

**INVESTIGATION AND IMPROVEMENT OF CRITICALITY  
CALCULATIONS IN MCNP5 INVOLVING SHANNON ENTROPY  
CONVERGENCE**

A Thesis  
Presented to  
The Academic Faculty

by

David Jeffrey Koch

In Partial Fulfillment  
of the Requirements for the Degree  
Master of Science in Nuclear and Radiological Engineering in the  
School of Mechanical Engineering

Georgia Institute of Technology  
May 2015

**COPYRIGHT 2014 BY DAVID KOCH**

**INVESTIGATION AND IMPROVEMENT OF CRITICALITY  
CALCULATIONS IN MCNP5 INVOLVING SHANNON ENTROPY  
CONVERGENCE**

Approved by:

Dr. Bojan Petrovic, Advisor  
School of Nuclear and Radiological Engineering  
*Georgia Institute of Technology*

Dr. Nolan Hertel  
School of Nuclear and Radiological Engineering  
*Georgia Institute of Technology*

Dr. Dingkang Zhang  
School of Nuclear and Radiological Engineering  
*Georgia Institute of Technology*

Date Approved: November 24, 2014

[To Annabel Lee, for her unrivaled support and love through the years.]

## **ACKNOWLEDGEMENTS**

I would like to personally thank my advisor, Dr. Bojan Petrovic, whose guidance, knowledge, and assistance was instrumental in completing my thesis work. I would like to show my deepest gratitude to Dr. Nolan Hertel and Dr. Dingkang Zhang for serving as my committee members. Additionally I wish to thank my parents, Nicholas and Eileen Koch, whose continued support has been vital to my time at Georgia Tech.

# TABLE OF CONTENTS

	Page
ACKNOWLEDGEMENTS	iv
LIST OF TABLES	vii
LIST OF FIGURES	viii
SUMMARY	ix
<u>CHAPTER</u>	
1 INTRODUCTION	1
1.1 Monte Carlo Methods	2
1.2 Shannon Entropy	3
2 METHOD	5
2.1 MCNP5 Shannon Entropy Source Convergence Estimation	5
2.2 Other Source Convergence Estimation Methods	7
2.3 Method Proposed for this Study	8
3 ANALYSIS OF A 2X2 PIN ARRAY	11
3.1 Model Description	11
3.2 Results for 2x2 Pin Array	15
4 MONTE CARLO BENCHMARK PROBLEM FOR FULL-SIZE REACTOR CORE	21
4.1 Model Description	21
4.2 Preliminary Results from 1k and 10k Runs	25
4.3 Results from 100k Run	29
4.4 Applying Procedure to 1mil Run	33
4.5 Confirming Source Convergence in the 1mil Run	36

5	CONCLUSIONS	43
	5.1 Future Work	44
APPENDIX A:	EXAMPLE MCNP5 INPUT OF 2X2 PIN ARRAY	46
APPENDIX B:	EXAMPLE BENCHMARK MODEL OF FULL CORE MCNP5 INPUT	50
REFERENCES		56

## LIST OF TABLES

	Page
Table 3.1: Radial dimensions	14
Table 3.2: Material density and composition	14
Table 3.3: Cycle within one st. dev. of the average of the last half of cycles	20
Table 4.1: Uranium oxide fuel composition	25
Table 4.2: $k_{\text{eff}}$ and Shannon entropy results from 1k and 10k runs	26
Table 4.3: $k_{\text{eff}}$ and Shannon entropy results from 100k run	30
Table 4.4: $k_{\text{eff}}$ and Shannon entropy results from 1mil run	33
Table 4.5: Symmetry of energy deposition per fuel assembly, axially integrated over the entire assembly	39
Table 4.6: Relative error of energy deposition for symmetrically located fuel assemblies	40
Table 4.7: Calculated relative error of symmetric fuel assemblies	41
Table 5.1: CPU run time	43

## LIST OF FIGURES

	Page
Figure 3.1: Axial view of fuel pins	12
Figure 3.2: Radial view of fuel pins	13
Figure 3.3: Distributed source case	16
Figure 3.4: Middle source case	17
Figure 3.5: Edge source case	18
Figure 4.1: Benchmark model (horizontal cross section)	22
Figure 4.2: Benchmark model (vertical cross section)	23
Figure 4.3: Fuel assembly (horizontal cross section)	24
Figure 4.4: $k_{\text{eff}}$ of 1k and 10k results	27
Figure 4.5: Shannon entropy of 1k and 10k results	28
Figure 4.6: $k_{\text{eff}}$ (track length) versus cycle of preliminary 100k run	31
Figure 4.7: Shannon entropy of preliminary 100k run	32
Figure 4.8: $k_{\text{eff}}$ (track length) versus cycle of 1mil run	34
Figure 4.9: Shannon entropy per cycle of 1mil run	35
Figure 4.10: Heat map of energy deposition for each fuel assembly	37

## SUMMARY

Monte Carlo methods are a staple in simulating nuclear systems, and with the recent advances in computing technology, these methods are particularly useful for modeling both shielding and criticality simulations. For Monte Carlo criticality simulations, it is necessary for the proposed source distribution to be converged before tallying results, as the initial source is generally not converged to a steady-state value. Shannon entropy is a concept that comes from information theory and is a useful concept for determining source convergence as it tends to converge to a single value alongside the source distribution. Shannon entropy has been introduced in many techniques for determining source convergence including MCNP5's source convergence criteria. None of these techniques are infallible and there is much room for improvement in both accuracy and efficiency.

This thesis aims to improve upon existing Shannon entropy convergence techniques by reducing the overall time of Monte Carlo criticality calculations and eliminating any "guessing" of when source convergence will occur. The proposed technique in this thesis improves upon efficiency of other techniques as it does not require substantial computational resources and uses shorter runs to predict when source convergence will occur for a desired simulation. The purpose of this thesis is to develop a model showing how one can use this concept and produce a streamlined approach for applying this concept to a criticality problem.

# CHAPTER 1

## INTRODUCTION

With current advances in computing technology, Monte Carlo (MC) methods are becoming increasingly more appealing for modeling nuclear reactors. The Monte Carlo N-Particle Transport Code (MCNP) is a popular MC code developed by Los Alamos National Lab (LANL). MC methods are desirable for modeling nuclear reactor cores because they can represent complex three-dimensional geometries as well as treat a model using continuous energy, space, and angle. However, in order to reduce statistical uncertainty in the results, a large number of particles must be simulated, which becomes very computationally intensive. A key feature of MCNP is that it can perform calculations on nuclear criticality. The calculations solve for  $k_{\text{eff}}$ , which is the ratio of the number of neutrons in successive generations and the eigenvalue of the neutron transport equation. In non-criticality calculations, a particle is tracked from a specified source until it is absorbed or leaves the system, and the code moves on to the next particle. Criticality calculations add an additional layer of depth. Criticality calculations include a fissionable material and calculate  $k_{\text{eff}}$  as the average number of fission neutrons produced from one fission neutron. A generation or particle history is considered to be the life of a neutron from birth in fission to death by escape, parasitic capture, or absorption leading to additional fission [1]. The user inputs an initial fission source distribution, and after the batch or cycle of the specified number of generations has been run, MCNP determines what the new fission source distribution is. This process is repeated for the specified number of cycles and the fission source continues to converge towards a steady state (fundamental mode). The user must determine whether or not the fission source has converged before tallying results. Techniques have been developed to help users understand when a model has converged.

Shannon entropy is a useful tool from information theory that has been added to the MCNP code to aid in determining when convergence has occurred. There are still limitations regarding the Shannon entropy and this research aims to add to and improve upon existing Shannon entropy methods that have been applied to MCNP.

## 1.1 Monte Carlo Methods

For solving the neutron transport equation, there are deterministic methods and Monte Carlo (MC) methods [2] [3]. Deterministic methods solve the equation using various mathematical numerical methods for the average particle behavior, whereas MC methods track each particle individually using predetermined nuclear data libraries to determine the average particle behavior. MC codes then tally the results based on the user's specifications. MC methods are based on probability and therefore have an associated statistical uncertainty with each value. To reduce the uncertainty and provide more precise results, a large number of particles are required to be simulated. This in turn causes MC codes to generally be more computationally intensive than deterministic codes. Advances in computer technology have allowed MC codes to be more feasible for simulating more particles in a shorter amount of time.

MC methods can be applied to any statistical process. The interaction of nuclear particles with a material is simply a probability distribution for specific events that can occur according to their predetermined transport data, and thus one can model this with MC methods for even the most complex scenarios [1].

Criticality calculations involve a sustainable chain reaction of fission neutrons. In MC criticality calculations, it is necessary to converge the source distribution before tallying results; otherwise, one would be tallying over the wrong data. The source should be converged during the inactive or skipped cycles. Earlier work in criticality calculations used  $k_{\text{eff}}$  convergence as an indicator for source convergence; however, this is not a good or ideal indicator.  $K_{\text{eff}}$  is an integral parameter represented in a single

global value over the entire model and does not describe a localized area of the source region. Each part of the source region must be converged in order for the source to have wholly converged. The more accepted approach for determining source convergence is to use a concept called the Shannon entropy.

## 1.2 Shannon Entropy

The Shannon entropy is a concept developed from information theory [4] that has been adapted to suit MC methods. The Shannon entropy is defined as

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

Here,  $B$  represents the number of meshes used to divide the entire system,  $i$  is the index number of each mesh, and  $S^B(i)$  is the portion of source generated in the  $i^{th}$  mesh after a certain cycle. In MC methods, the Shannon entropy represents the randomness of a system and is used as a means for estimating source convergence. It has been shown [5] that the source distribution converges concurrently with the Shannon entropy. MCNP5 calculates the Shannon entropy as a single value on a per cycle basis. These values can be graphed against the cycle number and MCNP5 produces an estimate for the number of cycles that should be skipped according to its calculations. How MCNP5 calculates this estimate will be discussed in Chapter 2.

The purpose of this research is to develop a technique to aid users in determining when source convergence will occur without requiring substantial computations. The idea here is that by using a computationally low intensive simulation to first determine how many cycles are required for convergence, a user can save time in the reference simulation with many particles rather than guessing how many cycles to skip. This is possible because the Shannon entropy tends to converge at a similar number of cycles when a fewer particles are used per batch. By using this idea, one can perform a shorter run with fewer particles per cycle to find how many cycles are required for the Shannon entropy to converge and then use this information to predict when source convergence

will occur for a longer run with more particles per cycle. This thesis will look to validate this approach of performing shorter runs (shorter by an order of magnitude or more) to give an indication of when source convergence occurs before performing longer runs.

In Chapter 2, the method proposed for this thesis will be introduced and investigate previous studies that involve determining Shannon entropy convergence. Chapter 3 will apply the method proposed to a simple 2 by 2 pin case for a proof of principle that this technique works. Chapter 4 will show the results when applied to a larger, more realistic model, and Chapter 5 will address conclusions and future work regarding this topic.

## CHAPTER 2

### METHOD

#### 2.1 MCNP5 Shannon Entropy Source Convergence Estimation

In MC codes it is necessary to allow the source distribution to converge and reach stationarity before tallying over quantities of interest. There was discussion with regard to identifying  $k_{\text{eff}}$  convergence based on a Brownian bridge [6]; however, this quantity based on  $k_{\text{eff}}$  did not provide an accurate representation of the source distribution. Ueki and Brown proposed using the Shannon entropy of the source distribution as a better representation of the source distribution [5]. MCNP5 automatically computes the Shannon entropy of the fission source distribution to aid users in assessing the convergence of the fissions source spatial distribution. MCNP5 provides a single number,  $H(S^B)$ , for the Shannon entropy of each cycle to determine convergence. The Shannon entropy for each cycle is calculated as

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

A 3-dimensional mesh is superimposed over the fuel region. Here  $B$  is the total number of meshes and  $i$  is the mesh index. Tallies for the number of fission sites in each mesh are taken and then used to form a discretized estimate of the source distribution, which is represented by  $S^B(i)$ , or simply the number of source sites in the  $i$  mesh divided by the total number of source sites. The mesh size can be submitted by the user; the MCNP5 manual recommends using a small number of meshes (e.g., 5-10 for each XYZ direction), chosen according to the symmetry of the problem. If the user does not manually input a

Shannon entropy mesh size, MCNP5 will automatically create a mesh that encompasses all the fission source sites for a cycle. This automatic mesh will expand if necessary for later cycles and the total number of meshes will be equal to the number of particle histories per cycle divided by 20 and then rounded to the nearest integer to form equal-sized meshes. Choosing a proper mesh size can prove difficult. A finer mesh will have higher fluctuations in local entropy since there will be fewer particles in each mesh. This will in turn cause slower Shannon entropy convergence and require more cycles to reach stationarity. A coarser mesh will converge faster over fewer cycles, but may not accurately represent local fluctuations, and therefore may produce a false convergence.

MCNP5 uses a fairly simple technique to calculate Shannon entropy convergence.

The MCNP5 Manual [1] states:

*Upon completion of the problem, MCNP will compute the average value of  $H(S^B)$  for the last half of the active cycles, as well as its (population) standard deviation. MCNP will then report the first cycle found (active or inactive) where  $H(S^B)$  falls within one standard deviation of its average for the last half of the cycles, along with a recommendation that at least that many cycles should be inactive. Plots of  $H(S^B)$  vs. cycle should be examined to further verify that the number of inactive cycles is adequate for fission source convergence.*

MCNP5 also states that for criticality calculations, users should look at the convergence of both  $k_{\text{eff}}$  and the fission source distribution before using active cycles to tally results. It is important to note that convergence is increasingly more difficult to judge for lesser particle runs as the statistical noise is higher. In these cases, the Shannon entropy appears

to converge sooner as a high noise doesn't allow for a very tight convergence and there is no reason to converge better than the noise fluctuations.

It should be noted that these studies were performed using MCNP5, even though the latest version of MCNP is MCNP6. Any additional features that were added to the latest version will not be discussed here. One drawback with MCNP5's Shannon entropy calculations is that they are performed after the completion of the run, rather than on-the-fly. This may require discarding the results of a large simulation and re-running with different number of cycles skipped. Several advancements in on-the-fly and entropy calculation improvements will be discussed in Section 2.2; however, for the purposes of this research, MCNP5's Shannon entropy calculations after completion of the run will be sufficient.

## **2.2 Other Source Convergence Estimation Methods**

The technique used by MCNP5 to determine source convergence can often be misleading. If a small number of particles are used, MCNP5 will often recommend the user use fewer inactive cycles than are necessary. A simple alternative technique, which MCNP5 actually recommends [1], is visual inspection of the  $H(S^B)$  vs. cycle plot. This technique is rather useful since it can be easy to determine whether or not  $H(S^B)$  has reached a steady-state value. The problems with this technique are that it is often difficult to determine exactly when stationarity has been achieved, and it may be difficult to judge the difference between stationarity and slow convergence of the source distribution. Even if a specific value cannot be determined, visual inspection can produce an estimated value for where the Shannon entropy converges.

More computationally driven methods have been developed for assessing source convergence. Kitada and Takeda propose using a fission matrix as a convergence diagnostic metric [7]. An on-the-fly calculation was developed by Shim and Kim based on inter-cycle correlation length [8]. Simpler posterior diagnostics with statistical diagnostic checks on the Shannon entropy were researched by Brown et al. to appeal to a large user base [9]. Ueki used the Wilcoxon signed rank sum to create an on-the-fly convergence criterion [10]. Romano proposed using the stochastic oscillator, an indicator used in financial markets, to assess the Shannon entropy convergence [11]. In his paper, Romano also states that

*“While visual inspection of a line-plot of the Shannon entropy is certainly a viable method of assessing source convergence, it places an unnecessary burden on the reactor analyst and necessitates making a subjective decision on how many batches to discard.”*

### **2.3 Method Proposed for this Study**

For this study, visual inspection is primarily used as the means to estimate source convergence, and the default MCNP5 convergence criterion is used for comparative purposes. The principal idea here is to develop a simple systematic approach to allow MCNP users to first determine how many inactive cycles to skip based on a short run before running the full, long problem. The goal is to reduce the overall run time by eliminating any guessing for how many cycles to skip and instead spending approximately one-tenth to one-fifth of total time determining the appropriate number of inactive cycles and then running the full problem.

Shannon entropy behavior is not directly dependent on the number of particles per cycle. When using the same mesh sizes, it is conjectured (and illustrated by numerical simulations) in this study that the Shannon entropy of a run with fewer particles per cycle will converge in a similar fashion to a run where more particles per cycle are used. This is the driving force behind this method as one could first find the point at which the Shannon entropy converges for the shorter run and use it to predict Shannon entropy behavior for a longer run. If that point does not occur during the run, continue-runs can be performed to add additional cycles until a point of convergence can be determined. Once this point is found, one should theoretically be able to skip the appropriate number of cycles for a longer run without guessing. When using convergence estimations that are calculated automatically, one could further automate the process to run a longer run with the appropriate number of skipped cycles once a point of convergence has been determined from the short run. However, for the purposes of this paper, visual inspection is used to determine source convergence. Thus, the general method here is to simulate a shorter run with the option of additional continue-runs until a point of convergence has been identified by visual inspection, and then use this information to simulate a longer run with the appropriate number of skipped cycles. As mentioned above, this shorter run would only take approximately one-tenth to one-fifth the total time of the full, longer run, and would reduce any guessing when deciding how many cycles to skip and would also eliminate the need to rerun an entire run if too few cycles are skipped.

Chapter 3 will propose a simple model that introduces this idea of using fewer particles to determine the number of cycles to skip, and showing that a longer run with more particles has to skip roughly the same number of cycles. Chapter 3 will show a

proof of principle that this technique can be effective and then in Chapter 4 this technique will be applied to a larger, more realistic model.

## CHAPTER 3

### ANALYSIS OF A 2X2 PIN ARRAY

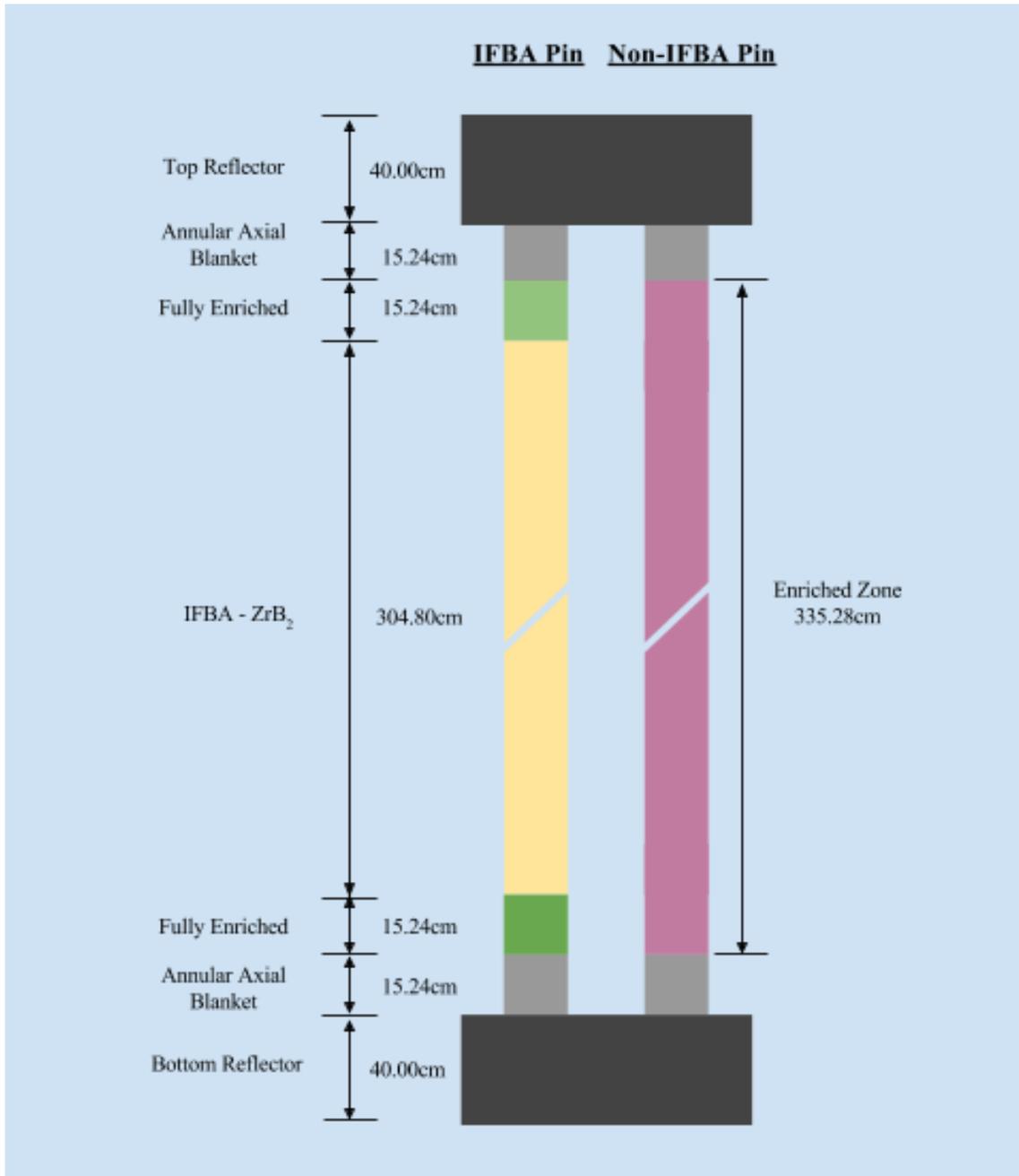
A simple model was created to analyze the approach of using the entropy of shorter runs, with a lesser number of particle histories per cycle, to predict the entropy convergence, and consequently source convergence, of longer runs. This will serve as an introductory case before applying this concept to a larger, more realistic model. The objective is to confirm that the method proposed in Chapter 2 applies to simple cases before applying it to more realistic models. The model was run for three different initial source distributions to show its effect on Shannon entropy convergence. Each of these initial source distributions should propagate throughout the source region at a different rate.

#### 3.1 Model Description

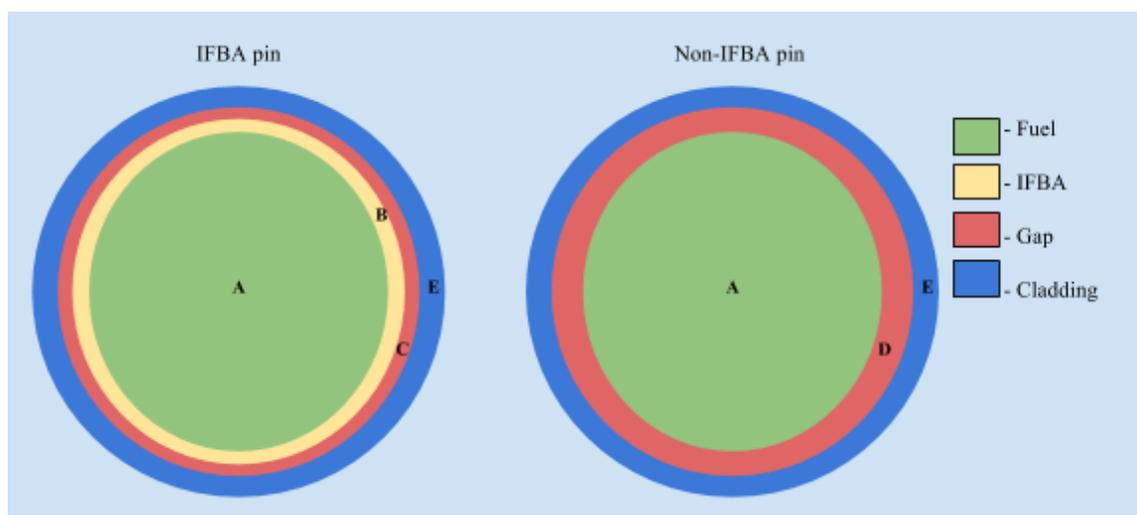
The model represents a square four-pin arrangement in a 2x2 configuration with two diagonal pins having burnable absorbers of the IFBA type. The pins were modeled after typical Westinghouse IFBA pins for light-water reactors [12]. Figure 3.1 shows an axial view of both the IFBA and non-IFBA fuel pins. As shown in the figure, there was a fully enriched zone with a total height of 335.28cm, an IFBA region centered over the enriched zone on the IFBA pins with a height of 304.80cm, annular axial blankets on either side of the enriched zone with a height of 15.24cm each, and top and bottom reflectors with a height of 40.00cm each.

Figure 3.2 shows a radial cut of one of the IFBA pins and one of the non-IFBA pins. The IFBA was comprised of a thin layer of zirconium diboride,  $ZrB_2$  coating on regular fuel pellets. All four fuel pins had a small pellet-to-cladding gap and were encased in a Zircaloy-4 cladding. Above and below the pins were reflectors consisting of

a 50% stainless steel (SS 304) and 50% water mixture. The corresponding dimensions for Figure 3.2 are listed in Table 3.1.



**Figure 3.1: Axial view of fuel pins**



**Figure 3.2: Radial view of fuel pins**

The center-to-center pitch of the pins was 1.26cm, and the moderator used was light water at 600K with a constant density of  $0.705\text{g/cm}^3$ . This density of the light water moderator was intentionally left constant to enable checking of how close the solution was to symmetry. The outer boundary of the model used periodic reflective boundary conditions. Table 3.2 shows the material densities and compositions for each material used in the MCNP5 input. The IFBA pins used 4.94w/o  $\text{UO}_2$  and the non-IFBA pins used 2.074w/o  $\text{UO}_2$ , emulating fresh fuel with burnable absorbers in the IFBA pins and once burnt fuel in the non-IFBA pins. Initially, the IFBA region's density was set in accordance to the linear content of  $2.35\text{mg }^{10}\text{B/in}$ , as given by the Westinghouse LWR pin design [12]; however, this value caused the axial flux profile to be too depressed along the IFBA region of the pins, and the problem to be too loosely-coupled. The density of the IFBA was set to  $0.416\text{ g/cm}^3$  to create a flux profile closer to a cosine shape and a more tightly coupled system. An example input for this 2x2 pin array is shown in Appendix A.

**Table 3.1: Radial dimensions**

<b>Region</b>	<b>Inner Radius</b>	<b>Outer Radius</b>	<b>Radial Thickness</b>
A – Fuel	0.0000cm	0.3951cm	0.3951cm
B – IFBA	0.3951cm	0.3991cm	0.0040cm
C – IFBA Gap	0.3991cm	0.4010cm	0.0019cm
D – Non-IFBA Gap	0.3951cm	0.4010cm	0.0059cm
E – Cladding	0.4010cm	0.4583cm	0.0573cm

**Table 3.2: Material density and composition**

<b>Material</b>	<b>Density (g/cm<sup>3</sup>)</b>	<b>Composition (percentages in weight percent)</b>
IFBA fuel	10.24	UO <sub>2</sub> , 4.94% enriched
Non-IFBA fuel	10.24	UO <sub>2</sub> , 2.074% enriched
Axial blanket	10.24	Natural Uranium - 0.711% UO <sub>2</sub>
IFBA	0.416	Zirconium Diboride - ZrB <sub>2</sub>
Gap	0.001654	Helium
Cladding	6.504	Zircaloy-4: 98.23% nat-Zr, 1.45% nat-Sn, 0.21% nat-Fe, 0.10% nat-Cr, 0.01% nat-Hf
Moderator	0.705	Light Water
Top/bottom reflectors	4.50	50% Light Water and 50% SS304 SS304: 74% nat-Fe, 18% nat-Cr, 8% nat-Ni

Three cases were created using different initial source distributions. The first case used an evenly distributed source along the fuel region (“distributed case”), the second case placed the initial source in the middle 1/100<sup>th</sup> of the fuel region (“middle case”), and the third case placed the initial source in the bottom 1/100<sup>th</sup> of the fuel region (“edge case”). Each case was identical aside from the initial source distribution and each was

completed four separate times using one thousand (1k), ten thousand (10k), one-hundred thousand (100k), and one million (1mil) particles per cycle.

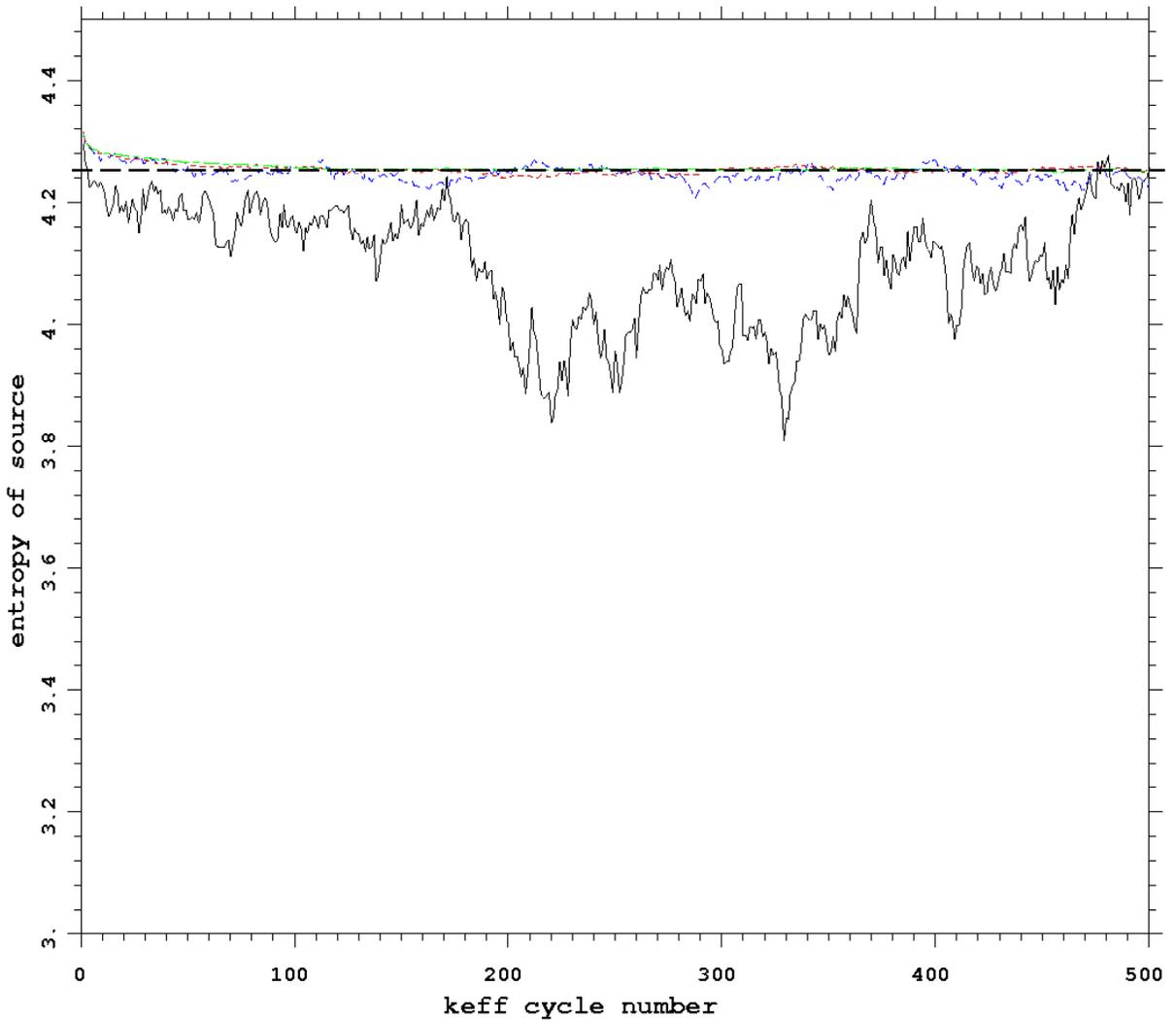
All runs were performed on the Linux cluster b2@neely.gatech.edu. For these runs, the 1k were performed on 8 CPUs over one node, 10k on 8 CPUs over one node, 100k on 16 CPUs over two nodes, and 1mil on 64 CPUs over eight nodes. Five hundred active cycles were completed for each run, with zero inactive cycles. Wall-clock run time varied with other activity on the cluster; however, under the cluster's best conditions, the 1k run over 8 CPUs took 1 minute 35 seconds to complete, the 10k run over 8 CPUs took 9 minutes, the 100k run over 16 CPUs took 52 minutes 18 seconds, and the 1mil run over 64 CPUs took 117 minutes 57 seconds.

### 3.2 Results for 2x2 Pin Array

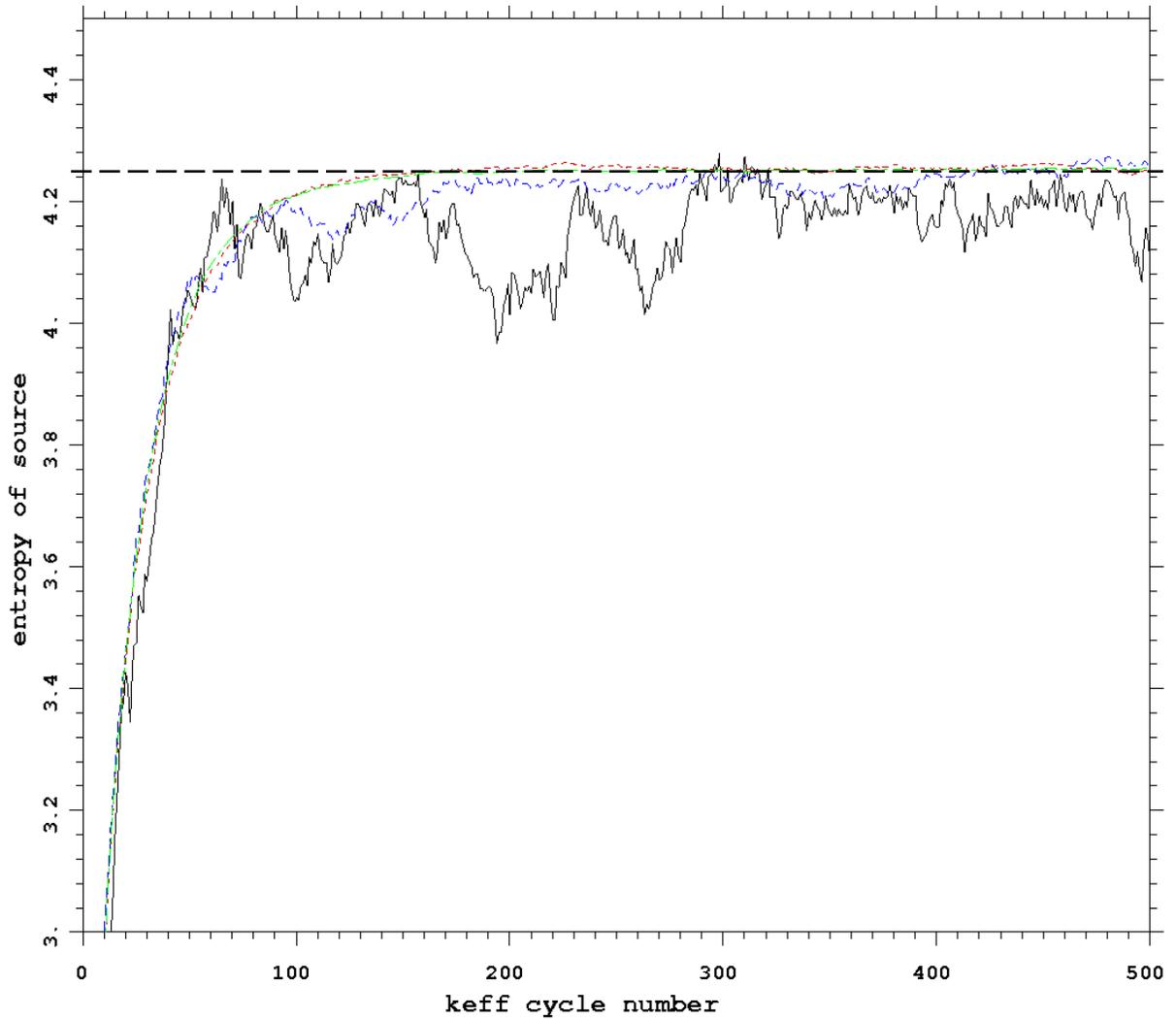
The purpose of this approach is to predict the Shannon entropy convergence of a longer run with more particles per cycle by using the results from a shorter run with fewer particles per cycle. Here we are interested in comparing the results of the 1k, 10k, and 100k particle runs against the 1mil particle run to see if one can use any of the first three to predict when the Shannon entropy of the 1mil run will converge. In each of the following figures, the Shannon entropy values for the 1k case are plotted in black, the 10k in blue, the 100k in red, and the 1mil in green. The first graph, Figure 3.3, shows the plot of Shannon entropy versus cycle number for the distributed case.

For this distributed case, it is initially clear that the 1k results are too noisy to provide any meaningful estimation of Shannon entropy convergence. Of the three cases, the distributed case is the closest to the true source distribution, and thus has the least amount of change in Shannon entropy from the beginning to the end of cycles. While it is difficult to judge a point of Shannon entropy convergence solely by visual inspection, it is evident that both the 10k and 100k runs converge in a similar fashion as the 1mil run. A black dotted line estimating the Shannon entropy value for convergence has been

added for visual comparison. Though difficult, the distributed case does appear to converge around 150 cycles for 10k, 100k, and 1mil particles per cycle. It is also worth noting that MCNP5's convergence criteria for Shannon entropy recommend skipping the first 27, 44, 59, and 114 cycles for the 1k, 10k, 100k, and 1mil runs, respectively, for the distributed case.



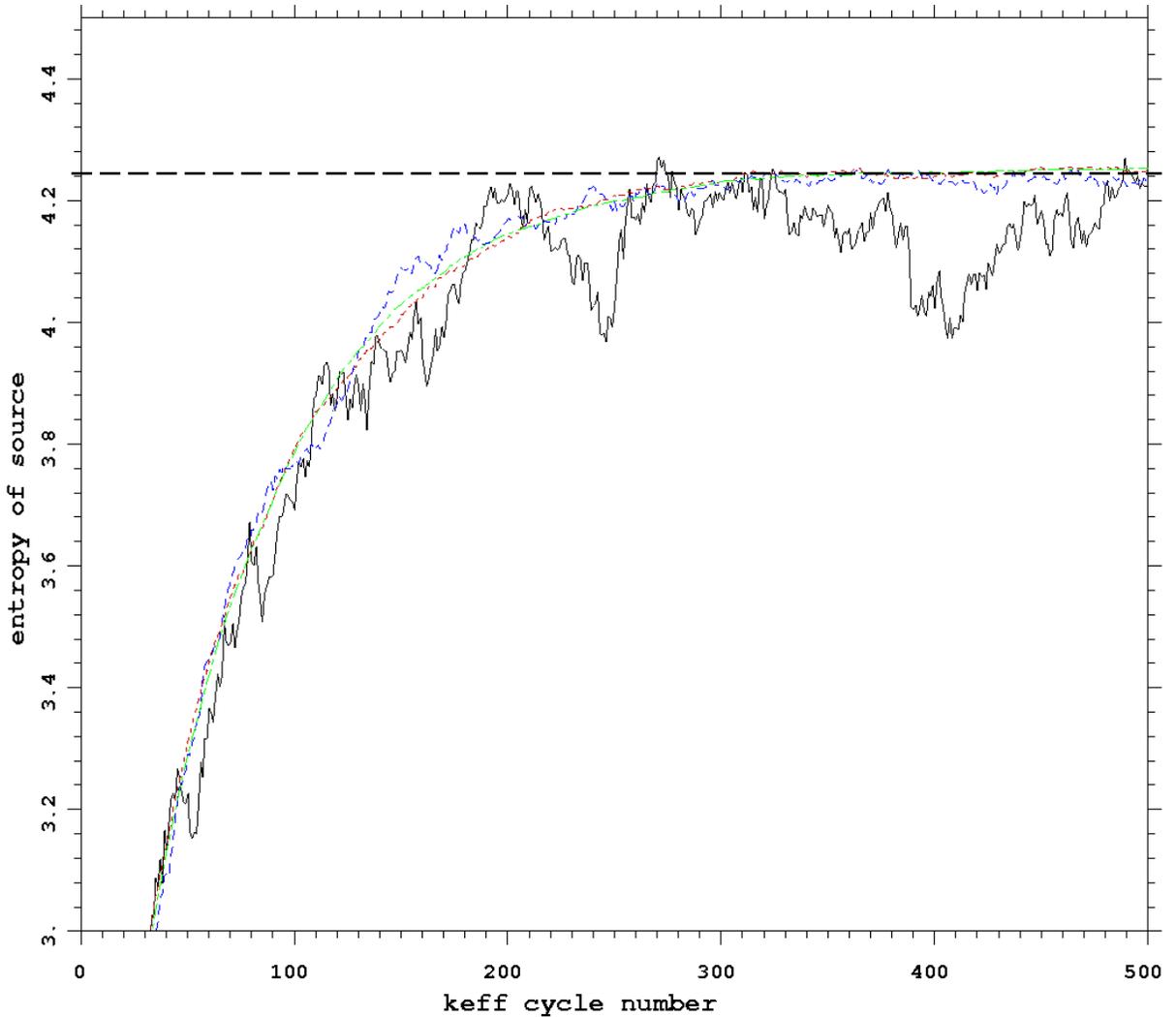
**Figure 3.3: Distributed source case**



**Figure 3.4: Middle source case**

Figure 3.4 shows the results Shannon entropy versus cycle number for the middle case, where an initial source is set in the middle  $1/100^{\text{th}}$  of the fuel region. This case is much easier to view convergence via visual inspection compared to the distributed case as the Shannon entropy rapidly converges over the first one hundred cycles before beginning to level off. Again, the 1k results appear to be too noisy to provide any meaningful results, but the 10k results behave in a similar manner to the 1mil results and the 100k results behave in a very similar manner to the 1mil results. In the 10k, 100k, and 1mil results for the middle case, one could verify that the Shannon entropy in all

three cases reaches a steady-state value around 200 cycles. Thus, if one were using either the 10k or 100k run before running a longer run, one could predict to use at least 200 inactive cycles. Additionally, MCNP5's convergence criteria recommend skipping at least 60, 169, 160, and 209 cycles, respectively, for the 1k, 10k, 100k, and 1mil runs.



**Figure 3.5: Edge source case**

Figure 3.5 displays the results of the Shannon entropy versus cycle number for the edge case. This case is similar to the middle case in that there is initially a rapid increase

in Shannon entropy before beginning to level off. As expected, this case takes more cycles than the other two to reach convergence. For this edge case, the steady-state convergence appears to occur around 300 cycles; however, upon closer inspection, it is clear that for 100k and 1mil, the Shannon entropy is still gradually increasing, even at 500 cycles. Since this case started with a source on the edge of the fuel region, it takes significantly longer to reach a steady-state value compared to the middle and distributed cases. A simple continue-run could be performed to increase the number of cycles if one wanted to find an estimate for Shannon entropy convergence. For the purpose of this simple model though, it is only necessary to show a proof of principle that the Shannon entropy of fewer particles runs, like the 10k and 100k runs, converge in a similar fashion to the greater particles runs, like the 1mil run. Once again, MCNP5's convergence criteria recommends skipping 181, 238, 288, and 292 cycles, respectively, for the 1k, 10k, 100k, and 1mil particle runs.

In all three cases, the 100k runs (in red) converge in a very similar fashion to the 1mil runs (in green) and have the greatest potential to provide a reasonable estimation for how many cycles to skip. The 10k runs (in blue) could also give a useful estimate, although it is more difficult to provide an accurate point from visual inspection for when the Shannon entropy converges. The 1k runs (in black) would not be recommended for use when estimating how many cycles to skip as there is far too much noise in all three cases to provide useful results.

One can also compare these results to those calculated in MCNP5. As mentioned in Chapter 2, MCNP5 calculates the Shannon entropy convergence by finding the first cycle having an entropy value within 1 standard deviation of the entropy of the last half of cycles. The cycle given will likely occur earlier than the true convergence since the MCNP5 only looks for the first value. Table 3.3 shows for all three cases the cycle number that MCNP5 estimates for Shannon entropy convergence.

**Table 3.3: Cycle within one st. dev. of the average of the last half of cycles**

	<b>1k</b>	<b>10k</b>	<b>100k</b>	<b>1mil</b>
Distributed Case	27	44	59	114
Middle Case	60	169	160	209
Edge Case	181	238	288	292

Table 3.3 shows that there are some inconsistencies in MCNP5's Shannon entropy convergence estimates. In particular, we can compare these to the visual inspection estimates of skipping at least 150 cycles for the distributed case, 200 for the middle case, and over 500 for the edge case. The estimate ideally requires the Shannon entropy to have reached convergence before the second half of the active cycles. This is particularly evident in the edge case, where the Shannon entropy has not converged even over 500 cycles for the 100k and 1mil runs, although MCNP5 recommends skipping significantly fewer cycles. Also, since MCNP5 only looks for the first value to fall within one standard deviation of average of the last half of cycles, a single outlier that falls within specified range can cause MCNP5 to estimate a cycle much earlier than it should. This unreliability in MCNP5's estimates is why visual inspection is being used to judge and confirm Shannon entropy convergence.

# **CHAPTER 4**

## **MONTE CARLO BENCHMARK PROBLEM FOR FULL-SIZE REACTOR CORE**

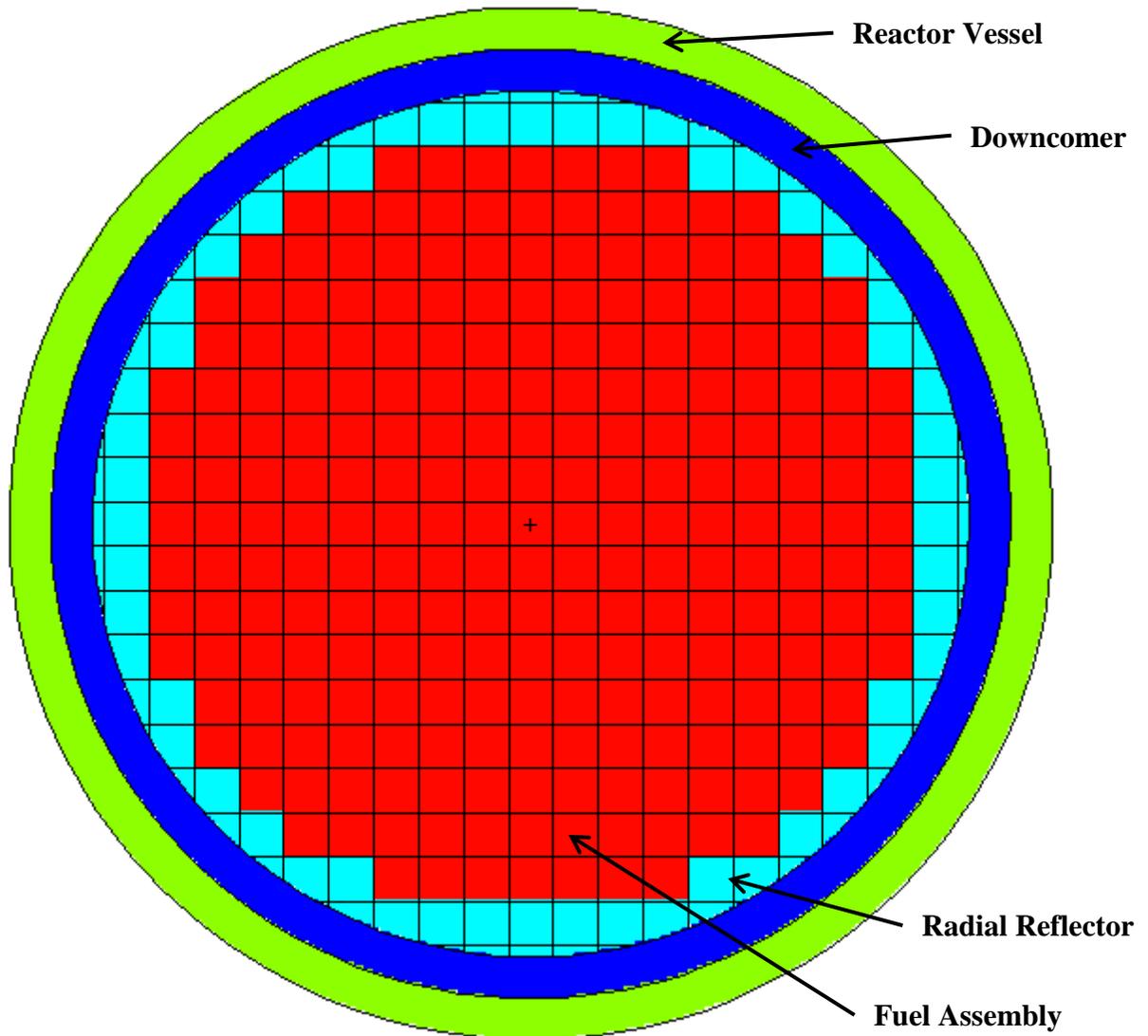
Chapter 3 applied the Shannon entropy approach to a simple model as a proof of principle to show the validity of this approach. In Chapter 4 we are looking to expand our approach to ensure that it can be applied to larger, more realistic models. This chapter will look to use the Shannon entropy from a run with fewer particles to predict when the source distribution will converge for a run with more particles.

### **4.1 Model Description**

The model chosen for this thesis was a full-size reactor core benchmark model for Monte Carlo code devised by Hoogenboom, Martin, and Petrovic [14]. The original goal of this benchmark problem was to monitor the performance of Monte Carlo calculations for a full-size reactor core and to stimulate improvements for Monte Carlo codes and their implementation [15]. The benchmark uses a PWR core as a reference, but simplifies the geometry and material composition. The goal of this benchmark was not to model any specific reactor core, but to create a core that requires Monte Carlo codes to perform realistic neutron history simulations.

The core consists of 241 square fuel assemblies with dimensions of 21.42 cm by 21.42 cm. The fuel assemblies are surrounded by a radial reflector (simplified region comprising baffle plates, former region, and core barrel) with an outer radius of 209cm, a downcomer with an outer radius of 229cm, and a reactor vessel with an outer radius of 249cm. The active fuel height for the model is 366cm, with the bottom half of the reactor using a cooler, higher density, cold water coolant, while the top half is using a lower

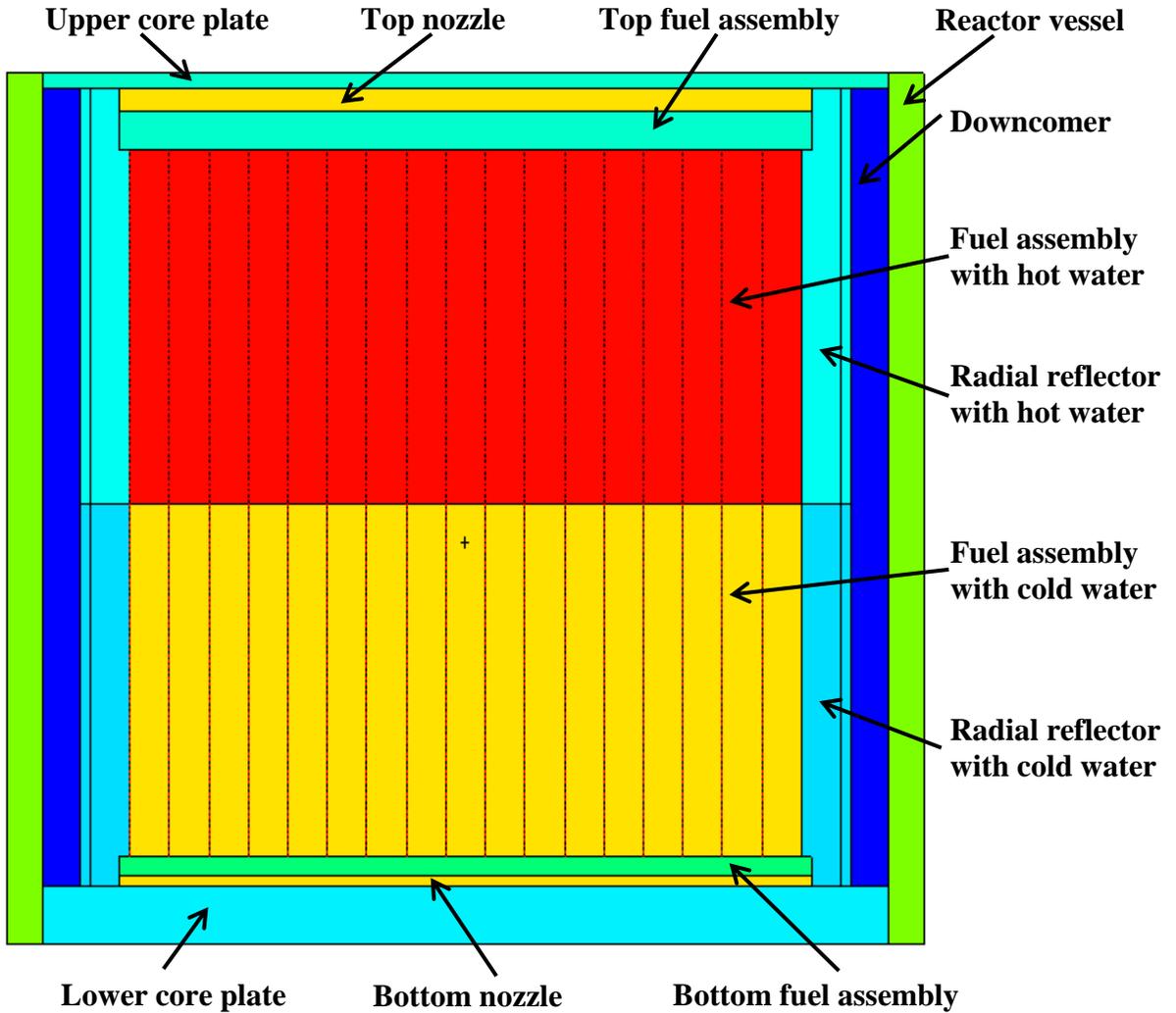
density, hot water coolant. Figure 4.1 shows a horizontal cross section of the reactor core, and Figure 4.2 shows a vertical cross section.



**Figure 4.1: Benchmark model (horizontal cross section)**

Each square fuel assembly consists of 17 by 17 square unit cells with dimensions of 1.26 by 1.26 cm<sup>2</sup>. Of these 289 unit cells, 264 are fuel pins, 24 are control rod guide tubes, and one centrally located cell is filled with an instrumentation tube. The control rod guide tubes have an inner and outer radius of 0.56cm and 0.62cm. The instrumentation tube has effectively the same dimensions. Each fuel pin has an outer

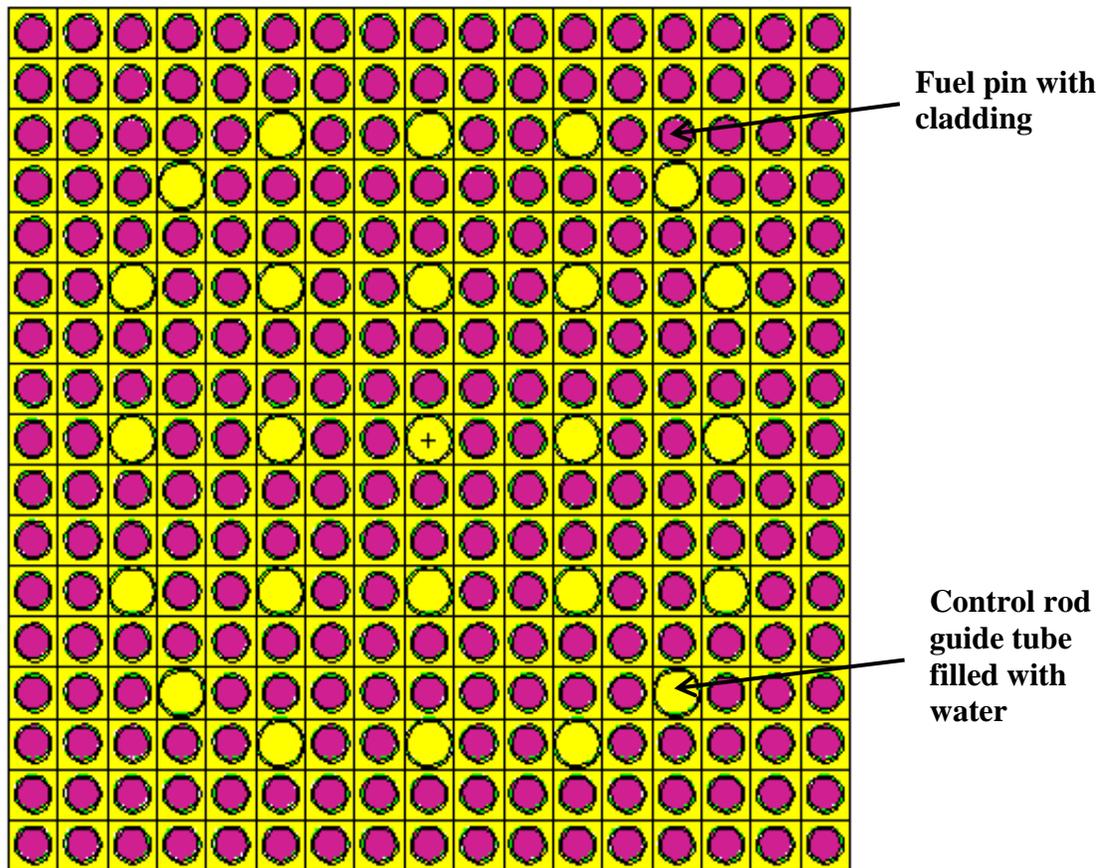
radius of 0.41cm and is surrounded by a cladding with outer radius of 0.475cm with no gap. Figure 4.3 shows a horizontal cross section of the model.



**Figure 4.2: Benchmark model (vertical cross section)**

The fuel regions were designed using fuel at a certain burnup stage, in this case, roughly 24,000 MWd/MTU. The fuel is a partially depleted uranium oxide fuel represented by 17 different actinides and 16 fission products. A more detailed fuel composition with specific isotopes and atomic densities of the uranium oxide fuel is shown in Table 4.1. The total atom and mass densities of the fuel were  $0.06822 \times 10^{24} \text{ cm}^{-3}$  and  $10.062 \text{ g/cm}^{-3}$ , respectively. The cladding and guide tubes consisted of natural

zirconium for simplicity. The coolant was borated water with both hot and cold halves for their respective elevations of the core. The boron concentration was chosen such that the reactor would be near critical. The core plates and nozzles used a stainless steel (SS 304) and water mixture. The downcomer simply used the cold borated water, and the reactor vessel was composed of a low-carbon steel (SA 508, Grade 2). The initial source distribution used in this problem was a cylindrical volume source comprising all fuel assemblies both radially and vertically while also using the default MCNP5 Watt fission energy spectrum. The dominance ratio for this model is 0.992 [16]. An example input showing other relevant information can be found in Appendix B.



**Figure 4.3: Fuel assembly (horizontal cross section)**

**Table 4.1: Uranium oxide fuel composition**

Isotope	Atom density x $10^{-24}$ cm <sup>-3</sup>	Isotope	Atom density x $10^{-24}$ cm <sup>-3</sup>
U-234	$4.9476 \times 10^{-6}$	Mb-95	$2.6497 \times 10^{-5}$
U-235	$4.8218 \times 10^{-4}$	Tc-99	$3.2772 \times 10^{-5}$
U-236	$9.0402 \times 10^{-5}$	Ru-101	$3.0742 \times 10^{-5}$
U-238	$2.1504 \times 10^{-2}$	Ru-103	$2.3505 \times 10^{-6}$
Np-237	$7.3733 \times 10^{-6}$	Ag-109	$2.0009 \times 10^{-6}$
Pu-238	$1.5148 \times 10^{-6}$	Xe-135	$1.0801 \times 10^{-8}$
Pu-239	$1.3955 \times 10^{-4}$	Cs-133	$3.4612 \times 10^{-5}$
Pu-240	$3.4405 \times 10^{-5}$	Nd-143	$2.6078 \times 10^{-5}$
Pu-241	$2.1439 \times 10^{-5}$	Nd-145	$1.9898 \times 10^{-5}$
Pu-242	$3.7422 \times 10^{-6}$	Sm-147	$1.6128 \times 10^{-6}$
Am-241	$4.5041 \times 10^{-7}$	Sm-149	$1.1627 \times 10^{-7}$
Am-242	$9.2301 \times 10^{-9}$	Sm-150	$7.1727 \times 10^{-6}$
Am-243	$4.7878 \times 10^{-7}$	Sm-151	$5.4947 \times 10^{-7}$
Cm-242	$1.0485 \times 10^{-7}$	Sm-152	$3.0221 \times 10^{-6}$
Cm-243	$1.4285 \times 10^{-7}$	Eu-153	$2.6209 \times 10^{-6}$
Cm-244	$8.8756 \times 10^{-8}$	Gd-155	$1.5369 \times 10^{-9}$
Cm-245	$3.5285 \times 10^{-9}$	O-16	$4.5737 \times 10^{-2}$

## 4.2 Preliminary Results from 1k and 10k Runs

Here shorter, preliminary runs are performed to estimate the number of cycles that need to be skipped for source convergence before running a longer run. As was shown in Chapter 3, 1k runs had too much noise to be reliable, and 10k runs also showed little promise. We are ideally looking at the 100k runs to estimate skipped cycles for a 1mil particles per cycle run. A run with 10 times fewer particles should complete in roughly one-tenth to one-fifth the total computational time of a longer run. Since we are only looking at entropy and its relation to source convergence, we can also perform these shorter runs without tallies to further decrease the time required for the runs to complete.

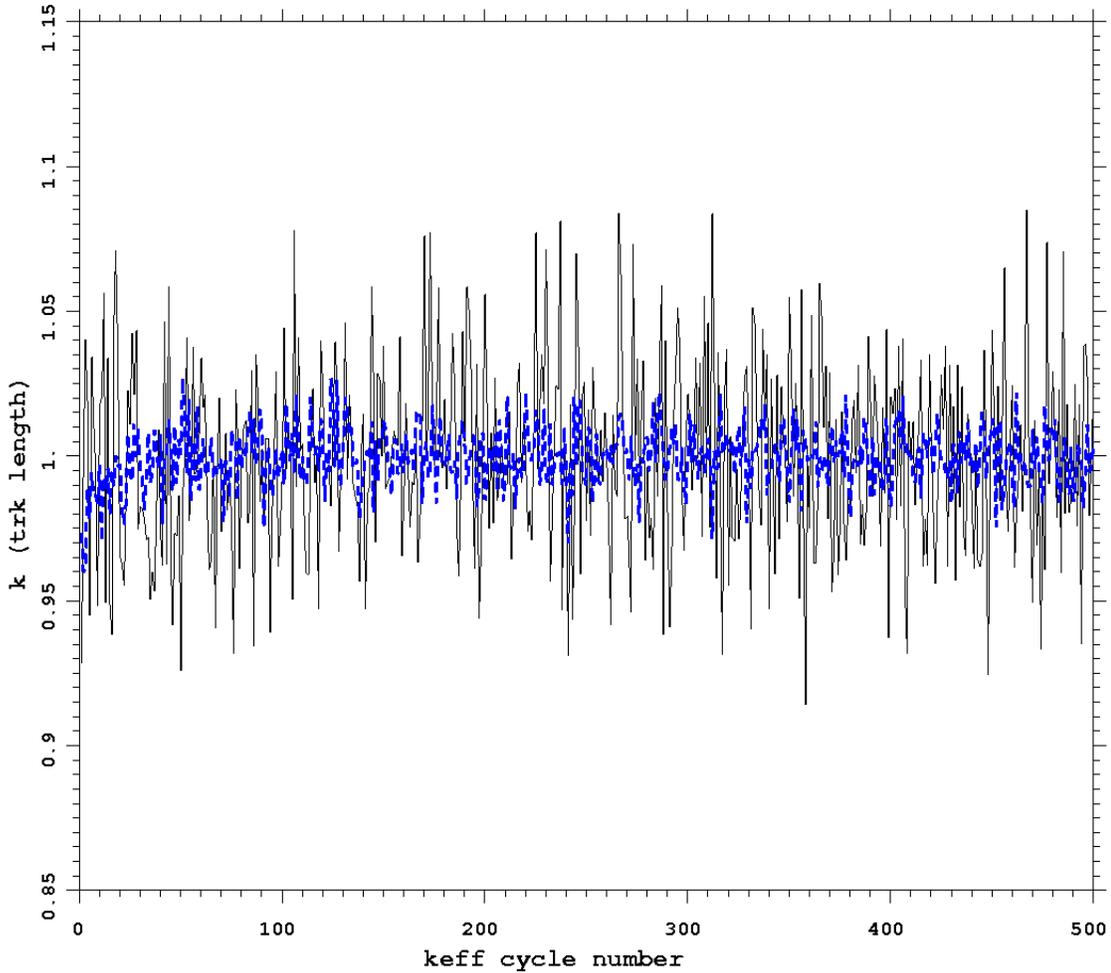
Preliminary full core runs using both 1k and 10k particles were completed over 500 active cycles (no inactive cycles) and without tallies. Each of these was performed on 8 CPUs on a single node. While the runs contain too few particles and may not be practical for our purposes, the information is still relevant for demonstrating the propagation of Shannon entropy as the number of particles is increased. The 1k run

completed in 13.29 minutes of CPU time, and a wall clock time of 2.85 minutes (00:02:51), while the 10k run completed in a CPU time of 116.25 minutes and a wall clock time of 15.72 minutes (00:15:43). We can observe that the MCNP5 total computational time of the 1k run completed in roughly 10% the time of the 10k run, or more precisely 11.43% of the total computation time.

Results for  $k_{\text{eff}}$  and Shannon entropy of the 1k and 10k runs are shown in Table 4.2. The  $k_{\text{eff}}$  values shown here are the collision/absorption/track-length average and the Shannon entropy value is the average fission-source entropy for the last half of cycles. These values were calculated by MCNP5 as well as their associated standard deviations shown in the table. A graph of track length estimated  $k_{\text{eff}}$  versus the cycle number is plotted in Figure 4.4 for both 1k and 10k particles per cycle. The 1k results are shown in black and the 10k results are shown in blue. The second half  $k_{\text{eff}}$  result (after  $k_{\text{eff}}$  has converged) for 1k has an uncertainty of 134 pcm and the 10k result has an uncertainty of 41 pcm. As expected, the uncertainty is reduced by a factor of roughly the square root of 10 from the 1k to 10k results.

**Table 4.2:  $k_{\text{eff}}$  and Shannon entropy results from 1k and 10k runs**

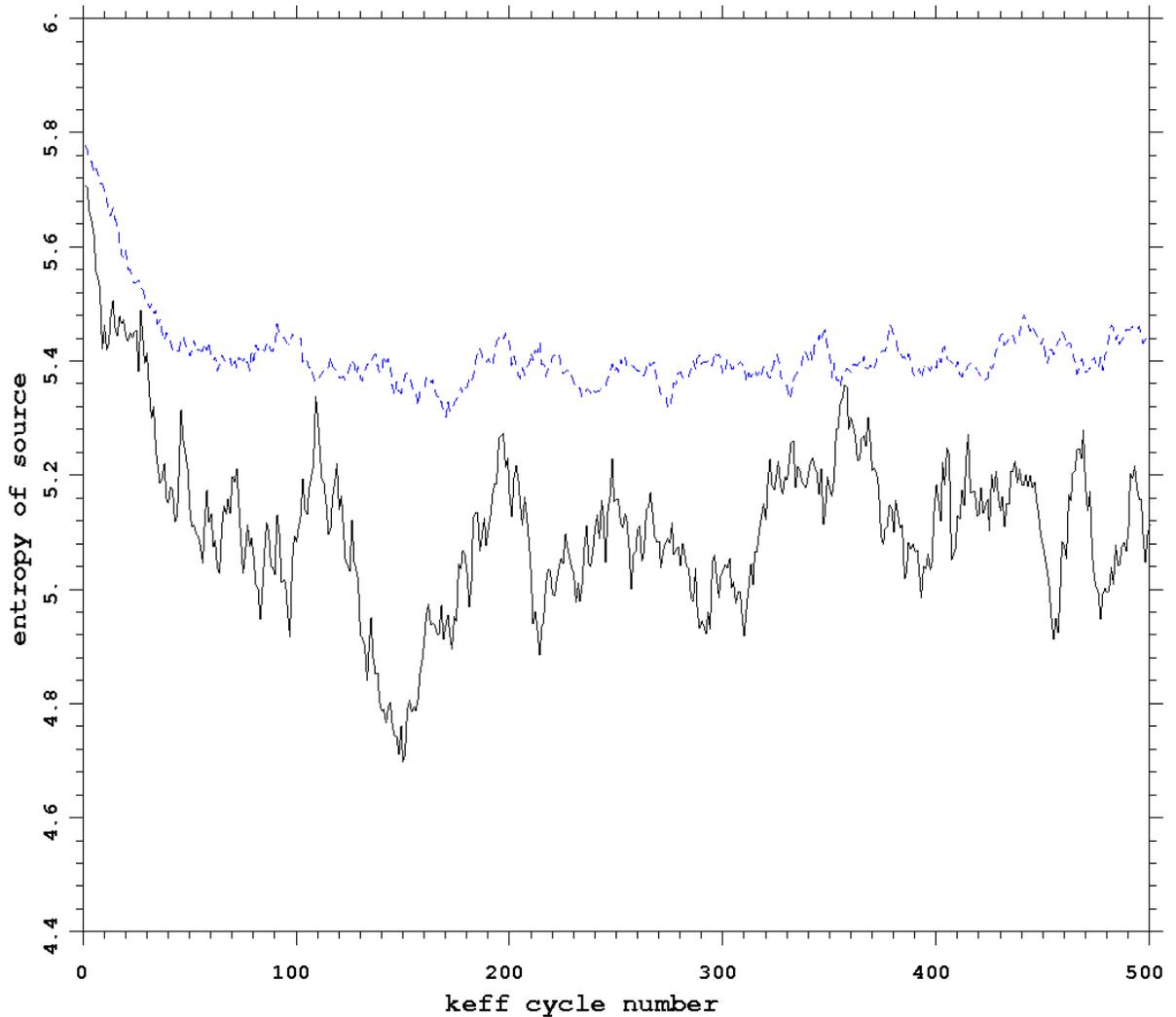
		<b>Value</b>	<b>St. Dev.</b>
<b>1k</b>	$k_{\text{eff}}$ first half	0.99829	0.00129
	$k_{\text{eff}}$ second half	0.99974	0.00134
	$k_{\text{eff}}$ final result	0.99901	0.00093
	Entropy	5.12E+00	9.30E-02
<b>10k</b>	$k_{\text{eff}}$ first half	0.99922	0.00045
	$k_{\text{eff}}$ second half	0.99951	0.00041
	$k_{\text{eff}}$ final result	0.99933	0.00030
	Entropy	5.40E+00	2.99E-02



**Figure 4.4:  $k_{\text{eff}}$  of 1k and 10k results**

As we can see in Figure 4.4, the 1k  $k_{\text{eff}}$  results vary between value of about 0.93 and 1.08, and the results appear to have too much noise to judge when  $k_{\text{eff}}$  has converged. The 10k  $k_{\text{eff}}$  results also have more noise than desired, but there is a pretty clear increase in  $k_{\text{eff}}$  over the first 50 cycles before reaching a value and oscillating between a value around 0.98 and 1.02 for the remainder of cycles. From this figure we can assume that for 10k particles  $k_{\text{eff}}$  converges within the first 50 cycles. Through visual inspection of  $k_{\text{eff}}$ , we can assume at least 50 inactive cycles are required to allow  $k_{\text{eff}}$  to converge, though one would still need to inspect the Shannon entropy figure. This is just a simple estimation and more extensive analysis could find a more accurate convergence point.

MCNP5 also states that “the minimum estimation standard deviation for the collision/absorption/track-length estimator occurs with 1 inactive cycle and 499 active cycles” for the 1k run and “with 12 inactive cycles and 488 active cycles” for the 10k run.



**Figure 4.5: Shannon entropy of 1k and 10k results**

Figure 4.5 shows the Shannon entropy of the 1k and 10k results, which provide a better estimation of when source convergence has been achieved than solely the  $k_{\text{eff}}$  results. Again, the 1k results appear to have too much noise to be useful. The 1k results do have an apparent decline in entropy over first 100 or 150 cycles, but there is far too much noise to give insight on when convergence has occurred. As was the case in

Chapter 3, the Shannon entropy should converge to roughly the same value regardless of the number of particles used, which is certainly not the case here for the 1k results in black when compared to the 10k results in blue. The average of the last half of cycles puts the entropy value at 5.12 for the 1k results, which is not even within one standard deviation of the value of 5.40, which was produced by the 10k results. MCNP5 output specifies cycle 36 as the first cycle having entropy within one standard deviation of the average of the last half of cycles.

The 10k results offer more promise as there is a general decline in Shannon entropy before oscillating around a steady-state value. For the 10k results, the entropy rapidly decreases over approximately the first 50 cycles, reaches a small plateau, and then gradually decreases until around cycle 200. While these results look more reliable, there is still too much noise to provide an estimate for the cycle number of when the Shannon entropy has converged as it is too difficult to tell whether entropy has converged at 50 cycles or at 200 cycles. For these results MCNP5 states that cycle 42 is the first cycle having entropy within one standard deviation of the average for the last half of cycles.

### **4.3 Results from 100k Run**

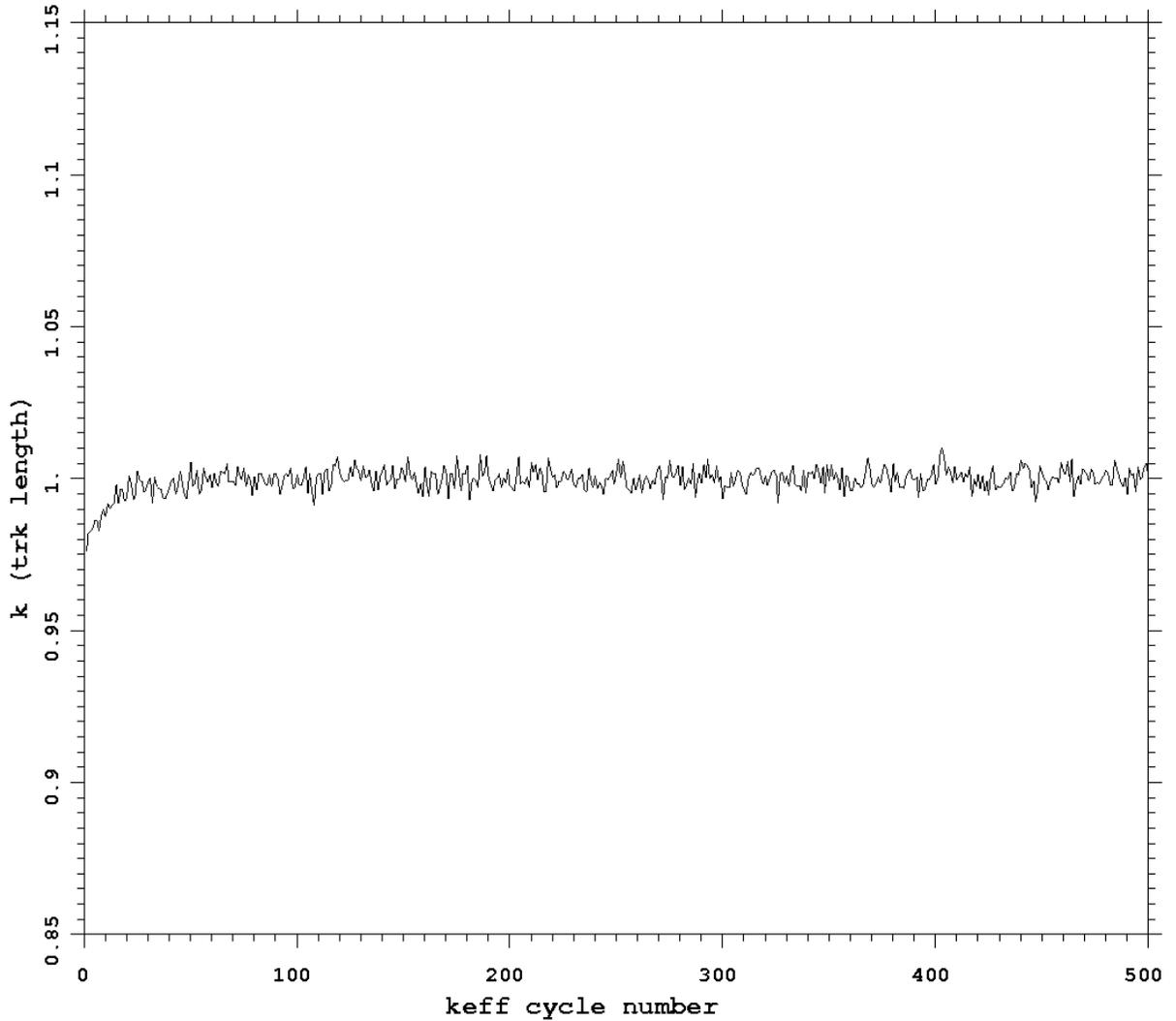
A full core run was then performed with 100k particles per cycle over 500 active cycles and without tallies. This particular run was completed in parallel on the Linux cluster using 16 CPUs over 2 nodes. The total CPU time was 1082.22 minutes according to the MCNP5 output file, and the wall clock time was 68.90 minutes (1:08:54). In the previous section, it was shown that the 1k and 10k results offered little promise; however, the goal here is to show that the 100k results can provide insight on source convergence for a longer run. More specifically, we intend to show that 100k particles can be used as an indicator of source convergence for a 1mil run with 10 times more particles, which would take approximately 10 times longer to run the same number of cycles, and significantly more time if detailed tallies are required.

Table 4.3 displays the average  $k_{\text{eff}}$  results and second half Shannon entropy average for the 100k results. For the  $k_{\text{eff}}$  results, we are primarily interested in the result for the second half, rather than the final result averaged over all active cycles. This is because there are no skipped cycles, and thus the value for  $k_{\text{eff}}$  has not yet converged before the beginning of the first half of active cycles. This is more noticeable for this 100k run than it is for the 1k and 10k runs. Interestingly, the standard deviation for the final result is actually higher than the standard deviation for the second half result. The second half  $k_{\text{eff}}$  standard deviation for 100k was 13 pcm, which is what would be expected when compared to the values of 134 and 41 pcm for the 1k and 10k runs, respectively.

**Table 4.3:  $k_{\text{eff}}$  and Shannon entropy results from 100k run**

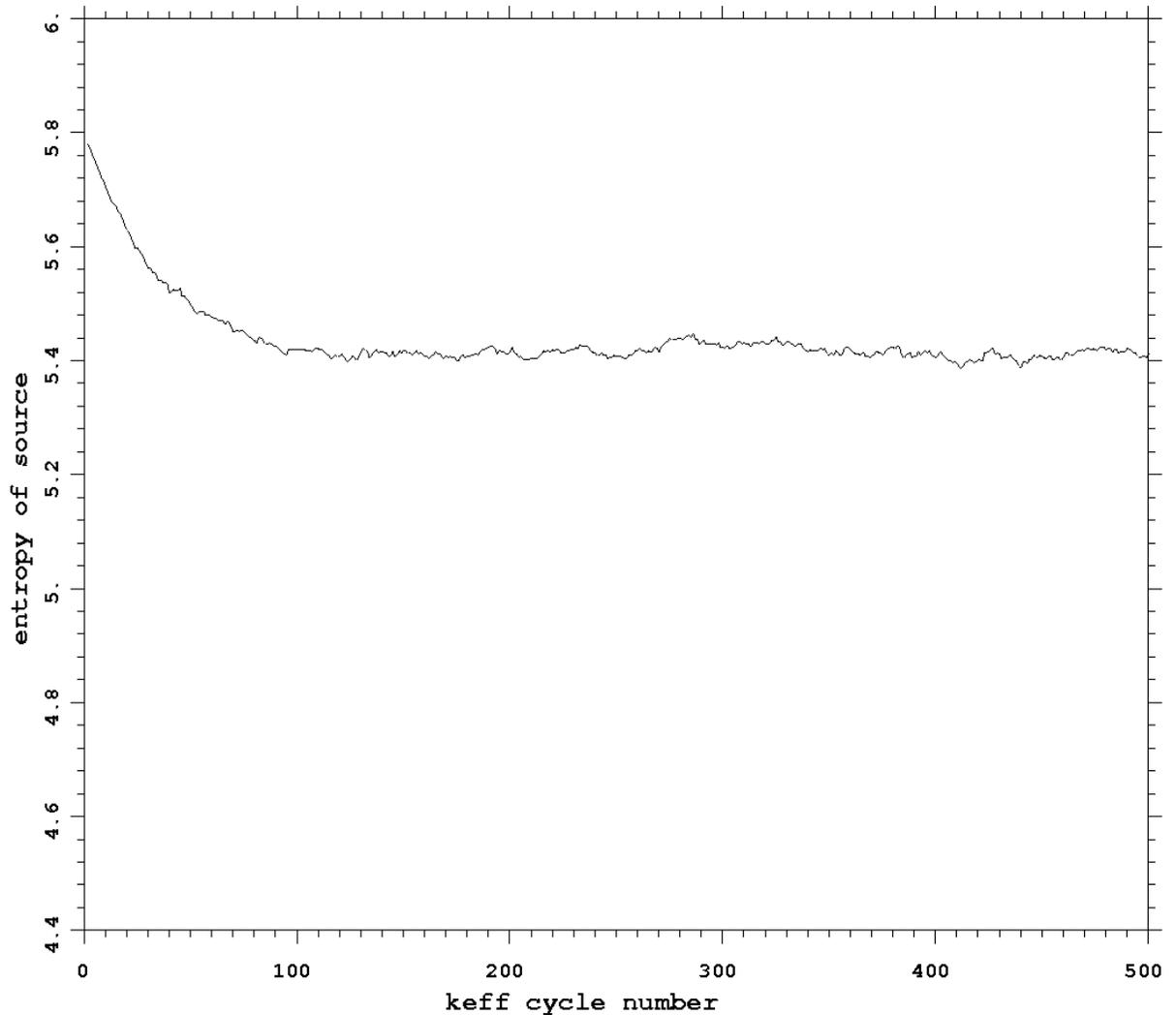
		<b>Value</b>	<b>St. Dev.</b>
<b>100k</b>	$k_{\text{eff}}$ first half	0.99875	0.00025
	$k_{\text{eff}}$ second half	1.00002	0.00013
	$k_{\text{eff}}$ final result	0.99938	0.00014
	Entropy	5.42E+00	1.19E-02

Figure 4.6 shows a graph of  $k_{\text{eff}}$  (track length estimation) versus cycle number for the 100k run. In this figure  $k_{\text{eff}}$  gradually increases before reaching a steady-state value after 40 or 50 cycles. This estimate is simply based on visual inspection, and MCNP5 output concurs with this estimation and states that “the minimum estimated standard deviation for the col/abs/tl keff estimator occurs with 40 inactive cycles and 460 active cycles”. From this figure and the output, one can assume that a minimum of 40 or 50 cycles must be skipped in order to have a converged  $k_{\text{eff}}$  for 100k particles per cycle.



**Figure 4.6:  $k_{\text{eff}}$  (track length) versus cycle of preliminary 100k run**

A more relevant indicator of source convergence is shown in Figure 4.7, a graph of the Shannon entropy over 500 cycles for the 100k run. As we can see from visual inspection of the graph, the Shannon entropy rapidly decreases over the first 100 cycles from an initial value around 5.80 to a steady-state value slightly above 5.40. When compared to the Shannon entropy of the 10k results in Figure 4.5, we can see that the two graphs have a similar slope and converge to a similar value near 5.40; however, the 100k results are much easier to determine a point where the entropy reaches steady-state.



**Figure 4.7: Shannon entropy of preliminary 100k run**

Through visual inspection we can estimate for this graph that the Shannon entropy converges at the earliest around 100 or 125 cycles. It is clear that the source does not converge before 100 cycles, and it seems that we can be confident that the source is converged by 150 cycles. MCNP5 output also specifies that cycle 88 is the first cycle having a entropy value within one standard deviation of the last half of cycles, though as stated previously in Chapter 2, this value will typically provide an underestimate when applied to a run with more particles and smaller noise. A prediction can be made using our visual inspection estimation for how many cycles must be skipped to ensure source

and  $k_{\text{eff}}$  convergence during skipped cycles. And although visual inspection is one of the better indicators we have available, it is still not as accurate as we would like. Because the earliest estimation implies skipping a minimum of 100 to 125 cycles, 150 cycles were skipped when performing the longer 1mil run.

#### 4.4 Applying Procedure to 1mil Run

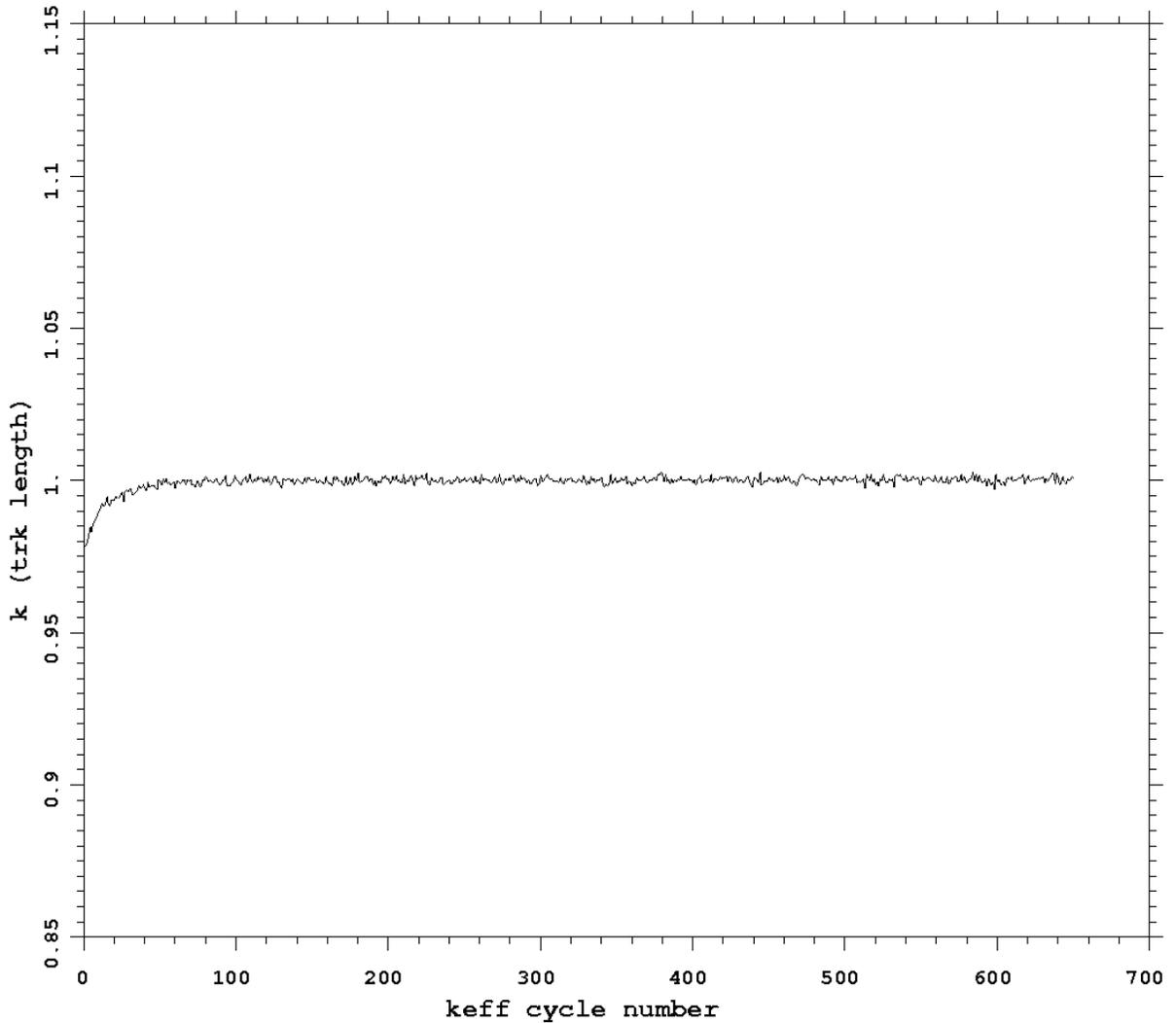
Using information from the 100k run, a 1mil run was performed with 150 inactive cycles and 500 active cycles for a total of 650 cycles. Our goal for this run was to show that the source and  $k_{\text{eff}}$  converge during the first 150 inactive cycles as we predicted from the 100k run. To show that this process works, and to reduce run time, this run was first completed without tallies. The 1mil run was performed on 64 CPUs over 8 nodes for a total CPU time of 21698.94 minutes and a wall clock change of 454.23 minutes (7:34:14). Each inactive cycle took on average 40.19 seconds and each active cycle took 42.1 seconds on average, which is not a significant difference for this tally-less run.

**Table 4.4:  $k_{\text{eff}}$  and Shannon entropy results from 1mil run**

		Value	St. Dev.
<b>1mil</b>	$k_{\text{eff}}$ first half	0.99999	0.00004
	$k_{\text{eff}}$ second half	1.00010	0.00004
	$k_{\text{eff}}$ final result	1.00005	0.00003
	Entropy	5.42E+00	3.98E-03

The  $k_{\text{eff}}$  and Shannon entropy results for the 1mil run are shown in Table 4.4. The first half  $k_{\text{eff}}$  result skips the 150 inactive cycles and uses the first 250 active cycles, while the second half uses the last 250 active cycles. The second half  $k_{\text{eff}}$  standard deviation is 4 pcm, roughly what one would expect when compared to previous runs, and the final standard deviation over all active cycles is 3 pcm. Also, the entropy value calculated in

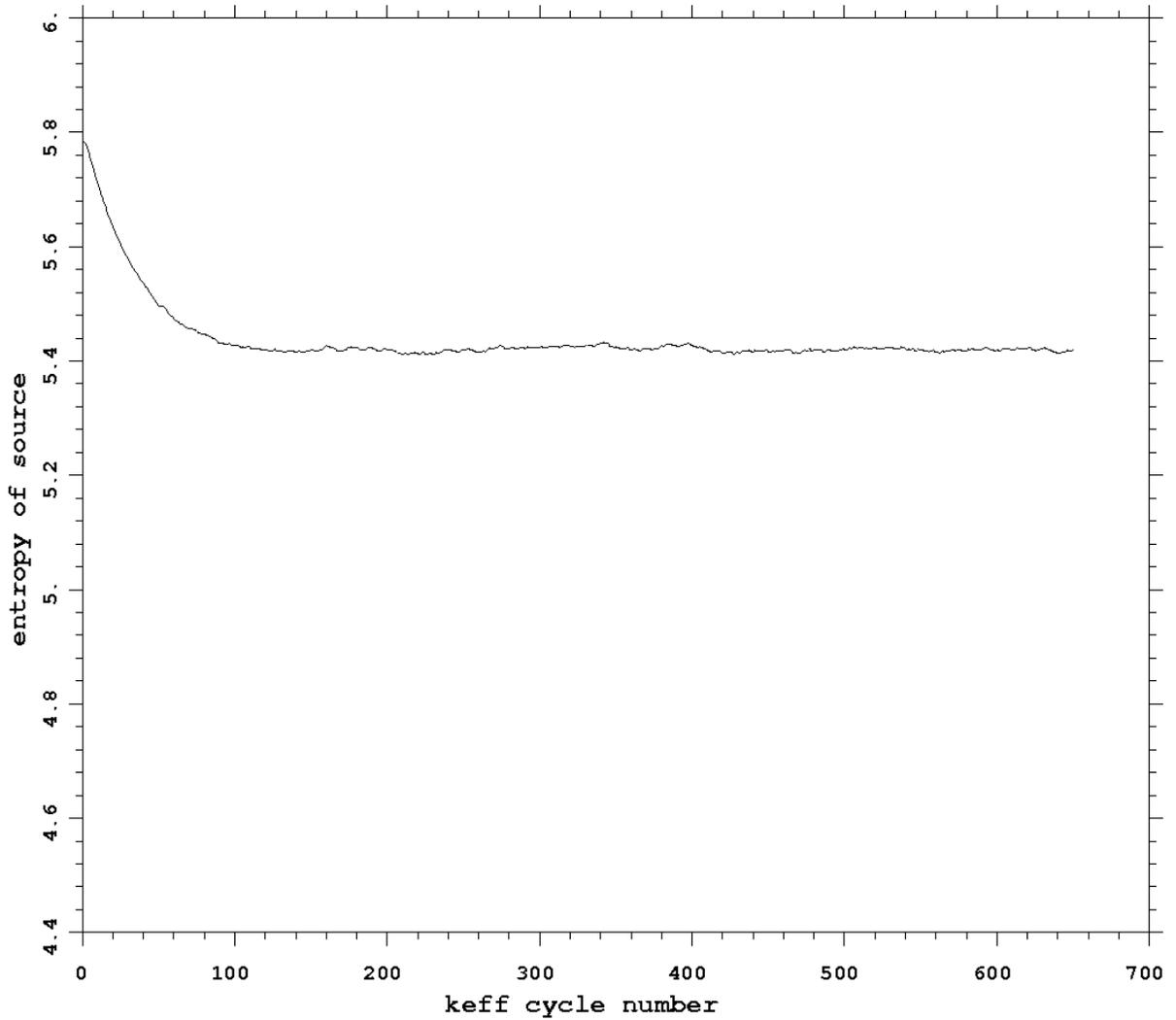
MCNP5 over the last half of cycles is  $5.42E+00$ , which is validating our approach as it is the same as the value calculated for the 100k run.



**Figure 4.8:  $k_{\text{eff}}$  (track length) versus cycle of 1mil run**

Figure 4.8 shows a graph of the track-length estimated  $k_{\text{eff}}$  per cycle for the 1mil run. This graph is similar to the 100k run as  $k_{\text{eff}}$  increases over the first 50 or so cycles before reaching a steady-state value. It is also very clear here that  $k_{\text{eff}}$  reaches steady-state well before the beginning of active cycles at cycle 150. MCNP5 output also states for this case that “the minimum estimated standard deviation for the

collision/absorption/track-length  $k_{\text{eff}}$  estimator occurs with 73 inactive cycles and 577 active cycles,” which is acceptable as 150 inactive cycles are used.



**Figure 4.9: Shannon entropy per cycle of 1mil run**

To confirm source convergence, Figure 4.9 shows the Shannon entropy per cycle of the 1mil run. As was shown in the 100k run, the Shannon entropy decreases over approximately the first 100 cycles before plateauing over the remainder of cycles. Again, through visual inspection it appears that the Shannon entropy does converge within the first 150 inactive cycles, confirming that our source distribution should be converged before the beginning of active cycles. MCNP5 output also states that the first cycle

having entropy within one standard deviation of the last half of cycles occurs at cycle 104, further indicating that the source distribution has converged prior to reaching 150 inactive cycles. In sum, both the  $k_{\text{eff}}$  and Shannon entropy graph confirm the source has converged within our estimate for the number of cycles to skip from the 100k run.

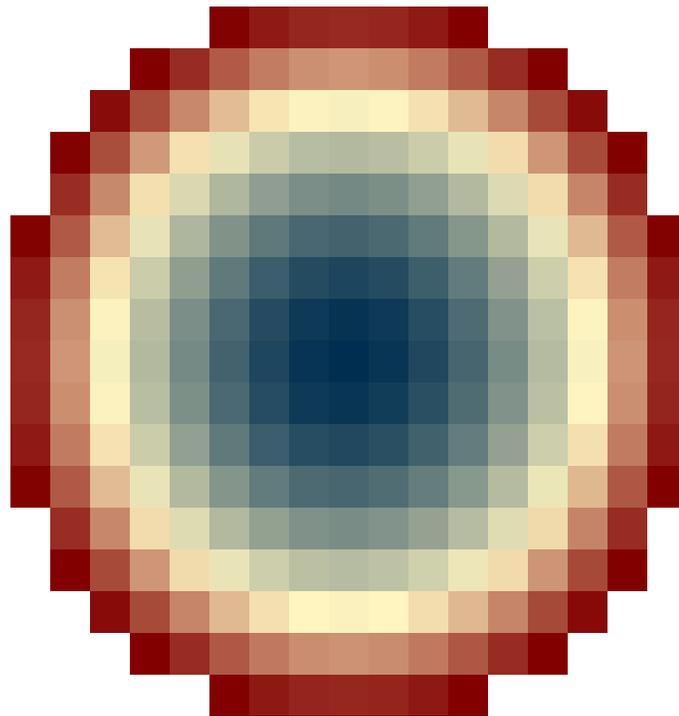
One point to note is that Shannon entropy does not change with regard to active or inactive cycles. For example, a run with 0 inactive cycles and 500 active cycles will produce the exact same Shannon entropy results as a run with 250 inactive and 250 active cycles. It is also helpful that, for this particular run, the source distribution needed to adjust before reaching stationary. An initial source distribution more similar to the actual source distribution could pose more challenges in determining the convergence point; however, for this case there is a distinct drop in Shannon entropy from its initial value, making it quite easy to determine source convergence.

#### **4.5 Confirming Source Convergence in the 1mil Run**

As Shannon entropy merely helps give insight on source convergence, it is also desirable to view flux distribution tallies and the normalized power distribution to ensure the source has in fact converged. While the true values may not be known, one can view symmetrical fuel assembly (FA) and pin positions to confirm equal power distribution across the entire fuel region. One of the reasons this model was chosen was its symmetry from left to right along the x-axis and from front to back along the y-axis. As coolant temperature and density is not consistent for the top and bottom halves, flux distributions will not be symmetrical along the z-axis.

A more realistic run with tallies was completed with 150 inactive cycles and 500 active cycles for one million particles per cycle once the tally-less run was completed to confirm Shannon entropy convergence during the first 150 cycles,. The run completed with a total CPU time of 138306.88 minutes and a wall clock time of 2164.05 minutes (36:04:03). Adding these large number of tallies increased the total run time nearly five-

fold. The two larger meshtallies added calculated prompt fission energy deposition over a specified fuel region. These tallies were unnormalized F4 type tallies and used MCNP5's tally multipliers to multiply the macroscopic fission cross section by the fission Q (energy in MeV per fission). The two large tallies added were energy deposition for each pin with 100 axial segments and energy deposition for each assembly, axially integrated from top to bottom. The tallies show the end of cycles results. The per pin tally is the primary culprit of the long run time as the mesh tally used 289 by 289 by 100 different segments for a total of 8,352,100 meshes, even though several of those were zero where fuel pins were not located and there was therefore no fission cross section. The per assembly tally only used 289 different meshes, with only 241 of those containing fuel regions.



**Figure 4.10: Heat map of energy deposition for each assembly**

The first way we can observe source convergence is the symmetry on the per FA level. Figure 4.10 shows a heat map of the energy deposition of each assembly. Of the

241 FAs, there is a single central FA, 56 FAs with 4-times symmetry, and 184 FAs with 8-times symmetry. Table 4.5 shows the energy depositions and Table 4.6 shows standard deviations of symmetric FAs. For these tables, the FA numbers are assembly coordinates, ranging from -8 to 8 in both the x- and y-axis, where the center assembly is located at 0,0. Each row of values represents a group of values at symmetric positions and should have similar values (i.e., consistent with the estimated statistical uncertainty) assuming the source is converged.

Observing each row separately, we see that the values and relative errors of each row match up fairly well. While the energy depositions are not perfectly symmetrical, there are no outliers and each value most closely resembles the respective symmetric assemblies rather than another neighboring assembly. For example, the values of the (2,2) FA group more closely match each other than the values of the adjacent (1,2) or (2,3) groups. While the values at symmetric positions agree with each other, they do not definitively match, as they do not necessarily fall within one standard deviation of each other.

**Table 4.5: Symmetry of energy deposition per fuel assembly, axially integrated over the entire assembly**

FA	Value														
<b>0,0</b>	3.676E-06														
FA	Value	FA	Value	FA	Value	FA	Value								
<b>0,1</b>	3.609E-06	<b>0,-1</b>	3.628E-06	<b>1,0</b>	3.612E-06	<b>-1,0</b>	3.622E-06								
<b>0,2</b>	3.417E-06	<b>0,-2</b>	3.446E-06	<b>2,0</b>	3.421E-06	<b>-2,0</b>	3.440E-06								
<b>0,3</b>	3.111E-06	<b>0,-3</b>	3.148E-06	<b>3,0</b>	3.119E-06	<b>-3,0</b>	3.149E-06								
<b>0,4</b>	2.713E-06	<b>0,-4</b>	2.752E-06	<b>4,0</b>	2.722E-06	<b>-4,0</b>	2.745E-06								
<b>0,5</b>	2.231E-06	<b>0,-5</b>	2.265E-06	<b>5,0</b>	2.243E-06	<b>-5,0</b>	2.265E-06								
<b>0,6</b>	1.690E-06	<b>0,-6</b>	1.715E-06	<b>6,0</b>	1.703E-06	<b>-6,0</b>	1.725E-06								
<b>0,7</b>	1.119E-06	<b>0,-7</b>	1.136E-06	<b>7,0</b>	1.128E-06	<b>-7,0</b>	1.138E-06								
<b>0,8</b>	5.457E-07	<b>0,-8</b>	5.543E-07	<b>8,0</b>	5.510E-07	<b>-8,0</b>	5.553E-07								
<b>1,1</b>	3.533E-06	<b>1,-1</b>	3.567E-06	<b>-1,1</b>	3.558E-06	<b>-1,-1</b>	3.565E-06								
<b>2,2</b>	3.163E-06	<b>2,-2</b>	3.191E-06	<b>-2,2</b>	3.203E-06	<b>-2,-2</b>	3.202E-06								
<b>3,3</b>	2.603E-06	<b>3,-3</b>	2.623E-06	<b>-3,3</b>	2.631E-06	<b>-3,-3</b>	2.652E-06								
<b>4,4</b>	1.904E-06	<b>4,-4</b>	1.923E-06	<b>-4,4</b>	1.914E-06	<b>-4,-4</b>	1.941E-06								
<b>5,5</b>	1.130E-06	<b>5,-5</b>	1.139E-06	<b>-5,5</b>	1.138E-06	<b>-5,-5</b>	1.157E-06								
<b>6,6</b>	3.893E-07	<b>6,-6</b>	3.932E-07	<b>-6,6</b>	3.957E-07	<b>-6,-6</b>	3.994E-07								
FA	Value	FA	Value	FA	Value	FA	Value	FA	Value	FA	Value	FA	Value	FA	Value
<b>1,2</b>	3.347E-06	<b>1,-2</b>	3.385E-06	<b>2,1</b>	3.342E-06	<b>2,-1</b>	3.369E-06	<b>-1,2</b>	3.370E-06	<b>-1,-2</b>	3.382E-06	<b>-2,1</b>	3.375E-06	<b>-2,-1</b>	3.381E-06
<b>1,3</b>	3.042E-06	<b>1,-3</b>	3.083E-06	<b>3,1</b>	3.048E-06	<b>3,-1</b>	3.060E-06	<b>-1,3</b>	3.065E-06	<b>-1,-3</b>	3.096E-06	<b>-3,1</b>	3.081E-06	<b>-3,-1</b>	3.089E-06
<b>1,4</b>	2.650E-06	<b>1,-4</b>	2.691E-06	<b>4,1</b>	2.662E-06	<b>4,-1</b>	2.661E-06	<b>-1,4</b>	2.671E-06	<b>-1,-4</b>	2.697E-06	<b>-4,1</b>	2.684E-06	<b>-4,-1</b>	2.701E-06
<b>1,5</b>	2.174E-06	<b>1,-5</b>	2.210E-06	<b>5,1</b>	2.190E-06	<b>5,-1</b>	2.194E-06	<b>-1,5</b>	2.189E-06	<b>-1,-5</b>	2.219E-06	<b>-5,1</b>	2.214E-06	<b>-5,-1</b>	2.218E-06
<b>1,6</b>	1.644E-06	<b>1,-6</b>	1.667E-06	<b>6,1</b>	1.662E-06	<b>6,-1</b>	1.666E-06	<b>-1,6</b>	1.654E-06	<b>-1,-6</b>	1.680E-06	<b>-6,1</b>	1.677E-06	<b>-6,-1</b>	1.678E-06
<b>1,7</b>	1.084E-06	<b>1,-7</b>	1.099E-06	<b>7,1</b>	1.095E-06	<b>7,-1</b>	1.100E-06	<b>-1,7</b>	1.089E-06	<b>-1,-7</b>	1.107E-06	<b>-7,1</b>	1.100E-06	<b>-7,-1</b>	1.105E-06
<b>1,8</b>	5.267E-07	<b>1,-8</b>	5.317E-07	<b>8,1</b>	5.307E-07	<b>8,-1</b>	5.334E-07	<b>-1,8</b>	5.298E-07	<b>-1,-8</b>	5.372E-07	<b>-8,1</b>	5.371E-07	<b>-8,-1</b>	5.379E-07
<b>2,3</b>	2.873E-06	<b>2,-3</b>	2.909E-06	<b>3,2</b>	2.881E-06	<b>3,-2</b>	2.896E-06	<b>-2,3</b>	2.899E-06	<b>-2,-3</b>	2.924E-06	<b>-3,2</b>	2.914E-06	<b>-3,-2</b>	2.916E-06
<b>2,4</b>	2.492E-06	<b>2,-4</b>	2.529E-06	<b>4,2</b>	2.505E-06	<b>4,-2</b>	2.505E-06	<b>-2,4</b>	2.509E-06	<b>-2,-4</b>	2.541E-06	<b>-4,2</b>	2.522E-06	<b>-4,-2</b>	2.537E-06
<b>2,5</b>	2.030E-06	<b>2,-5</b>	2.066E-06	<b>5,2</b>	2.045E-06	<b>5,-2</b>	2.047E-06	<b>-2,5</b>	2.047E-06	<b>-2,-5</b>	2.077E-06	<b>-5,2</b>	2.059E-06	<b>-5,-2</b>	2.070E-06
<b>2,6</b>	1.519E-06	<b>2,-6</b>	1.537E-06	<b>6,2</b>	1.531E-06	<b>6,-2</b>	1.539E-06	<b>-2,6</b>	1.527E-06	<b>-2,-6</b>	1.553E-06	<b>-6,2</b>	1.542E-06	<b>-6,-2</b>	1.548E-06
<b>2,7</b>	9.833E-07	<b>2,-7</b>	9.929E-07	<b>7,2</b>	9.924E-07	<b>7,-2</b>	9.992E-07	<b>-2,7</b>	9.853E-07	<b>-2,-7</b>	9.991E-07	<b>-7,2</b>	9.939E-07	<b>-7,-2</b>	9.989E-07
<b>2,8</b>	4.658E-07	<b>2,-8</b>	4.691E-07	<b>8,2</b>	4.668E-07	<b>8,-2</b>	4.712E-07	<b>-2,8</b>	4.675E-07	<b>-2,-8</b>	4.723E-07	<b>-8,2</b>	4.702E-07	<b>-8,-2</b>	4.751E-07
<b>3,4</b>	2.234E-06	<b>3,-4</b>	2.263E-06	<b>4,3</b>	2.241E-06	<b>4,-3</b>	2.259E-06	<b>-3,4</b>	2.261E-06	<b>-3,-4</b>	2.287E-06	<b>-4,3</b>	2.262E-06	<b>-4,-3</b>	2.290E-06
<b>3,5</b>	1.802E-06	<b>3,-5</b>	1.830E-06	<b>5,3</b>	1.808E-06	<b>5,-3</b>	1.820E-06	<b>-3,5</b>	1.821E-06	<b>-3,-5</b>	1.847E-06	<b>-5,3</b>	1.821E-06	<b>-5,-3</b>	1.838E-06
<b>3,6</b>	1.318E-06	<b>3,-6</b>	1.329E-06	<b>6,3</b>	1.318E-06	<b>6,-3</b>	1.326E-06	<b>-3,6</b>	1.322E-06	<b>-3,-6</b>	1.343E-06	<b>-6,3</b>	1.334E-06	<b>-6,-3</b>	1.337E-06
<b>3,7</b>	8.000E-07	<b>3,-7</b>	8.067E-07	<b>7,3</b>	8.020E-07	<b>7,-3</b>	8.100E-07	<b>-3,7</b>	8.063E-07	<b>-3,-7</b>	8.156E-07	<b>-7,3</b>	8.130E-07	<b>-7,-3</b>	8.110E-07
<b>3,8</b>	3.339E-07	<b>3,-8</b>	3.381E-07	<b>8,3</b>	3.380E-07	<b>8,-3</b>	3.422E-07	<b>-3,8</b>	3.383E-07	<b>-3,-8</b>	3.421E-07	<b>-8,3</b>	3.396E-07	<b>-8,-3</b>	3.406E-07
<b>4,5</b>	1.501E-06	<b>4,-5</b>	1.515E-06	<b>5,4</b>	1.496E-06	<b>5,-4</b>	1.510E-06	<b>-4,5</b>	1.506E-06	<b>-4,-5</b>	1.535E-06	<b>-5,4</b>	1.508E-06	<b>-5,-4</b>	1.530E-06
<b>4,6</b>	1.046E-06	<b>4,-6</b>	1.055E-06	<b>6,4</b>	1.043E-06	<b>6,-4</b>	1.049E-06	<b>-4,6</b>	1.048E-06	<b>-4,-6</b>	1.064E-06	<b>-6,4</b>	1.059E-06	<b>-6,-4</b>	1.066E-06
<b>4,7</b>	5.641E-07	<b>4,-7</b>	5.665E-07	<b>7,4</b>	5.648E-07	<b>7,-4</b>	5.687E-07	<b>-4,7</b>	5.639E-07	<b>-4,-7</b>	5.724E-07	<b>-7,4</b>	5.739E-07	<b>-7,-4</b>	5.751E-07
<b>5,6</b>	7.241E-07	<b>5,-6</b>	7.294E-07	<b>6,5</b>	7.237E-07	<b>6,-5</b>	7.309E-07	<b>-5,6</b>	7.325E-07	<b>-5,-6</b>	7.441E-07	<b>-6,5</b>	7.329E-07	<b>-6,-5</b>	7.445E-07
<b>5,7</b>	3.324E-07	<b>5,-7</b>	3.320E-07	<b>7,5</b>	3.340E-07	<b>7,-5</b>	3.344E-07	<b>-5,7</b>	3.358E-07	<b>-5,-7</b>	3.377E-07	<b>-7,5</b>	3.381E-07	<b>-7,-5</b>	3.415E-07

**Table 4.6: Relative error of energy deposition for symmetrically located fuel assemblies**

FA	St. Dev.														
0,0	6.218E-04														
FA	St. Dev.	FA	St. Dev.	FA	St. Dev.	FA	St. Dev.								
0,1	6.274E-04	0,-1	6.260E-04	1,0	6.273E-04	-1,0	6.265E-04								
0,2	6.453E-04	0,-2	6.424E-04	2,0	6.442E-04	-2,0	6.428E-04								
0,3	6.760E-04	0,-3	6.723E-04	3,0	6.753E-04	-3,0	6.722E-04								
0,4	7.244E-04	0,-4	7.191E-04	4,0	7.235E-04	-4,0	7.202E-04								
0,5	7.989E-04	0,-5	7.930E-04	5,0	7.968E-04	-5,0	7.930E-04								
0,6	9.190E-04	0,-6	9.119E-04	6,0	9.144E-04	-6,0	9.101E-04								
0,7	1.128E-03	0,-7	1.121E-03	7,0	1.123E-03	-7,0	1.119E-03								
0,8	1.617E-03	0,-8	1.607E-03	8,0	1.611E-03	-8,0	1.606E-03								
1,1	6.340E-04	1,-1	6.316E-04	-1,1	6.321E-04	1,-1	6.315E-04								
2,2	6.704E-04	2,-2	6.671E-04	-2,2	6.668E-04	2,-2	6.664E-04								
3,3	7.392E-04	3,-3	7.363E-04	-3,3	7.359E-04	3,-3	7.325E-04								
4,4	8.653E-04	4,-4	8.607E-04	-4,4	8.624E-04	4,-4	8.566E-04								
5,5	1.122E-03	5,-5	1.119E-03	-5,5	1.121E-03	5,-5	1.110E-03								
6,6	1.920E-03	6,-6	1.907E-03	-6,6	1.904E-03	6,-6	1.894E-03								
FA	St. Dev.	FA	St. Dev.	FA	St. Dev.	FA	St. Dev.	FA	St. Dev.	FA	St. Dev.	FA	St. Dev.	FA	St. Dev.
1,2	6.517E-04	1,-2	6.482E-04	2,1	6.517E-04	2,-1	6.497E-04	-1,2	6.496E-04	-1,-2	6.482E-04	-2,1	6.486E-04	-2,-1	6.484E-04
1,3	6.835E-04	1,-3	6.788E-04	3,1	6.825E-04	3,-1	6.816E-04	-1,3	6.814E-04	-1,-3	6.776E-04	-3,1	6.793E-04	-3,-1	6.784E-04
1,4	7.329E-04	1,-4	7.272E-04	4,1	7.312E-04	4,-1	7.313E-04	-1,4	7.300E-04	-1,-4	7.256E-04	-4,1	7.284E-04	-4,-1	7.263E-04
1,5	8.095E-04	1,-5	8.024E-04	5,1	8.064E-04	5,-1	8.061E-04	-1,5	8.068E-04	-1,-5	8.008E-04	-5,1	8.026E-04	-5,-1	8.014E-04
1,6	9.310E-04	1,-6	9.246E-04	6,1	9.266E-04	6,-1	9.249E-04	-1,6	9.282E-04	-1,-6	9.212E-04	-6,1	9.225E-04	-6,-1	9.214E-04
1,7	1.147E-03	1,-7	1.139E-03	7,1	1.141E-03	7,-1	1.139E-03	-1,7	1.145E-03	-1,-7	1.135E-03	-7,1	1.138E-03	-7,-1	1.136E-03
1,8	1.647E-03	1,-8	1.640E-03	8,1	1.641E-03	8,-1	1.636E-03	-1,8	1.643E-03	-1,-8	1.631E-03	-8,1	1.633E-03	-8,-1	1.630E-03
2,3	7.036E-04	2,-3	6.993E-04	3,2	7.026E-04	3,-2	7.015E-04	-2,3	7.002E-04	-2,-3	6.975E-04	-3,2	6.990E-04	-3,-2	6.982E-04
2,4	7.563E-04	2,-4	7.500E-04	4,2	7.539E-04	4,-2	7.539E-04	-2,4	7.530E-04	-2,-4	7.482E-04	-4,2	7.511E-04	-4,-2	7.493E-04
2,5	8.373E-04	2,-5	8.303E-04	5,2	8.346E-04	5,-2	8.340E-04	-2,5	8.339E-04	-2,-5	8.281E-04	-5,2	8.311E-04	-5,-2	8.291E-04
2,6	9.687E-04	2,-6	9.625E-04	6,2	9.645E-04	6,-2	9.630E-04	-2,6	9.660E-04	-2,-6	9.584E-04	-6,2	9.617E-04	-6,-2	9.595E-04
2,7	1.205E-03	2,-7	1.199E-03	7,2	1.199E-03	7,-2	1.196E-03	-2,7	1.203E-03	-2,-7	1.195E-03	-7,2	1.196E-03	-7,-2	1.194E-03
2,8	1.751E-03	2,-8	1.744E-03	8,2	1.748E-03	8,-2	1.739E-03	-2,8	1.748E-03	-2,-8	1.741E-03	-8,2	1.746E-03	-8,-2	1.736E-03
3,4	7.980E-04	3,-4	7.931E-04	4,3	7.971E-04	4,-3	7.947E-04	-3,4	7.937E-04	-3,-4	7.891E-04	-4,3	7.935E-04	-4,-3	7.894E-04
3,5	8.892E-04	3,-5	8.831E-04	5,3	8.873E-04	5,-3	8.859E-04	-3,5	8.852E-04	-3,-5	8.789E-04	-5,3	8.843E-04	-5,-3	8.807E-04
3,6	1.040E-03	3,-6	1.035E-03	6,3	1.040E-03	6,-3	1.037E-03	-3,6	1.038E-03	-3,-6	1.031E-03	-6,3	1.035E-03	-6,-3	1.033E-03
3,7	1.336E-03	3,-7	1.330E-03	7,3	1.334E-03	7,-3	1.327E-03	-3,7	1.331E-03	-3,-7	1.322E-03	-7,3	1.327E-03	-7,-3	1.327E-03
3,8	2.070E-03	3,-8	2.057E-03	8,3	2.059E-03	8,-3	2.045E-03	-3,8	2.057E-03	-3,-8	2.043E-03	-8,3	2.051E-03	-8,-3	2.053E-03
4,5	9.747E-04	4,-5	9.701E-04	5,4	9.762E-04	5,-4	9.718E-04	-4,5	9.729E-04	-4,-5	9.646E-04	-5,4	9.723E-04	-5,-4	9.656E-04
4,6	1.168E-03	4,-6	1.164E-03	6,4	1.170E-03	6,-4	1.165E-03	-4,6	1.165E-03	-4,-6	1.158E-03	-6,4	1.160E-03	-6,-4	1.157E-03
4,7	1.594E-03	4,-7	1.588E-03	7,4	1.594E-03	7,-4	1.585E-03	-4,7	1.589E-03	-4,-7	1.579E-03	-7,4	1.578E-03	-7,-4	1.576E-03
5,6	1.403E-03	5,-6	1.399E-03	6,5	1.403E-03	6,-5	1.396E-03	-5,6	1.396E-03	-5,-6	1.386E-03	-6,5	1.395E-03	-6,-5	1.384E-03
5,7	2.073E-03	5,-7	2.076E-03	7,5	2.070E-03	7,-5	2.067E-03	-5,7	2.068E-03	-5,-7	2.059E-03	-7,5	2.059E-03	-7,-5	2.048E-03

**Table 4.7: Calculated relative error of symmetric fuel assemblies**

FA Group	Calculated Rel. Err.	Average Rel. Err.	Ratio (C/A)	FA Group	Calculated Rel. Err.	Average Rel. Err.	Ratio (C/A)
<b>0,1</b>	2.399E-03	6.268E-04	3.8276338	<b>1,2</b>	4.801E-03	6.495E-04	7.3912765
<b>0,2</b>	4.128E-03	6.437E-04	6.4136149	<b>1,3</b>	6.439E-03	6.804E-04	9.46287176
<b>0,3</b>	6.294E-03	6.740E-04	9.3386045	<b>1,4</b>	7.032E-03	7.291E-04	9.64406515
<b>0,4</b>	6.870E-03	7.218E-04	9.5179237	<b>1,5</b>	7.532E-03	8.045E-04	9.36220731
<b>0,5</b>	7.472E-03	7.954E-04	9.3936906	<b>1,6</b>	7.480E-03	9.251E-04	8.08601018
<b>0,6</b>	8.955E-03	9.139E-04	9.7990925	<b>1,7</b>	7.003E-03	1.140E-03	6.14339656
<b>0,7</b>	7.667E-03	1.123E-03	6.8280611	<b>1,8</b>	7.616E-03	1.638E-03	4.65071314
<b>0,8</b>	7.822E-03	1.610E-03	4.857469	<b>2,3</b>	6.106E-03	7.002E-04	8.72055193
<b>1,1</b>	4.448E-03	6.323E-04	7.0352763	<b>2,4</b>	6.888E-03	7.520E-04	9.16035139
<b>2,2</b>	5.804E-03	6.677E-04	8.693036	<b>2,5</b>	7.651E-03	8.323E-04	9.19253511
<b>3,3</b>	7.793E-03	7.360E-04	10.588274	<b>2,6</b>	7.329E-03	9.631E-04	7.61047265
<b>4,4</b>	8.073E-03	8.612E-04	9.3732174	<b>2,7</b>	6.183E-03	1.199E-03	5.15891075
<b>5,5</b>	9.897E-03	1.118E-03	8.8507718	<b>2,8</b>	6.596E-03	1.744E-03	3.7820236
<b>6,6</b>	1.071E-02	1.906E-03	5.619002	<b>3,4</b>	8.558E-03	7.936E-04	10.7848176
				<b>3,5</b>	8.154E-03	8.843E-04	9.22044331
				<b>3,6</b>	6.935E-03	1.036E-03	6.69359361
				<b>3,7</b>	6.625E-03	1.329E-03	4.98428325
				<b>3,8</b>	7.921E-03	2.054E-03	3.85574478
				<b>4,5</b>	8.877E-03	9.711E-04	9.14179848
				<b>4,6</b>	8.136E-03	1.163E-03	6.99350002
				<b>4,7</b>	8.031E-03	1.585E-03	5.06642353
				<b>5,6</b>	1.079E-02	1.395E-03	7.7307036
				<b>5,7</b>	9.528E-03	2.065E-03	4.61391256

It is interesting to compare “true” relative errors obtained by observing a group of symmetric FAs and relative errors estimated and reported by MCNP5. Table 4.7 reports them as the “calculated relative errors” and “average relative errors” of each group of FAs. Each group is identified by the first FA listed in previous tables. These calculated relative errors are rather high compared to the average relative errors, as there are only either 4 or 8 assemblies per group. The calculated relative error is roughly 3 to 10 times larger than the average relative error as denoted by the ratio of calculated to average relative errors (C/A). And while this information is interesting in determining source convergence, it may more or less be a consequence of how large and loosely-coupled the system is. MCNP5’s manual also states estimated standard deviation for tallies could be

much smaller (by a factor of 5 or more) than the true standard deviation, which is in accordance to our findings [1]. Again, each symmetric FA has values at symmetric positions that closely resemble each other; however, as shown in Table 4.7, these values do not necessarily fall within one standard deviation of each other, most likely due to the size of the system.

One can also look at the energy deposition on the per pin level in the axial direction. These results do have a significant amount of noise as 241 assemblies with 264 pins each means 63,624 total fuel pins. For this run each pin was further split into 100 axial regions, creating 6,362,400 fuel pin meshes while only one million particles were run per cycle. On the fuel assembly level, there are enough particles to balance out any noise from only running one million particles per cycle; however, on the pin level when split into small axial regions, there is too much noise to confirm or deny source convergence on such a localized level. For the scope of this topic, it is evident that one can use a shorter run, such as 100k particles per cycle without tallies, to predict the number of cycles to skip for a longer run, such as 1mil or possible more particles per cycle with tallies. In any case, this research focuses on global source convergence and underestimate of its statistical uncertainty. Once the global convergence is achieved, the local convergence is expected to be easier to predict, if not to achieve.

## CHAPTER 5

### SUMMARY AND CONCLUSIONS

This thesis describes developing and testing of a simple technique for improving the efficiency and reliability of Monte Carlo criticality simulations by providing an estimate of the number of cycles that needs to be skipped. In terms of computer resources, this estimate uses a relatively inexpensive initial run and is obtained based on Shannon entropy. The method was tested using MCNP5, but it is generally applicable to any Monte Carlo criticality simulation. By applying this approach, one can hope to reduce the overall run time by predicting an adequate number of cycles to be skipped rather than too few cycles (that would require an additional run), or too many cycles (that would unnecessarily increase run time).

Table 5.1 reiterates the total CPU run times of each of the examples shown in Chapter 4. Some of these values could have been affected by the computer cluster load at the time, but it is clear that the 100k run takes significantly shorter time to run than the 1mil cases, in particular, the one with tallies. As was shown, the Shannon entropy of the 100k case converges in a similar fashion to the 1mil case, and by first running this short case, one could spend a relatively insignificant amount of time predicting the number of cycles to skip. This concept could further be extrapolated to runs with more particles per cycle and larger tallies to save even more time.

**Table 5.1: CPU run time**

<b>Particles per cycle</b>	<b>Total CPU time</b>	<b>Number of CPUs</b>
1k	13.29	8
10k	116.25	8
100k	1082.22	16
1mil	21698.94	64
1mil with tallies	138306.88	64

## 5.1 Future Work

The thesis demonstrated the proof-of-concept for an enhanced way of identifying source convergence; however, there is room for improvement in future work. Currently, we are only applying visual inspection of Shannon entropy convergence even though there are other methods. The current method relies heavily on a human aspect to estimate convergence, while other methods incorporate specific calculations, and visual inspection can then be used to confirm convergence. The existing MCNP5's entropy-based convergence criterion has its limitations, as it often produces an estimate lower than that which should be used for the number of cycles to skip, thus it may be non-conservative and at the same time requires significant computational time. Having available another method (such as the one proposed) would be particularly helpful when the source convergence is very slow and the related change in Shannon entropy very gradual, and it is therefore desirable to be able to run many neutron generations at an acceptable cost.

Another implementation for future work would be to apply an on-the-fly approach. Using an on-the-fly approach would reduce run time by running a lower number of cycles after convergence has been reached. Unfortunately, on-the-fly calculations can be ineffective if the Shannon entropy appears to have converged before it actually does. An on-the-fly approach may also not be too useful for the purpose of this concept. The shorter run already takes significantly less time to complete and determine how many cycles to skip compared to the longer run, and an on-the-fly approach would be better applied to a longer run, where once the convergence criteria has been met, the run can begin running active cycles.

Automating the entire process would be the biggest improvement one could make to improve on this concept. In order to do that, one would first need to use a different method to determine Shannon entropy convergence. For automating this process, one would first run a shorter run, with identical input aside from the number of cycles, particles per cycle, and tallies. After the predetermined number of cycles has been completed, the code would determine when and if source convergence has occurred. If it has not yet been determined, the code would run continue-runs with additional cycles until a point of convergence has been determined. Once the cycle number where Shannon entropy convergence has been determined, the automated code would set up and run a longer run with at least the specified number of inactive cycles determined from the shorter run. The difficulty in adding this technique comes from finding reliable convergence criteria and also determining how many particles per cycle is enough to ensure reliable Shannon entropy convergence. One reason visual inspection was used was the ease in showing convergence of shorter runs alongside longer runs. It was clear that the MCNP5 convergence criteria proved more useful when applied to runs with more particles, and this would most likely apply to other calculated methods, where more particles would allow for a better estimation of where convergence occurs.

# APPENDIX A

## EXAMPLE MCNP5 INPUT OF 2X2 PIN ARRAY

```
1k NO SKIPPED - 500 cycles
c
c =====+=====+=====+=====+=====+=====+=====+=====
c =====+=====+=====+=====+=====+=====+=====+=====
c CELL CARDS
c -----+-----+-----+-----pin-1-----+-----+-----+-----
c IFBA, higher enriched pin
11 11 -10.24 -11 52 -55 imp:n=1 $Fuel
16 13 -10.24 (-11 51 -52):(-11 55 -56) imp:n=1 $Natural UO_2
21 20 -0.416 11 -21 53 -54 imp:n=1 $IFBA
31 30 -0.001654 (21 -31 53 -54):(11 -31 51 -53):(11 -31 54 -56) imp:n=1 $Gap
41 40 -6.504 31 -41 51 -56 imp:n=1 $Clad
c
c -----+-----+-----+-----pin-2-----+-----+-----+-----
c Lower enrichment, no burnable absorber
12 12 -10.24 -12 52 -55 imp:n=1 $Fuel
17 13 -10.24 (-12 51 -52):(-12 55 -56) imp:n=1 $Natural UO_2
32 30 -0.001654 12 -32 51 -56 imp:n=1 $Gap
42 40 -6.504 32 -42 51 -56 imp:n=1 $Clad
c
c -----+-----+-----+-----pin-3-----+-----+-----+-----
c Lower enrichment, no burnable absorber
13 12 -10.24 -13 52 -55 imp:n=1 $Fuel
18 13 -10.24 (-13 51 -52):(-13 55 -56) imp:n=1 $Natural UO_2
33 30 -0.001654 13 -33 51 -56 imp:n=1 $Gap
43 40 -6.504 33 -43 51 -56 imp:n=1 $Clad
c
c -----+-----+-----+-----pin-4-----+-----+-----+-----
c IFBA, higher enriched pin
14 11 -10.24 -14 52 -55 imp:n=1 $Fuel
19 13 -10.24 (-14 51 -52):(-14 55 -56) imp:n=1 $Natural UO_2
24 20 -0.416 14 -24 53 -54 imp:n=1 $IFBA
34 30 -0.001654 (24 -34 53 -54):(14 -34 51 -53):(14 -34 54 -56) imp:n=1 $Gap
44 40 -6.504 34 -44 51 -56 imp:n=1 $Clad
c
c Water Moderator
50 50 -0.705 1 -3 4 -6 41 42 43 44 51 -56 imp:n=1
c
c Stainless Steel (SS 304) and Water Reflector
61 60 -4.50 1 -3 4 -6 50 -51 imp:n=1 $Bottom Reflector
62 60 -4.50 1 -3 4 -6 56 -57 imp:n=1 $Top Reflector
c
999 0 -1:3:-4:6:-50:57 imp:n=0

c =====+=====+=====+=====+=====+=====+=====+=====
c =====+=====+=====+=====+=====+=====+=====+=====
c SURFACE CARDS
c
c Outer box - periodic reflective
1 -3 px 0
2 px 1.26
3 -1 px 2.52
4 -6 py 0
5 py 1.26
```

```

6 -4 py 2.52
c
c Pins
11 c/z 0.63 0.63 0.3951    $Pin-1 Fuel
12 c/z 0.63 1.89 0.3951    $Pin-2 Fuel
13 c/z 1.89 0.63 0.3951    $Pin-3 Fuel
14 c/z 1.89 1.89 0.3951    $Pin-4 Fuel
21 c/z 0.63 0.63 0.3991    $Pin-1 IFBA
c No Pin-2 IFBA
c No Pin-3 IFBA
24 c/z 1.89 1.89 0.3991    $Pin-4 IFBA
31 c/z 0.63 0.63 0.4010    $Pin-1 Gap
32 c/z 0.63 1.89 0.4010    $Pin-2 Gap
33 c/z 1.89 0.63 0.4010    $Pin-3 Gap
34 c/z 1.89 1.89 0.4010    $Pin-4 Gap
41 c/z 0.63 0.63 0.4583    $Pin-1 Cladding
42 c/z 0.63 1.89 0.4583    $Pin-2 Cladding
43 c/z 1.89 0.63 0.4583    $Pin-3 Cladding
44 c/z 1.89 1.89 0.4583    $Pin-4 Cladding
c Z Dividers
50 pz  0.00    $bottom
51 pz  40.00   $bottom reflector
52 pz  55.24   $natural uranium UO_2
53 pz  70.48   $fully enriched zone
54 pz  375.28  $ifba zone
55 pz  390.52  $another fully enriched zone
56 pz  405.76  $natural uranium
57 pz  445.76  $top reflector

```

```

c =====+=====+=====+=====+=====+=====+=====+=====
c =====+=====+=====+=====+=====+=====+=====+=====

```

```

c DATA CARDS

```

```

c

```

```

c Materials

```

```

m11 92235.72c 0.05  & $Pin 1 and 4 Fuel IFBA ~4.95w/o
      92238.72c 0.95  &
      8016.72c 2
m12 92235.72c 0.021 & $Pin 2 and 3 Fuel Lower Enrichment
      92238.72c 0.979 &
      8016.72c 2
m13 92235.72c 0.0072 & $Natural Uranium
      92238.72c 0.9928 &
      8016.72c 2
m20 40090.71c 0.25725 & $Pin 1 and 4 IFBA      ZrB_2
      40091.71c 0.05610 &
      40092.71c 0.08575 &
      40094.71c 0.08690 &
      40096.71c 0.014   &
      5010.71c 0.6     &
      5011.71c 0.4
m30 2004.71c 1          $Helium Gap
m40 40090.71c -0.50539 & $Pins Cladding      Zircaloy-4 Cladding
      40091.71c -0.11021 &
      40092.71c -0.16846 &
      40094.71c -0.17072 &
      40096.71c -0.02750 &
      50112.71c -0.000141 &
      50114.71c -0.000096 &
      50115.71c -0.000049 &
      50116.71c -0.002108 &
      50117.71c -0.001114 &
      50118.71c -0.003512 &
      50119.71c -0.001246 &

```

```

50120.71c -0.004724 &
50122.71c -0.000671 &
50124.71c -0.000840 &
26054.71c -0.000122 &
26056.71c -0.001926 &
26057.71c -0.000046 &
26058.71c -0.000006 &
24050.71c -0.000043 &
24052.71c -0.000838 &
24053.71c -0.000095 &
24054.71c -0.000024 &
72174.71c -0.0000002 &
72176.71c -0.0000052 &
72177.71c -0.0000186 &
72178.71c -0.0000273 &
72179.71c -0.0000136 &
72180.71c -0.0000351
m50 8016.71c 1 & $Coolant
    1001.71c 2
m60 26054.71c -0.04292 & 50% H2O / 50% Stainless Steel 304
    26056.71c -0.678728 &
    26057.71c -0.01628 &
    26058.71c -0.002072 &
    24050.71c -0.007821 &
    24052.71c -0.1508208 &
    24053.71c -0.0171018 &
    24054.71c -0.004257 &
    28058.71c -0.0544616 &
    28060.71c -0.0209784 &
    28061.71c -0.000912 &
    28062.71c -0.0029072 &
    28064.71c -0.0007408 &
    8016.71c -0.8879 &
    1001.72c -0.1121
mt50 lwtr.16t $Light Water at 600K
c
c
c Criticality Source
kcode 1000 1.0 0 500
hsrc 1 0 2.52 1 0 2.52 20 36.576 365.76
c ksrc 0.63 0.63 222.88 0.63 1.89 222.88 1.89 0.63 222.88 1.89 1.89 222.88
sdef par=1 erg=d1 pos=d2 axs=0 0 1 rad=d4 ext=d5
sp1 -3 0.988 2.249
si2 L 0.63 0.63 40.00 0.63 1.89 40.00 1.89 0.63 40.00 1.89 1.89 40.00
sp2      1      1      1      1
si4 0 0.3951
sp4 -21 1
si5 0 365.76
sp5 0 1
c
c ----- Tallies -----
c Flux averaged over each pin and all pins
F4:n (11 16) (12 17) (13 18) (14 19) T
c Energy bins for F4 tally - thermal, epithermal, fast
E4 1E-6 0.1 20
c
FQ4 F D U S M C E T
FQ0 S C
c
c MESH TALLIES
FMESH214:n geom=cyl origin=0.63 0.63 40.00 &
    axs=0 0 1 vec=1 0 0 &
    imesh=0.3951 iints=1 &

```

```

jmesh=36.576 73.152 109.728 146.304 182.880 219.456 &
256.032 292.608 329.184 365.76 &
jints=10 10 10 10 10 10 10 10 10 10 &
kmesh=1 kints=1
FMESH224:n geom=cyl origin=0.63 1.89 40.00 &
axs=0 0 1 vec=1 0 0 &
imesh=0.3951 iints=1 &
jmesh=36.576 73.152 109.728 146.304 182.880 219.456 &
256.032 292.608 329.184 365.76 &
jints=10 10 10 10 10 10 10 10 10 10 &
kmesh=1 kints=1
FMESH234:n geom=cyl origin=1.89 0.63 40.00 &
axs=0 0 1 vec=1 0 0 &
imesh=0.3951 iints=1 &
jmesh=36.576 73.152 109.728 146.304 182.880 219.456 &
256.032 292.608 329.184 365.76 &
jints=10 10 10 10 10 10 10 10 10 10 &
kmesh=1 kints=1
FMESH244:n geom=cyl origin=1.89 1.89 40.00 &
axs=0 0 1 vec=1 0 0 &
imesh=0.3951 iints=1 &
jmesh=36.576 73.152 109.728 146.304 182.880 219.456 &
256.032 292.608 329.184 365.76 &
jints=10 10 10 10 10 10 10 10 10 10 &
kmesh=1 kints=1
FMESH314:n geom=cyl origin=0.63 0.63 40.00 &
axs=0 0 1 vec=1 0 0 &
imesh=0.3951 iints=1 &
jmesh=36.576 73.152 109.728 146.304 182.880 219.456 &
256.032 292.608 329.184 365.76 &
jints=10 10 10 10 10 10 10 10 10 10 &
kmesh=1 kints=1
FM314 (-1 11 -7)
FMESH324:n geom=cyl origin=0.63 1.89 40.00 &
axs=0 0 1 vec=1 0 0 &
imesh=0.3951 iints=1 &
jmesh=36.576 73.152 109.728 146.304 182.880 219.456 &
256.032 292.608 329.184 365.76 &
jints=10 10 10 10 10 10 10 10 10 10 &
kmesh=1 kints=1
FM324 (-1 12 -7)
FMESH334:n geom=cyl origin=1.89 0.63 40.00 &
axs=0 0 1 vec=1 0 0 &
imesh=0.3951 iints=1 &
jmesh=36.576 73.152 109.728 146.304 182.880 219.456 &
256.032 292.608 329.184 365.76 &
jints=10 10 10 10 10 10 10 10 10 10 &
kmesh=1 kints=1
FM334 (-1 12 -7)
FMESH344:n geom=cyl origin=1.89 1.89 40.00 &
axs=0 0 1 vec=1 0 0 &
imesh=0.3951 iints=1 &
jmesh=36.576 73.152 109.728 146.304 182.880 219.456 &
256.032 292.608 329.184 365.76 &
jints=10 10 10 10 10 10 10 10 10 10 &
kmesh=1 kints=1
FM344 (-1 11 -7)
print

```

## APPENDIX B

### EXAMPLE OF BENCHMARK MODEL FULL CORE MCNP5 INPUT

```
PWR core for performance benchmark
c
c as specified in:
c JE Hogenboom, WR Martin, B Petrovic, "Monte Carlo performance Benchmark
c for Detailed Power Density Calculation in a Full Size Reactor Core",
c OECD/NEA document, October, 2009.
c
c
c 1 digit = universe numbers
c 2 digits = surface numbers
c 3 digits = cell numbers
c 4 digits = material numbers
c
c ===== cell info - start
c =====
c
101 1000 0.06822 -11 u=1 imp:n=1 $ fuel pin
102 2000 -5.77 +11 -12 u=1 imp:n=1 $ cladding
103 3100 -0.74 +12 u=1 imp:n=1 $ borated coolant; COLD
c
201 3100 -0.74 -13 u=2 imp:n=1 $ inner guide tube with borated
202 2000 -5.77 +13 -14 u=2 imp:n=1 $ guide tube
203 3100 -0.74 +14 u=2 imp:n=1 $ borated coolant; COLD
c
301 1000 0.06822 -11 u=3 imp:n=1 $ fuel pin
302 2000 -5.77 +11 -12 u=3 imp:n=1 $ cladding
303 4100 -0.66 +12 u=3 imp:n=1 $ borated coolant; HOT
c
401 4100 -0.66 -13 u=4 imp:n=1 $ inner guide tube with borated
402 2000 -5.77 +13 -14 u=4 imp:n=1 $ guide tube
403 4100 -0.66 +14 u=4 imp:n=1 $ borated coolant; HOT
c
500 0 -20 lat=1 u=5 imp:n=1 $ FA 17x17 pin cells, COLD
fill=-8:8 -8:8 0:0
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 2 1 1 2 1 1 2 1 1 1 1
1 1 1 2 1 1 1 1 1 1 1 1 1 2 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 2 1 1 2 1 1 2 1 1 2 1 1 2 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 2 1 1 2 1 1 2 1 1 2 1 1 2 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 2 1 1 2 1 1 2 1 1 2 1 1 2 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 2 1 1 1 1 1 1 1 1 1 2 1 1
1 1 1 1 1 2 1 1 2 1 1 2 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
c
600 0 -20 lat=1 u=6 imp:n=1 $ FA 17x17 pin cells, HOT
fill=-8:8 -8:8 0:0
```

```

3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
3 3 3 3 3 4 3 3 4 3 3 4 3 3 3 3
3 3 3 4 3 3 3 3 3 3 3 3 3 4 3 3
3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
3 3 4 3 3 4 3 3 4 3 3 4 3 3 4 3 3
3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
3 3 4 3 3 4 3 3 4 3 3 4 3 3 4 3 3
3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
3 3 4 3 3 4 3 3 4 3 3 4 3 3 4 3 3
3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
3 3 3 4 3 3 3 3 3 3 3 3 4 3 3 3
3 3 3 3 3 4 3 3 4 3 3 4 3 3 3 3
3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3

```

c

```

700 3050 -4.32 -30 lat=1 u=7 imp:n=1 $ core - 21x21 FAs, COLD
fill=-10:10 -10:10 0:0
7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
7 7 7 7 7 7 7 5 5 5 5 5 5 5 7 7 7 7 7 7 7
7 7 7 7 7 5 5 5 5 5 5 5 5 5 5 5 5 5 7 7 7
7 7 7 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 7 7 7
7 7 7 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 7 7 7
7 7 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 7 7 7
7 7 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 7 7 7
7 7 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 7 7 7
7 7 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 7 7 7
7 7 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 7 7 7
7 7 7 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 7 7 7
7 7 7 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 7 7 7
7 7 7 7 7 7 7 5 5 5 5 5 5 5 5 5 5 5 7 7 7
7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7

```

c

```

800 4050 -4.28 -30 lat=1 u=8 imp:n=1 $ core - 21x21 FAs, HOT
fill=-10:10 -10:10 0:0
8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
8 8 8 8 8 8 6 6 6 6 6 6 6 6 8 8 8 8 8 8 8
8 8 8 8 8 6 6 6 6 6 6 6 6 6 6 6 8 8 8 8 8
8 8 8 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 8 8 8
8 8 8 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 8 8 8
8 8 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 8 8 8
8 8 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 8 8 8
8 8 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 8 8 8
8 8 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 8 8 8
8 8 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 8 8 8
8 8 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 8 8 8
8 8 8 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 8 8 8
8 8 8 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 8 8 8
8 8 8 8 8 6 6 6 6 6 6 6 6 6 6 6 6 6 8 8 8
8 8 8 8 8 6 6 6 6 6 6 6 6 6 6 6 6 6 8 8 8
8 8 8 8 8 8 6 6 6 6 6 6 6 6 6 6 6 6 8 8 8
8 8 8 8 8 8 8 8 6 6 6 6 6 6 6 6 8 8 8 8 8
8 8 8 8 8 8 8 8 6 6 6 6 6 6 6 6 8 8 8 8 8

```

```

      8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
      8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
c
410 4050 -4.28 -41          imp:n=1    $ top core plate
420 4085 -1.746 -42        imp:n=1    $ top nozzle
430 4070 -1.762 -43        imp:n=1    $ top FA
510 0          -51 42 43 fill=8 imp:n=1    $ core + reflector - top
520 0          -52 61 62 fill=7 imp:n=1    $ core + reflector - bottom
610 3060 -3.044 -61        imp:n=1    $ bottom FA
620 3075 -2.53 -62        imp:n=1    $ bottom nozzle
630 3010 -7.184 -63        imp:n=1    $ bottom core plate
c
910 3100 -0.74 71 -72      imp:n=1    $ downcomer; cold water
920 5000 -7.9 72 63 41 -81 imp:n=1    $ reactor vessel
999 0          81          imp:n=0    $ outside
c ===== cell info - end
=====

c ===== surface info - start
=====
11  cz  0.41          $ pellet radius (no gap)
12  cz  0.475        $ cladding outer radius
c
13  cz  0.56          $ guide tube inner radius
14  cz  0.62          $ guide tube outer radius
c
20  rpp  -.63 .63 -.63 .63 0 0 $ unit cell
30  rpp -10.71 10.71 -10.71 10.71 0 0 $ assembly
c
41  rcc 0. 0. 215. 0. 0. 8. 229. $ top core plate
42  rcc 0. 0. 203. 0. 0. 12. 187.6 $ top nozzle
43  rcc 0. 0. 183. 0. 0. 20. 187.6 $ top FA
51  rcc 0. 0. 0. 0. 0. 215. 209. $ core + reflector - top
52  rcc 0. 0. -199. 0. 0. 199. 209. $ core + reflector - bottom
61  rcc 0. 0. -193. 0. 0. 10. 187.6 $ bottom FA
62  rcc 0. 0. -199. 0. 0. 6. 187.6 $ bottom nozzle
63  rcc 0. 0. -229. 0. 0. 30. 229. $ bottom core plate
c
71  rcc 0. 0. -199. 0. 0. 414. 209. $ downcomer - inside
72  rcc 0. 0. -199. 0. 0. 414. 229. $ downcomer - outside
81  rcc 0. 0. -229. 0. 0. 452. 249. $ reactor vessel - outside
c ===== surface info - end
=====

c
=====
kcode 1000 1. 0 500
hsrc 4 -196.9 196.9 4 -196.9 196.9 4 -183 183
prtmp j 200 0 1 999999
sdef pos=0. 0. 0. axs=0. 0. 1. rad=d1 ext=d2 erg=d3 $ cylindrical vol.
source
  sil 0. 177.25 $ within all fuel assemblies
  spl -21 1
  si2 -183. 183.
  sp2 0. 1.
  sp3 -3
print -10 -30 -40 -50 -70 -72 -98 -102 -110 -120 -128 -130 -140 -160 -175 -178
c
m1000 $====> fuel
92234 4.9476E-06 92235 4.8218E-04 92236 9.0402E-05 92238 2.1504E-02
93237 7.3733E-06 94238 1.5148E-06 94239 1.3955E-04 94240 3.4405E-05
94241 2.1439E-05 94242 3.7422E-06 95241 4.5041E-07 95242 9.2301E-09
95243 4.7878E-07 96242 1.0485E-07 96243 1.4268E-09 96244 8.8756E-08

```

96245	3.5285E-09	42095	2.6497E-05	43099	3.2772E-05	44101	3.0742E-05
44103	2.3505E-06	47109	2.0009E-06	54135	1.0801E-08	55133	3.4612E-05
60143	2.6078E-05	60145	1.9898E-05	62147	1.6128E-06	62149	1.1627E-07
62150	7.1727E-06	62151	5.4947E-07	62152	3.0221E-06	63153	2.6209E-06
64155	1.5369E-09	8016	0.045737				
m2000 \$====> clad - pure Zr							
40090	0.5145	40091	0.1122	40092	0.1715	40094	0.1738
40096	0.0280						
c							
m3100 \$====> 100% COLD borated water							
1001	2.	8016	1.	5010	6.490e-4	5011	2.689e-3
mt3100 lwtr.60t							
m3075 \$====> 75% COLD borated water + 25% SS304							
1001	-0.0245014	8016	-0.1944274	5010	-7.89917e-5	5011	-3.59854e-4
26054	-0.0304114	26056	-0.4950122	26057	-0.0116345	26058	-0.0015782
14028	-0.0071714	14029	-0.0003774	14030	-0.0002576	24050	-0.0061909
24052	-0.1241425	24053	-0.0143485	24054	-0.0036383	25055	-0.0156126
28058	-0.0472112	28060	-0.0188120	28061	-0.0008311	28062	-0.0026944
28064	-0.0007082						
mt3075 lwtr.60t							
m3060 \$====> 60% COLD borated water + 40% Zr							
1001	-0.0162913	8016	-0.1292776	5010	-5.25228e-5	5011	-2.39272e-4
40090	-0.4331009	40091	-0.0955004	40092	-0.1475791	40094	-0.1528149
40096	-0.0251441						
mt3060 lwtr.60t							
m3050 \$====> 50% COLD borated water + 50% SS304							
1001	-0.0095661	8016	-0.0759107	5010	-3.08409e-5	5011	-1.40499e-4
26054	-0.0356208	26056	-0.5798060	26057	-0.0136275	26058	-0.0018485
14028	-0.0083998	14029	-0.0004420	14030	-0.0003017	24050	-0.0072514
24052	-0.1454076	24053	-0.0168063	24054	-0.0042615	25055	-0.0182870
28058	-0.0552984	28060	-0.0220344	28061	-0.0009735	28062	-0.0031559
28064	-0.0008295						
mt3050 lwtr.60t							
m3010 \$====> 10% COLD borated water + 90% SS304							
1001	-0.0011505	8016	-0.0091296	5010	-3.70915e-6	5011	-1.68974e-5
26054	-0.0385561	26056	-0.6275851	26057	-0.0147505	26058	-0.0020009
14028	-0.0090920	14029	-0.0004784	14030	-0.0003266	24050	-0.0078489
24052	-0.1573900	24053	-0.0181913	24054	-0.0046127	25055	-0.0197940
28058	-0.0598552	28060	-0.0238502	28061	-0.0010537	28062	-0.0034159
28064	-0.0008979						
mt3010 lwtr.60t							
c							
m4100 \$====> 100% HOT borated water							
1001	2.	8016	1.	5010	6.490e-4	5011	2.689e-3
mt4100 lwtr.60t							
m4085 \$====> 85% HOT borated water + 15% SS304							
1001	-0.0358870	8016	-0.2847761	5010	-1.15699e-4	5011	-5.270754e-4
26054	-0.0264402	26056	-0.4303714	26057	-0.0101153	26058	-0.0013721
14028	-0.0062349	14029	-0.0003281	14030	-0.0002240	24050	-0.0053825
24052	-0.1079314	24053	-0.0124748	24054	-0.0031632	25055	-0.0135739
28058	-0.0410462	28060	-0.0163554	28061	-0.0007226	28062	-0.0023425
28064	-0.0006157						
mt4085 lwtr.60t							
m4070 \$====> 70% HOT borated water + 20% Zr + 10% void							
1001	-0.0292856	8016	-0.2323919	5010	-9.44159e-5	5011	-4.30120e-4
40090	-0.3741087	40091	-0.0824924	40092	-0.1274775	40094	-0.1320002
40096	-0.0217192						
mt4070 lwtr.60t							
m4050 \$====> 50% hot borated water + 50% SS304							
1001	-0.0086117	8016	-0.0683369	5010	-2.77638e-5	5011	-1.26481e-4
26054	-0.0359537	26056	-0.5852247	26057	-0.0137549	26058	-0.0018658
14028	-0.0084783	14029	-0.0004461	14030	-0.0003046	24050	-0.0073191
24052	-0.1467666	24053	-0.0169634	24054	-0.0043013	25055	-0.0184579

```

28058 -0.0558152 28060 -0.0222403 28061 -0.0009826 28062 -0.0031854
28064 -0.0008373
mt4050 lwtr.60t
c
m5000 $====> low Carbon steel SA 508, Grade 2 (use Fe for nuclides < 0.1%)
26054 -0.05437 26056 -0.88501 26057 -0.02080 26058 -0.00282
6000 -0.0025 14028 -0.00367 14029 -0.00019 14030 -0.00013
24050 -0.000104 24052 -0.002092 24053 -0.000242 24054 -0.000061
25055 -0.010 28058 -0.006720 28060 -0.002678 28061 -0.000118
28062 -0.000384 28064 -0.000101 29063 -0.001370 29065 -0.000630
42000 -0.006
c
c every fuel pin, with 100 axial segments
fmesh104:n geom=xyz origin= -182.07 -182.07 -183.00
imesh= 182.07 iints= 289
jmesh= 182.07 jint= 289
kmesh= 183.00 kints= 100
fm104 -1. 0 -6 -8 $ macroscopic Sigma_fis * Q_fis
c every assembly, axially integrated
fmesh204:n geom=xyz origin= -182.07 -182.07 -183.00
imesh= 182.07 iints= 17
jmesh= 182.07 jint= 17
kmesh= 183.00 kints= 1
fm204 -1. 0 -6 -8
c
fmesh314:n geom=xyz origin= -10.71 -10.71 -183.00
imesh= 10.71 iints= 1
jmesh= 10.71 jint= 1
kmesh= 183.00 kints= 1
fm314 -1. 0 -6 -8
fmesh324:n geom=xyz origin= 53.55 32.13 -183.00
imesh= 74.97 iints= 1
jmesh= 53.55 jint= 1
kmesh= 183.00 kints= 1
fm324 -1. 0 -6 -8
fmesh334:n geom=xyz origin= -74.97 32.13 -183.00
imesh= -53.55 iints= 1
jmesh= 53.55 jint= 1
kmesh= 183.00 kints= 1
fm334 -1. 0 -6 -8
fmesh344:n geom=xyz origin= -74.97 -53.55 -183.00
imesh= -53.55 iints= 1
jmesh= -32.13 jint= 1
kmesh= 183.00 kints= 1
fm344 -1. 0 -6 -8
fmesh354:n geom=xyz origin= 53.55 -53.55 -183.00
imesh= 74.97 iints= 1
jmesh= -32.13 jint= 1
kmesh= 183.00 kints= 1
fm354 -1. 0 -6 -8
fmesh364:n geom=xyz origin= -74.97 -182.13 -183.00
imesh= -53.55 iints= 1
jmesh=-160.65 jint= 1
kmesh= 183.00 kints= 1
fm364 -1. 0 -6 -8
fmesh374:n geom=xyz origin= -139.23 117.81 -183.00
imesh=-117.81 iints= 1
jmesh= 139.23 jint= 1
kmesh= 183.00 kints= 1
fm374 -1. 0 -6 -8
fmesh384:n geom=xyz origin= 117.23 117.81 -183.00
imesh= 139.23 iints= 1
jmesh= 139.23 jint= 1

```

```

      kmesh= 183.00   kints= 1
fm384  -1. 0  -6 -8
fmesh394:n  geom=xyz  origin=  53.55   32.13  -183.00
            imesh=  84.81   iints= 1
            jmesh=  33.39   jints= 1
            kmesh= 183.00   kints= 1
fm394  -1. 0  -6 -8
fmesh404:n  geom=xyz  origin=  66.15   43.47  -183.00
            imesh=  67.41   iints= 1
            jmesh=  44.73   jints= 1
            kmesh= 183.00   kints= 1
fm404  -1. 0  -6 -8
fmesh414:n  geom=xyz  origin=  66.15   43.47  -183.00
            imesh=  67.41   iints= 1
            jmesh=  44.73   jints= 1
            kmesh=-179.34   kints= 1
fm414  -1. 0  -6 -8
fmesh424:n  geom=xyz  origin=  66.15   43.47   0.00
            imesh=  67.41   iints= 1
            jmesh=  44.73   jints= 1
            kmesh=   3.66   kints= 1
fm424  -1. 0  -6 -8
fmesh434:n  geom=xyz  origin=  66.15   43.47  179.34
            imesh=  67.41   iints= 1
            jmesh=  44.73   jints= 1
            kmesh= 183.00   kints= 1
fm434  -1. 0  -6 -8

```

## REFERENCES

- [1] X-5 Monte Carlo Team, "MCNP – A General Monte Carlo N-particle Transport Code," Version 5, Los Alamos National Laboratory, (2003)
- [2] E. E. Lewis and W. F. Miller, Jr., "Computational Methods of Neutron Transport," American Nuclear Society, Inc., 1993.
- [3] J. Spanier and E. M. Gelbard, "Monte Carlo Principles and Neutron Transport Problems," Addison-Wesley, Reading, MA, 1969.
- [4] C. E. Shannon, "A Mathematical Theory of Communication," The Bell System Technical Journal, 27, 379– 423, 623– 656 (1948).
- [5] T. Ueki and F.B. Brown, "Stationarity Diagnostics Using Shannon Entropy in Monte Carlo Criticality Calculation I: F test," Trans. Am. Nuc, 87, 156 (2002).
- [6] O. Jacquet et al., "Eigenvalue Uncertainty Evaluation in MC Calculations, Using Time Series Methodologies," Advanced Monte Carlo for Radiation Physics, Particle Transport Simulation and Applications, A. Kling et al., Eds., Springer-Verlag, Berlin Heidelberg (2001).
- [7] T. Kitada and T. Takeda, "Effective Convergence of Fission Source Distribution in Monte Carlo Simulation," J. Nucl. Sci. Eng., 38, 324 (2001).
- [8] H. J. Shim and C. H. Kim, "Stopping Criteria of Inactive Cycle Monte Carlo Calculations," Nucl. Sci. Eng., 157, 132 (2007).
- [9] F. Brown et al., "Convergence Testing for MCNP5 Monte Carlo Eigenvalue Calculations," Joint International Topical Meeting on Mathematics & Computation and Supercomputing in Nuclear Applications (M&C + SNA 2007), Monterey, California, April 15-19, 2007, on CD-ROM, American Nuclear Society, LaGrange Park, IL (2007).
- [10] T. Ueki, "On-The-Fly Judgements of Monte Carlo Fission Source Convergence," Trans. Am. Nucl. Sci., 98, 512 (2008).
- [11] P. Romano, "Application of the Stochastic Oscillator to Assess Source Convergence in Monte Carlo Criticality Calculations, Intl. Conf. on Mathematics," Computational Methods and Reactor Physics (M&C 2009).
- [12] "Westinghouse Burnable Absorbers – Advanced Core Management Products," NF-FE-0009, Westinghouse Electric Company, May 2006.

- [13] I. Lux and L Koblinger, “Monte Carlo Particle Transport Methods: Neutron and Photon Calculations,” CRC Press, Boca Raton, Florida, 1991.
- [14] J. E. Hoogenboom, W. R. Martin and B. Petrovic, “Monte Carlo Performance Benchmark for Detailed Power Density Calculation in a Full Size Reactor Core” Benchmark Specifications Revision 1.2, July 2011, [http://www.oecd-nea.org/dbprog/documents/MonteCarlobenchmarkguideline\\_004.pdf](http://www.oecd-nea.org/dbprog/documents/MonteCarlobenchmarkguideline_004.pdf) (2011).
- [15] J. E. Hoogenboom, W. Martin, B. Petrovic, “The Monte Carlo Performance Benchmark Test: Aims, Specifications and First Results,” Proc. Intl. Conf. Math. Comp. (M&C’2011), Rio de Janeiro, Brazil, May 8-12, 2011.
- [16] D. J. Kelly et al., “MC21 Monte Carlo Analysis of the Hoogenboom-Martin Full-Core PWR Benchmark Problem,” Proceedings of PHYSOR 2010 – Advances in Reactor Physics to Power the Nuclear Renaissance, Pittsburgh, Pennsylvania, USA, May 9-14, 2010, on CD-ROM American Nuclear Society, LaGrange Park, IL (2010).