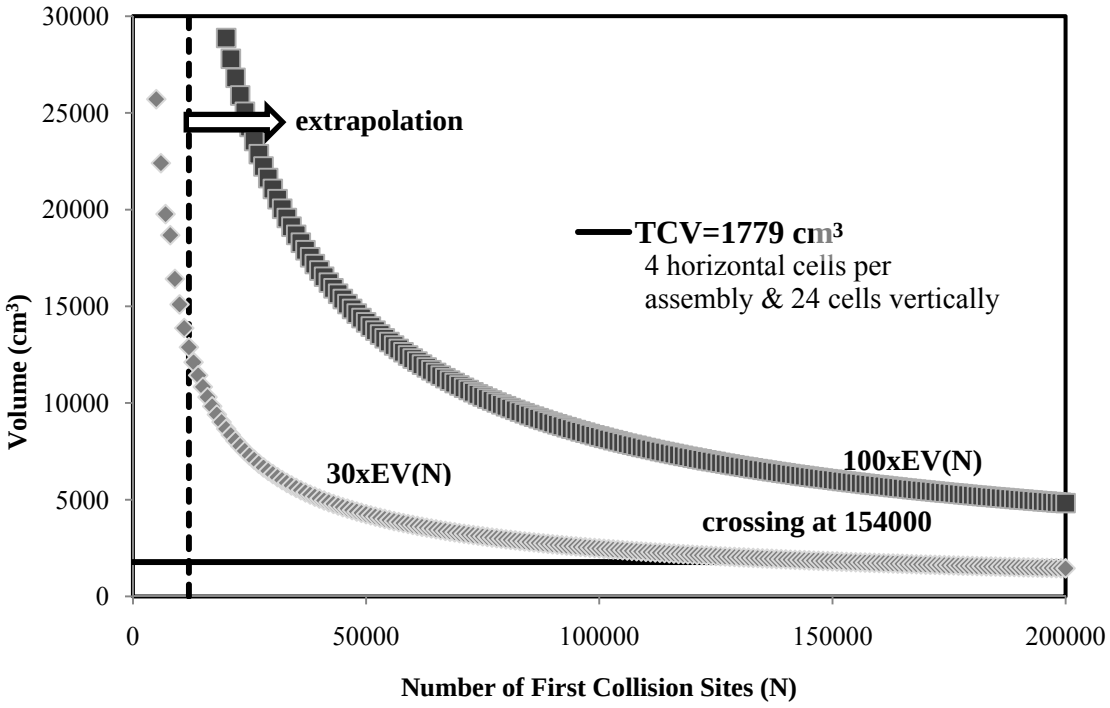


Figure 7.9: Effective Volume per Particle (*EV*) of First Collision Sites and Tally Cell Volume (*TCV*) for PWR Initial Full Core Model

In Figure 7.9, it is observed that the downward crossing of *TCV* by $30\times EV(N)$ for first collision sites occurs at 154,000 first collision sites (*N*=154,000) while $100\times EV(N)$ does not cross *TCV* by 200,000 first collision sites. Because the strong requirement of the crossing of $100\times EV(N)$ downward of *TCV* is not met for either source particle or first collision site and that the first collision site $30\times EV(N)$ does not cross till *N*=154,000, leads one to question the value of f being equal to 0.025. By examining Figure 7.10,

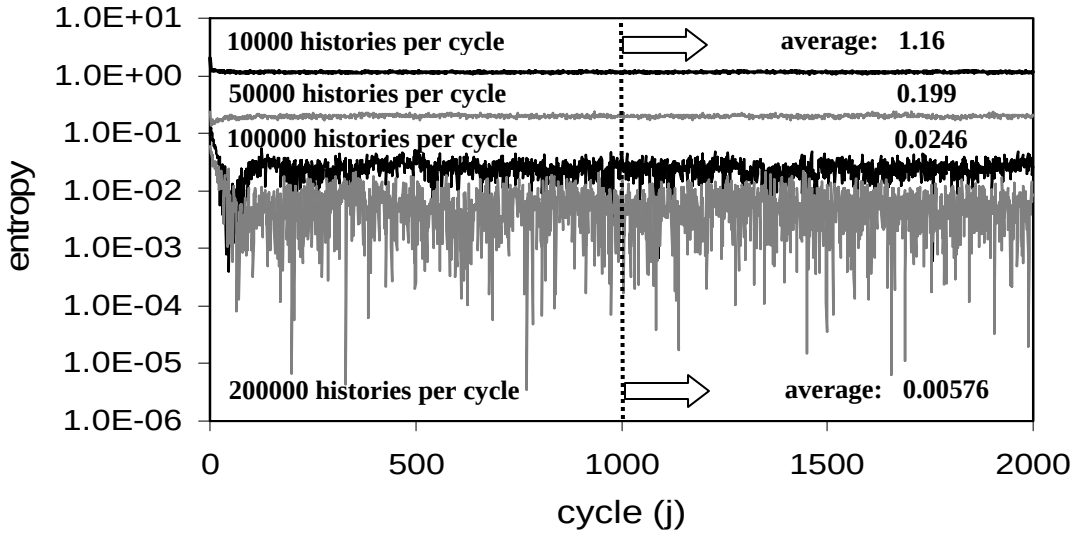


Figure 7.10: Shannon Entropy Diagnostic Measure for PWR Full Initial Core Model [17]

$D\chi$ has an average value over 1001-2000 cycles of 0.00576 for 200,000 particles per history and is found to take values smaller than 0.001 a great many of times. This observation lends credibility to the notion that *f* should have a value lower than 0.025 to be an acceptable criterion for power distribution computation. For example, setting *f* equal to 0.01 would be an acceptable criterion for determining if a particle population size is sufficiently large for power distribution determinations. Finally the comparison of $30\times EV(N)$ and $30\times(\text{fissile volume})/N$ for the PWR Initial Full Core Model is found in Figure 7.11. The fissile volume of this model is the total volume of fuel pellets. It is seen that up until about 3,000,000 particles, $30\times EV(N)$ is smaller in value than $30\times(\text{fissile volume})/N$ and becomes less than the *TCV* for a smaller number of source particles. From this, $30\times EV(N)$ is a more conservative population diagnostic, also

making it superior to $30\times(\text{fissile volume})/N$ for power distribution computations.

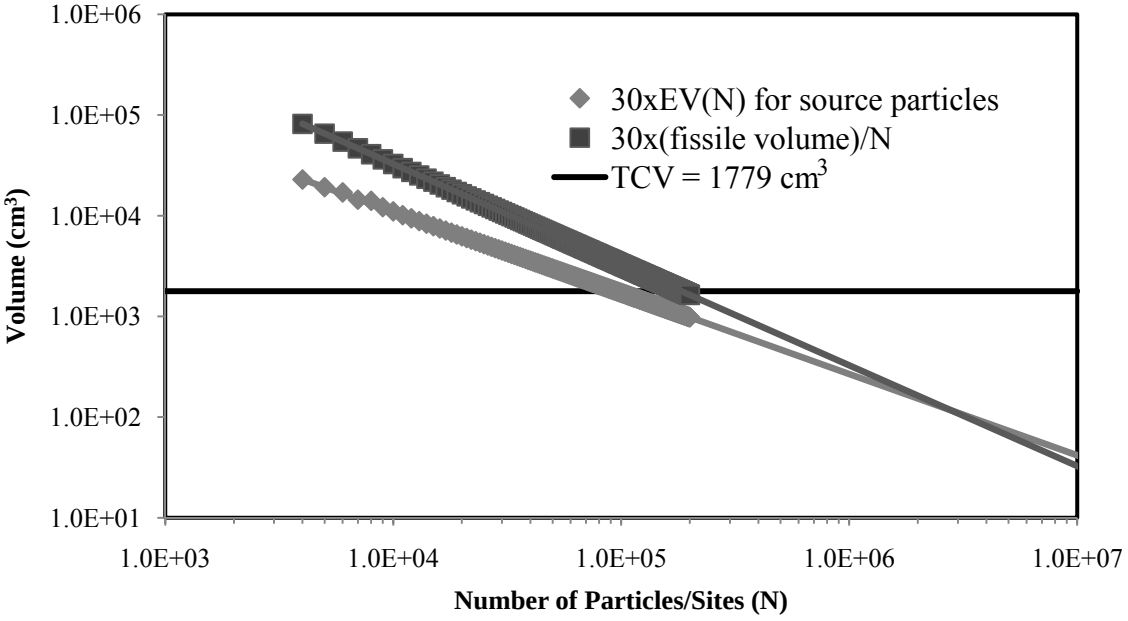


Figure 7.11: Volume Comparison for PWR Initial Full Core Model

The comparison of the effective volume per source particle population diagnostic or meshless diagnostic of the particle population to the mesh based diagnostic, PD-MESH, proposed by Ueki, shows that the meshless diagnostic can determine if the defined particle population is large enough for power distribution computations to be done accurately. By comparing $30\times EV(N)$ against $30\times(\text{fissile volume})/N$, it is seen that the effective volume per source particle is a more conservative gauge of neighborhood distance, also making it a superior diagnostic of the particle population size. The comparisons have shown that the meshless diagnostic can be used to determine if the population is of large enough size for a given tally cell volume, but improvements can be made. For example it was seen that *b* departs from its theoretical value of three depending on the heterogeneity of the system and the number of particles in a simulation. It is this departure that needs to be better understood, and this can be accomplished

through rigorous modeling and tracking. The theory behind the power law could also be refined by trying to understand what assumptions were over looked in its development. When the theory is refined the extrapolation of the *AEL* can be better estimated. These improvements need to be made so that EMST based population diagnostics will be more reliable at determining sufficient particle population size. These improvements will be discussed in more depth in Chapter 8 as part of future work for this area of investigation.

Chapter 8

Future Work

The methods discussed in this thesis for source distribution and particle population diagnostics can be further investigated, developed and improved. The *RAPI* was used to determine when the source distribution had reached a stationary state and when the tallying of the physical properties of the system could begin or when the declaration of active cycles occurs. The techniques used to develop the *RAPI* source distribution diagnostic or automatic mesh creation was not used in physical property of the system estimation. To obtain information about the physical properties of a nuclear system, tallies are averaged over the active cycles in the process. Currently when tallies of the system are done, the user must input the mesh coordinates or bin boundary specifications, and it is from the information in the bins that the tallies of the system are averaged. Bins can be created by the use of automatic meshing. By taking the average particle position of a system the geometry can be divided into bins. Then by taking the average position within these bins more bins can be created; this process can be continued until an effective size and number of bins, for tallying purposes, is achieved. The use of automatic

meshing in Monte Carlo methods needs to be investigated to determine if the automatic meshing of the geometry from average particle position can be used effectively for tallying purposes. If the need for user input mesh can be eliminated, Monte Carlo calculations can be simplified.

Just as the number of bins that can be created from meshing can be increased, the *RAPI* method can be expanded to include the next level of bin average particle position. Up to now, in 3-dimensional problems, *RAPI* is the sum of the distance of nine average coordinates in each cycle or iteration to the first iteration's nine average coordinates. As described in chapter 3 for three-dimensional simulations, the number of bins used for average position calculation can increase from 8 bins to 64 bins and from 9 distances to 73 distances. The number of bins and the number of centers is described by the following relationships,

$$B = \left(2^D\right)^L \qquad T = \sum_{i=0}^{L-1} \left(2^D\right)^i. \tag{8.1}$$

Where B is the number of bins, D is the number of dimensions the geometry occupies, L is the level of divisions, and T is the number of coordinates or distances that are used to determine the *RAPI*. For three-dimensional problems D is equal to three. The same calculation used to obtain the *RAPI* at each iteration will be used and is found in equation (8.2).

$$D_{xyz}^j = \sum_{i=1}^{T} d_i^j = \sum_{i=1}^{T} \sqrt{\left(x_{avg-i}^j - x_{avg-i}^1\right)^2 + \left(y_{avg-i}^j - y_{avg-i}^1\right)^2 + \left(z_{avg-i}^j - z_{avg-i}^1\right)^2} \tag{8.2}$$

By increasing the number of distances used for the *RAPI*, the fluctuation in *RAPI* may decrease allowing for quicker determination of the active cycle by the step refined on-the-fly judgment of the source distribution's state. The effects on the *RAPI* by increasing T needs to be investigated to determine if investing more computational resources by expanding the number of coordinates is necessary and useful for resolving the cycle at which the source distributions reaches stationarity.

Computational resources and time are important in Monte Carlo calculations. As mentioned in Chapter 4, using techniques from graph theory can significantly increase the computational time of a problem and using these techniques in the currently programmed form, exceeds the limits of array size.

The limit imposed by the upper bound of array size can be avoided by changing how the distances between the particles array is used. In current programming the array that defines the distances between all the particles in the geometry is defined by a double precision real N by N array. Currently the size of the array that can be used in the MC research code is 16000 by 16000. The amount of memory for this array can be reduced by performing the distance calculation in double precision and then storing the distance value between any two particles as single precision real in an N by N array. By using single precision storage, the number of particles used in Prim's algorithm can be increased to $16000 \times \sqrt{2} = 23000$. This only slightly increases the number of particles that can be used by Prim's algorithm. This will be useful if six digits of precision are enough to describe the problem geometry. The way that the data is accessed by Prim's algorithm could also be altered. Another improvement to memory storage could be to write the distance array to file instead of storing the data in a temporary array. Prim's algorithm

could then access the file, for its determination of the EMST, instead of an array, but this possibility was not investigated as a means of avoiding runtime memory shortages.

In Monte Carlo the computational time increases linearly with respect to the number of particles per cycle, or $O(N)$. When Prim's algorithm is employed to find EMST from particle positions the computational time increases as the square of the number of particles, or $O(N^2)$. To reduce the amount of time needed to perform Prim's algorithm, the geometry can be automatically divided into 8 sub-regions by the use of the automatic meshing technique. This technique finds the average particle position, (x_A, y_A, z_A), in the geometry and this average position divides the geometry into 8 regions. These regions are defined by R1: $x > x_A, y > y_A, z > z_A$ (top-upper-right), R2: $x > x_A, y < y_A, z > z_A$ (top-lower-right),…, R8: $x < x_A, y < y_A, z < z_A$ (bottom-lower-left). Next, use Prim's algorithm to find the EMST of the particles in each region. This will construct eight EMST's. From these eight EMST, eight average edge lengths and eight EMST total costs are found. Next choose a representative particle from each region and create a ninth EMST from these 8 particles. From this ninth EMST a ninth total cost and ninth average edge length will be obtained. By combining the nine EMST's a super EMST will be constructed with the average edge length calculated from the average edge lengths of the nine EMST's and the total cost of the EMST calculated by the sum of the edge lengths of the super EMST. These nine *AEL*s are averaged with weights according to the number of particles in each of the trees. The total cost of the super EMST would be a representation or approximation to the total cost of an EMST constructed from all the particles in the geometry. "The average of nine *AEL*s is approximation larger than the true *AEL* obtained if Prim's algorithm is applied to the collection of all particles" thus

making this a conservative approximation of the average edge length of the EMST that would be computed for the entire geometry [17]. By using these approximations, the time needed to compute the average edge length and total cost of an EMST is reduced by a factor of 8, i.e. $(N/8)^2 * 8 = N^2/8$. The divisions can be taken one step further and each of the eight regions subdivided into eight more regions for a total of 64 regions. Doing this will reduce the computational time by a factor of 64 or $(N/64)^2 * 64 = N^2/64$. The approximation of the total cost of an EMST can then be used as an indicator of when the source distribution reaches a stationary state, and the conservative approximation of the *AEL* can be used in the particle population diagnostic of the tally cell volume in power distribution calculations. Using the approximation of the *AEL* will lead to a larger estimate of the minimum tally cell volume making the diagnosis conservative. The only challenge with these approximations would be in choosing the representative particle from each of the regions to retain the conservative approximation of the *AEL* and the representative nature of the approximation to the total cost of the EMST.

The final issue is the refinement of the power law fit of the average edge length found in Chapters 4 and 7. The total cost of an EMST, $L_{EMST}(N)$, can be defined as

$$L_{EMST}(V_N) \equiv \min_{M(V_N)} \sum_{e \in M(V_N)} |e| \tag{8.3}$$

Where V_N is the set of all vertices in the graph represented by particle positions, $M(V_N)$ is all possible sets of edges in an acyclic and connected graph and e is the edges in this

graph connecting the particle positions. Equation (8.3) can be generalized resulting in equation (8.4).

$$L^{p}_{EMST}\left(V_N\right) \equiv \min_{M(V_N)} \sum_{e \in M(V_N)} |e|^{p} \tag{8.4}$$

A limit law of this equation has been discussed by Redmond and Yukich [19].

$$\lim_{N\to\infty} L^{p}_{EMST}(V_N) \big/ N^{(d-p)/d} = \beta(p,d) \int_{[0,1]^d} g(x)^{(d-p)/d}\, dx \tag{8.5}$$

In this equation $\beta(p,d)$ is a constant that depends only on p and d, and $g(x)$ is the absolutely continuous part of the probability density function of the vertices in V_N whose domain is presumed to be in $[0,1]^d$. Values of p and d corresponding to EMST are found to be p=1 and d=3. By taking this values equation (8.5) becomes,

$$\lim_{N\to\infty} L^{1}_{EMST}(V_N) \big/ N^{2/3} = \beta(1,3) \int_{[0,1]^d} g(x)^{2/3}\, dx \tag{8.6}$$

The domain of the Monte Carlo source can be scaled to be contained in $[0,1]^d$ and by combining equation (8.7) with equation (8.6),

$$AEL(N) = \frac{L_{EMST}\left(V_N\right)}{N-1}, \tag{8.7}$$

one can arrive at equation (8.8).

$$AEL(N) = \frac{L^{1}_{EMST}(V_N)}{N-1} \simeq N^{-1/3} \beta(1,3) \int_{[0,1]^d} g(x)^{2/3}\, dx \quad \text{as N} \to \infty \tag{8.8}$$

This equation implies that the average edge length of an EMST decreases by $N^{-1/3}$, regardless of material composition, as the particle population is increased to infinity. It

can be guessed that the average edge length approach to $N^{-1/3}$ is very slow from the power law fitting figures found in Chapter 7. In the power law theory found in Chapter 4 the average edge length is proportional to $N^{-1/b}$. Because of the relationship found in equation (8.8), modeling needs to be performed to investigate how the exponent -1/*b* approaches -1/3. It can be done, but it is beyond the scope of this thesis.

Chapter 9

Conclusion

The goal of this thesis was to analyze and demonstrate the performance of mesh-input-free diagnostics of the source distribution and particle population. The mesh-input-free convergence indicator, *RAPI*, of the source distribution was compared against established convergence indicators for three different cases. From this comparison, it was found that *RAPI* performed as well as or better than the established methods for the determination of the cycle at which the source distribution reached a stationary state. As a result of the *RAPI* performance, mesh-free diagnosis of the source distribution's state shows potential for use in reducing error in Monte Carlo calculations and eliminating the need for mesh based diagnostics of the source distribution. Because of the promise demonstrated with this technique, expansion and further development of this method is warranted.

From exploring graph theory techniques for convergence diagnostics, a mesh-free diagnostic of the particle population in a simulation was developed. This diagnostic used the average edge length of an EMST to develop indicators to the sufficiency of the size of

a particle population in power distribution calculations. By comparing this diagnostic to a previously validated diagnostic, PD_MESH, the meshless EMST-based diagnostic was found to be a suitable method of determining an adequate particle population for a model with a given tally cell volume in power distribution calculations. With the promising performance of the EMST based diagnostic, the need for a mesh-based diagnostic can be eliminated, simplifying the diagnostic process. However, because the use of graph theory techniques in Monte Carlo methods has been limited, incorporating such techniques needs to be further investigated and improved.

References

[1] Ueki, T., "Information Theory and Undersampling Diagnostics for Monte Carlo Simulation of Nuclear Criticality", *Nuclear Science and Engineering*, v 151, 283-292, 2005.

[2] Brown, F. and T. Ueki, "Stationarity Modeling and Informatics-Based Diagnostics in Monte Carlo Criticality Calculations", *Nuclear Science and Engineering*, v 149, 38-50, 2005.

[3] Ueki, T., "Step-Refined On-the-Fly Convergence Diagnostics of Monte Carlo Fission Source Distribution", *Nuclear Science and Engineering*, v 163, 99-117, 2009.

[4] Hero, A.O. and O.J.J. Michel, "Asymptotic Theory of Greedy Approximations to Minimal k-Point Random Graphs", *IEEE Transactions on Information Theory*, 45, 6, 1921-1938, 1999.

[5] Hero, A.O. and O.J.J. Michel, "Estimation of Renyi Information Divergence via Pruned Minimal Spanning Trees", *Proc. IEEE Workshop on Higher Order Statistics*, Caesaria, Israel, 1999.

[6] Sabuncu, M.R., and P. Ramdage, "Using Spanning Graphs for Efficient Image Registration," *IEEE Transactions on Image Processing*, 17, 5, 788-797, 2008.

[7] Hoffman, R. and A.K. Jain, "A Test of Randomness based on the Minimum Spanning Tree", *Pattern Recognition Letters*, 1, 175-180, 1983.

[8] Dussert, C., G. Rasigni, M. Rasigni and J. Palmari, "Minimal Spanning Tree: A New Approach for Studying Order and Disorder", *Physical Review B*, 34, 5, 3528-3532, 1982.

[9] Ueki, T., "On-the-Fly Diagnostics of Particle Population in Iterated-Source Monte Carlo Methods", *Nuclear Science and Engineering*, v 158, 15-27, 2008.

[10] Appelbaum, D., *Probability and Information Theory: An Integrated Approach*, Cambridge University Press, New York, NY, 1996.

[11] Cover, T.M. and J.A. Thomas, *Elements of Information Theory*, 2nd ed., John Wiley & Sons, Hoboken, New Jersey, 2006.

[12] Ueki, T., F.B. Brown, D.K. Parsons, and D.E. Kornreich, "Autocorrelation and Dominance Ratio in Monte Carlo Criticality Calculations", *Nuclear Science and Engineering*, v 145, 279-290, 2003.

[13] X-5 Monte Carlo Team, "MCNP — A General Monte Carlo N-Particle Transport Code, Version, Volume I: Overview and Theory", LA-UR-03-1987, Los Alamos, NM, 2003.

[14] Clark, J. and D. Holton, *A First Look At Graph Theory*, World Scientific Publishing Company, Hackensack, NJ, 1991.

[15] Whitesides, G.E., "A Difficulty in Computing the k-effective of the World", *Trans. Am. Nucl. Soc.*, 14, 68, 1971.

[16] Warsa, J.S., T.A. Wareing, J.E. Morel, J.M. McGHEE and R.B. Lehoucq, "Krylov Subspace Iterations for the Calculation of k-Eigenvalues with S_N Transport Codes", *M&C 2003 proceedings*, ANS, 2003.

[17] Ueki, T. and B. Chapman, "Particle Population Diagnostics in Monte Carlo Calculation of Power Distribution", submitted for publication, 2010.

[18] Nakagawa, M. and T. Mori, "Whole Core Calculations of Power Reactors by Use of Monte Carlo Method," *Journal of Nuclear Science and Technology*, 30[7], 692-701, 1993.

[19] Redmond, C. and J.E. Yukich, "Asymptotics for Euclidean Functionals with Power-Weighted Edges", *Stochastic Processes and their Applications*, 61, 289-304, 1996.