Transactions of the Korean Nuclear Society Autumn Meeting
Gyeongju, Korea, October 29-30, 2015

Uncertainty Propagation Analysis for the Monte Carlo Time-Dependent Simulations

Nadeem Shaukat[a], Hyung Jin Shim[a*]
[a]*Seoul National University, 1 Gwanak-ro, Gwanak-gu, Seoul 151-742, Korea*
[*]*Corresponding author: shimhj@snu.ac.kr*

1. Introduction

In order to check the transient behavior and safety analysis of nuclear reactors, it is required to solve the space-time dependent neutron transport equation. Due to flexibility of Monte Carlo methods for neutron cross section and complex nuclear reactor geometry, it has received increasing attention with the development of computer hardware [1-4]. To simulate the time-dependent behavior of a nuclear reactor, based on the Monte Carlo neutron transport method, the direct simulation method (DSM), has been used in MCNP [5]. A time-dependent simulation mode has also been implemented in the development version of SERPENT 2 Monte Carlo code. In this mode, sequential population control mechanism has been proposed for modeling of prompt super-critical systems [6]. A Monte Carlo method has been properly used in TART code for dynamic criticality calculations. For super-critical systems, the neutron population is allowed to grow over a period of time. The neutron population is uniformly combed to return it to the neutron population started with at the beginning of time boundary [7].

In this study, conventional time-dependent Monte Carlo (TDMC) algorithm is implemented. There is an exponential growth of neutron population in estimation of neutron density tally for super-critical systems and the number of neutrons being tracked exceed the memory of the computer. In order to control this exponential growth at the end of each time boundary, a conventional time cut-off controlling population strategy is included in TDMC [7]. A scale factor is introduced to tally the desired neutron density at the end of each time boundary.

The main purpose of this paper is the quantification of uncertainty propagation in neutron densities at the end of each time boundary for super-critical systems. This uncertainty is caused by the uncertainty resulting from the introduction of scale factor. The effectiveness of TDMC is examined for one-group infinite homogeneous problem (the rod model) and two-group infinite homogeneous problem.

2. Neutron History Based Monte Carlo Time-Dependent Scheme

In this study, the Monte Carlo time-dependent method is implemented, which is based on neutron history, thus it can also be called as neutron history based method (NHBM). In this method, all the neutron histories are tracked successively from the very beginning of time boundary to the end of each time boundary when all of its progenies are disappeared. Figure 1 shows a real simulation of a typical neutron histories in the time-dependent process, where each neutron history refers to the solid lines, representing the free paths and collisions of the source neutrons. All the progenies within a certain time interval just create history branches by fission reactions instead of new neutron histories. On the horizontal axis, t_0 represents the initial time of the time-dependent simulation and T represents the fixed step for the continuation of next time bins.

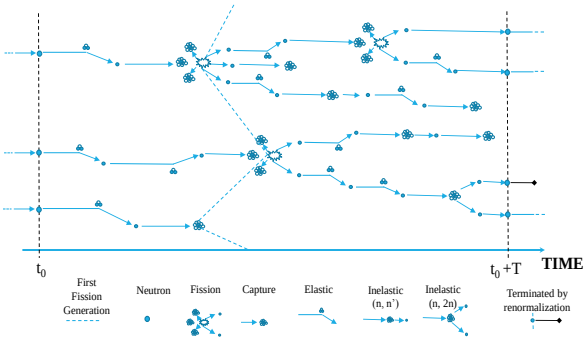

Figure 1: Division of History Based Neutron Simulation in Time.

In Figure 2, the general scheme for performing TDMC calculation is presented. At the start of each time boundary, system properties are read. Neutrons are simulated and the new neutrons due to fission and splitting are simulated directly after the current particle. Neutron history is followed, until it crosses the time boundary. When a neutron crosses the time boundary, it is stopped and stored for the next time interval, (similar to fission neutrons in critical bank). New path lengths and new system properties after the end of each time boundary are updated.

An integral form of the time-dependent Boltzmann transport equation for the collision density $\psi(\mathbf{P})$, where $\mathbf{P}$ denotes the state vector of a neutron in the seven-dimensional phase space (three in space, two in direction, and one each in energy and time) $(\mathbf{r}, E, \hat{\mathbf{\Omega}}, t)$ can be written as;

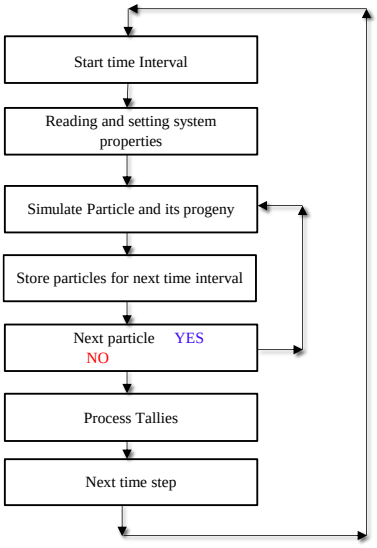

Figure 2: Scheme for Performing Time-Dependent Monte Carlo Calculation.

$$\psi(\bar{\mathbf{P}}) = \tilde{S}(\bar{\mathbf{P}}) + \int d\bar{\mathbf{P}}' K(\bar{\mathbf{P}}' \to \bar{\mathbf{P}})\psi(\bar{\mathbf{P}}') \quad (1a)$$

where;

$$\psi(\bar{\mathbf{P}}) = \Sigma_t(\bar{\mathbf{r}}, E)\phi(\bar{\mathbf{r}}, E, \hat{\mathbf{\Omega}}, t) \quad (1b)$$

= Collision Density

$$K(\bar{\mathbf{P}}' \to \bar{\mathbf{P}}) = C(\bar{\mathbf{r}}', t'; E', \hat{\mathbf{\Omega}}' \to E, \hat{\mathbf{\Omega}})T(E, \hat{\mathbf{\Omega}}, t; \bar{\mathbf{r}}' \to \bar{\mathbf{r}}) \quad (1c)$$

= Transport Kernel

$$C(\bar{\mathbf{r}}', t'; E', \hat{\mathbf{\Omega}}' \to E, \hat{\mathbf{\Omega}}) = \sum_r \frac{\Sigma_r(\bar{\mathbf{P}}')}{\Sigma_t(\bar{\mathbf{P}}')} \nu_r f_r(E', \hat{\mathbf{\Omega}}' \to E, \hat{\mathbf{\Omega}}) \quad (1d)$$

= Collision Kernel

$$T(E, \hat{\mathbf{\Omega}}, t; \bar{\mathbf{r}}' \to \bar{\mathbf{r}}) = \frac{\Sigma_t(\bar{\mathbf{r}}, E)}{|\bar{\mathbf{r}} - \bar{\mathbf{r}}'|} \exp\left[-\int_0^{|\bar{\mathbf{r}} - \bar{\mathbf{r}}'|} \Sigma_t\left(\bar{\mathbf{r}} - s\frac{\bar{\mathbf{r}} - \bar{\mathbf{r}}'}{|\bar{\mathbf{r}} - \bar{\mathbf{r}}'|}, E\right) ds\right] \delta\left(\hat{\mathbf{\Omega}} \cdot \frac{\bar{\mathbf{r}} - \bar{\mathbf{r}}'}{|\bar{\mathbf{r}} - \bar{\mathbf{r}}'|} - 1\right) \quad (1e)$$

= Transition Kernel

$$\tilde{S}(\bar{\mathbf{P}}) = \int d\bar{\mathbf{P}} S(\bar{\mathbf{r}}'', E, \hat{\mathbf{\Omega}}, t) T(E, \hat{\mathbf{\Omega}}, t; \bar{\mathbf{r}}'' \to \bar{\mathbf{r}}) \quad (1f)$$

= First Collision Density of Source

Where, ν_r is average number of neutrons produced from reaction type *r*, f_r is probability that a collision of type *r* by a neutron direction $\hat{\mathbf{\Omega}}'$ and energy E' will produce a neutron in direction interval $d\hat{\mathbf{\Omega}}$ about $\hat{\mathbf{\Omega}}$ with energy dE about E and S is source distribution [8].

After reviewing Neumann series solution, the neutron density tally at the end of m^{th} time step can be calculated as;

$$N^m(\mathbf{r}, t) = \frac{1}{n_m}\left(f^{m-1}\sum_{i=1}^{n_m}\sum_j w_{i,j}q_{i,j}\right) \quad (2)$$

where, f^{m-1} is a scale factor introduced at the end of *(m-1)*th time step, $w_{i,j}$ is weight of neutron after the j^{th} collision of neutron i and $q_{i,j}$ is response of neutron density tally for the j^{th} collision of neutron *i*.

In order to obtain the desired neutron density level, a scale factor is introduced. It is defined as the sequence based on the ratio of the number of neutrons survived at the time boundary to the number of neutron histories gives the next term as a function of previous term. Mathematically, the scale factor at the m^{th} time step can be written as;

$$f^m = f^{m-1} \cdot \frac{n_{s,m}}{n_m} \quad (3)$$

Where, $n_{s,m}$ and n_m are the number of survival neutrons and the number of neutron histories at the end of m^{th} time step respectively. A recursive formulation for the scale factor by using the preceding term to define the next term of the sequence can be written as;

$$f^m = \prod_{k=0}^{m}\left(\frac{n_{s,k}}{n_k}\right) \quad (4)$$

By substituting the value of scale factor from Eq. (4) into Eq. (2), the Neutron density tally can be rewritten as;

$$N^m(\mathbf{r}, t) = \frac{1}{n_m}\left(\prod_{k=0}^{m-1}\left(\frac{n_{s,k}}{n_k}\right)\cdot\sum_{i=1}^{n_m}\sum_j w_{i,j}q_{i,j}\right) \quad (5)$$

The objective of the introduction of scaling factor is to change the level of neutron density to the desired amount in super-critical reactors.

To control the exponential growth of neutron population and handle the memory of computer, population controlling mechanism must be applied to the neutron histories. Figure 4 represents the algorithm for controlling neutron population in Monte Carlo time-dependent simulations. After completing all the neutron histories at the end of each time boundary, neutron population controlling mechanism is imposed on the banked neutrons. To match the neutron population size to the initial number of neutrons at the beginning of each time simulation, all the randomly selected neutrons are discarded [7].

2.1 Time-Dependent Neutron Density Estimator

There is no time information with the neutrons simulated in the Monte Carlo criticality calculations. However, it is needed to save the time information to describe the time-dependent behavior of neutrons, so that time mark is introduced in the TDMC algorithm. In Figure 3, t_0 denotes the start time of the time-dependent simulation, t_m represents the time boundary, i.e., time cut-off. $f^0, f^1, f^2, .., f^{m-1}$ are calculated at the end of each time bin. For neutron

density tally, f^0 contributes to the time bin $t_0 \sim t_1$, f^1 contributes to the time bin $t_1 \sim t_2$, and f^{m-1} contributes to the time bin $t_{m-1} \sim t_m$.

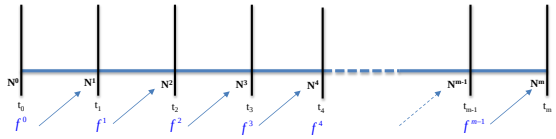

Figure 3: Time-Dependent Neutron Density Tally.

3. Uncertainty Propagation in Neutron Density Tally

The statistical uncertainty of one time step is accumulated with the uncertainty propagating through the calculation from the preceding time step. The main purpose of this section is to study this uncertainty propagation at the end of each time boundary. With the help of formal uncertainty propagation approach, it is investigated that how a change in the scale factor at the end of each time boundary affects and propagates the uncertainty in the neutron density of the system. That is a term affected by the propagation of statistical uncertainty in the neutron density from the Monte Carlo time-dependent simulation [9].

The discussion starts with the uncertainty propagated in scale factor which is a main source of uncertainty in calculating neutron density $\left(N^m\right)$ during time-dependent Monte Carlo calculations and this propagation is continued with the term affected by scale factor $\left(f^{m-1}\right)$. Since uncertainty in scale factor further depends on the change in number of survival neutrons $\left(n_{s,k}\right)$ at the end of each time boundary, so the change in the neutron density distribution can be written as;

$$\Delta N^m = \Delta N^m\left(\left\{n_{s,k}\right\}_{k=0}^{m-1}\right) = \frac{1}{n_m}\left(\prod_{k=0}^{m-1}\left(\frac{n_{s,k}}{n_k}\right)\sum_{i=1}^{n_m}\sum_j w_{i,j}q_{i,j}\right) \qquad (6a)$$

Eq. (6) can also be written as;

$$\Delta N^m\left(\left\{n_{s,k}\right\}_{k=0}^{m-1}\right) = \prod_{k=0}^{m-1}\left(\frac{n_{s,k}}{n_k}\right)\cdot A_0 \qquad (6b)$$

Where;

$$A_0 = \frac{1}{n_m}\left(\sum_{i=1}^{n_m}\sum_j w_{i,j}q_{i,j}\right)$$

The formal error propagation approach is to compute standard deviation from the survival neutron components $\left\{n_{s,k}\right\}_{k=0}^{m-1}$ and combine all into standard deviation for scale factor f^{m-1} using the approximation for products of (m-1) variables (ignoring the covariance between $\left\{n_{s,k}\right\}_{k=0}^{m-1}$).

The general formulation for uncertainty propagation in absolute units can be written as;

$$S_{N^{m-1}} = N^{m-1}\sqrt{\sum_{k=1}^{m-1}\frac{S_{n_{s,k}}^2}{\left(n_{s,k}\right)^2}}\ ; \qquad m = 2,3,4,... \qquad (7)$$

In percent units, uncertainty propagation can be written as;

$$\frac{S_{N^{m-1}}}{N^{m-1}} = \sqrt{\sum_{k=1}^{m-1}\frac{S_{n_{s,k}}^2}{\left(n_{s,k}\right)^2}}\ ; \qquad m = 2,3,4,... \qquad (8)$$

where, $n_{s,k}$ represents the number of survival neutrons at the end of k^{th} time step and $S_{n_{s,k}}$ is a standard deviation of survival neutrons at the end of k^{th} time step, calculated as;

$$S_{n_{s,m-1}}^2 = \frac{1}{m-2}\left\{\sum_{i=1}^{m-1}\left(n_{s_i}\right)^2 - \frac{\left(\sum_{i=1}^{m-1}n_{s_i}\right)^2}{m-1}\right\} \qquad (9)$$

In operator notation, Eq. (8) can expressed as;

$$e_{N^m} = \left(\frac{S_{N^m}}{N^m}\right)^2 = e_{N^{m-1}} + \varepsilon_{N^m} \qquad (10)$$

Where, $\varepsilon_{N^m} = \frac{S_{n_{s,m}}^2}{\left(n_{s,m}\right)^2}$

The repeated application of Eq. (10) gives;

$$e_{N^m} = e_{N^0} + \sum_{i=1}^{m}\varepsilon_{N^i} \qquad (11)$$

4. Numerical Results and Analysis

4.1 The Rod Model

The rod model is a simplest example of time-dependent neutron transport through time independent media, i.e., there is no change in material with time through which neutron is transporting. An infinite homogeneous and isotropic media in which neutrons move at constant speed υ along the line (the rod) and undergo collision events at a rate $\nu\Sigma$. A true analog calculation is done for this Monte Carlo time-dependent simulation without considering the source term S. In terms of neutron density, time-dependent Boltzmann transport equation can be written as [7, 10];

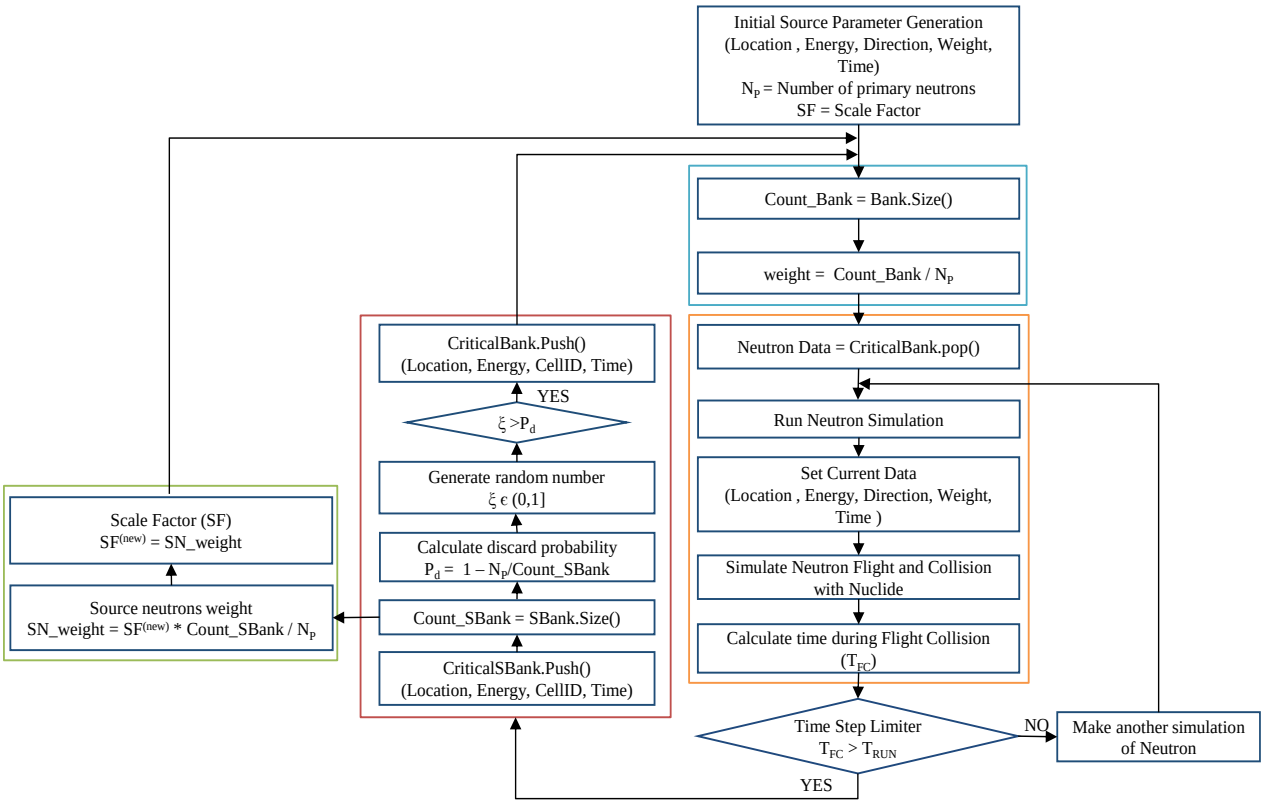

Figure 4: Algorithm for Controlling Neutron Population in Monte Carlo Time-Dependent Simulations.

$$\frac{\partial N}{\partial t}+\hat{\Omega}.\nabla\left(N*\upsilon\right)+\Sigma_t\left(N*\upsilon\right)=\int\int\left[\nu\Sigma_f+\Sigma_s\right]F\left(E',\hat{\Omega}'\rightarrow E,\hat{\Omega}\right)\left(N*\upsilon\right)dE'd\hat{\Omega}' \quad (12)$$

Under certain assumptions, Eq. (12) can be interpreted as;

$$\frac{dN}{dt}+\upsilon\left(L*N+\Sigma_t*N\right)=\upsilon\left(\nu\Sigma_f+\Sigma_s\right)N \quad (13)$$

The above equation can be interpreted as;

$$\frac{1}{N}\frac{dN}{dt}=\frac{d\left(\ln N\right)}{dt}=\left[\left(K-1\right)/T_R\right] \quad (14)$$

Where;

$$K=\nu\Sigma_f/\left(L+\Sigma_a\right);\qquad T_R=1/\left[\nu\left(L+\Sigma_a\right)\right]$$

The solution of above equation under the assumption that the properties of the medium are time independent is given by;

$$N\left(t\right)=N\left(0\right)\exp\left[\left(K-1\right)/T_R\right] \quad (15)$$

Table 1 shows one-group cross sections for infinite homogeneous super critical system with $k_\infty = 1.875$.

Table 1: One-Group Cross Sections for Infinite Homogeneous Problem.

ν	**Σ_f**	**Σ_a**	**Σ_s**	**$1/\upsilon$ [sec/cm]**
2.5	0.3	0.4	0.2	1.02245×10^{-9}

The TDMC neutron density calculations are performed for 10,000 neutron histories and 200 time steps with time step size 1.0nsec. Table 2 shows comparisons of neutron density calculated by TDMC algorithm and analytical solutions. From the table it can be seen that the results from the implemented method agree well with the results calculated from analytical solution within 95% confidence intervals.

In true analog calculation, it is started from any instantaneous source distribution, that is the initial condition N(0) and it is followed the evolution of neutron density in time. Figure 5 represents the linear increase in neutron density with time on logarithmic plot. The secondary y-axis represents the relative standard deviation with the evolution of neutron density in time. The lower right corner plot is a view of some part of the logarithmic plot represents the exponential increase in neutron density with time.

Table 2: Comparison of Neutron Densities for One-Group Infinite Homogeneous Problem.

Time (sec)	Neutron Density		Relative Std. Dev. (RSD) [%]
	Exact Solution	TDMC Solution	
$1.00x10^{-9}$	$1.40820x10^{4}$	$1.41570x10^{4}$	0.84
$2.00x10^{-9}$	$1.98304x10^{4}$	$1.99911x10^{4}$	0.83
$3.00x10^{-9}$	$2.79253x10^{4}$	$2.79176x10^{4}$	0.83
$4.00x10^{-9}$	$3.93245x10^{4}$	$3.93638x10^{4}$	0.84
$5.00x10^{-9}$	$5.53770x10^{4}$	$5.51920x10^{4}$	0.82
$6.00x10^{-9}$	$7.79821x10^{4}$	$7.81573x10^{4}$	0.83
$7.00x10^{-9}$	$1.09815x10^{5}$	$1.09866x10^{5}$	0.83
$8.00x10^{-9}$	$1.54642x10^{5}$	$1.54878x10^{5}$	0.85
$9.00x10^{-9}$	$2.17767x10^{5}$	$2.19756x10^{5}$	0.84
$1.00x10^{-8}$	$3.06661x10^{5}$	$3.05351x10^{5}$	0.84
$1.10x10^{-8}$	$4.31842x10^{5}$	$4.38545x10^{5}$	0.86
$1.20x10^{-8}$	$6.08121x10^{5}$	$6.23874x10^{5}$	0.86
$1.30x10^{-8}$	$8.56360x10^{5}$	$8.91953x10^{5}$	0.84
$1.40x10^{-8}$	$1.20593x10^{6}$	$1.26711x10^{6}$	0.85
$1.50x10^{-8}$	$1.69820x10^{6}$	$1.81159x10^{6}$	0.84
$1.60x10^{-8}$	$2.39141x10^{6}$	$2.55289x10^{6}$	0.85
$1.70x10^{-8}$	$3.36759x10^{6}$	$3.56664x10^{6}$	0.85
$1.80x10^{-8}$	$4.74226x10^{6}$	$5.04287x10^{6}$	0.83
$1.90x10^{-8}$	$6.67808x10^{6}$	$7.13263x10^{6}$	0.84

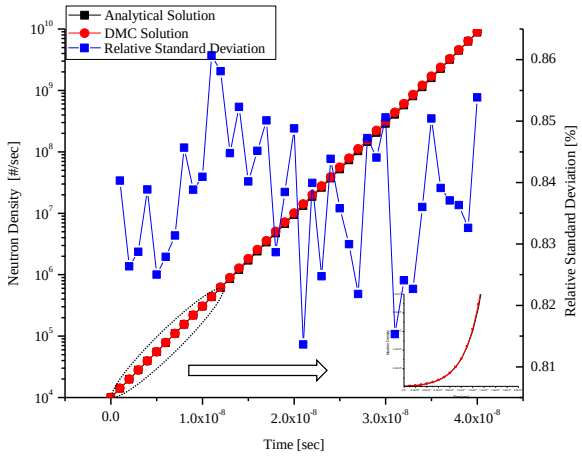

Figure 5: Exponential variation of Neutron Density Tally with Time.

It is analyzed that there is a propagation in statistical uncertainty in neutron density. Table 3 shows the numerical results for the propagation in standard deviation for neutron density tally and real standard deviation of the mean for 100 independent TDMC simulations. It is observed that the results are within 95% confidence intervals. Figure 6 represents the propagation in standard deviation in neutron density with time by using proposed uncertainty propagation model compared with the real standard deviation (RSD).

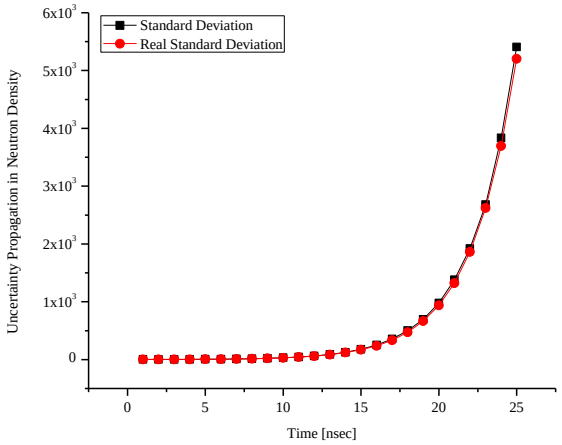

Figure 6: Standard Deviation vs Real Standard Deviation.

Table 3: Comparison of Standard Deviation and Real Standard Deviation in Neutron Densities.

Time (sec)	Std. Dev. in Neutron Density	Real Standard Deviation [RSD]	Difference
$1.0x10^{-9}$	$1.402x10^{0}$	$1.404x10^{0}$	-0.002
$2.0x10^{-9}$	$1.963x10^{0}$	$1.978x10^{0}$	-0.015
$3.0x10^{-9}$	$2.776x10^{0}$	$2.784x10^{0}$	-0.008
$4.0x10^{-9}$	$3.886x10^{0}$	$3.914x10^{0}$	-0.028
$5.0x10^{-9}$	$5.518x10^{0}$	$5.517x10^{0}$	0.001
$6.0x10^{-9}$	$7.820x10^{0}$	$7.776x10^{0}$	0.044
$7.0x10^{-9}$	$1.103x10^{1}$	$1.095x10^{1}$	0.08
$8.0x10^{-9}$	$1.574x10^{1}$	$1.544x10^{1}$	0.3
$9.0x10^{-9}$	$2.188x10^{1}$	$2.171x10^{1}$	0.17
$1.0x10^{-8}$	$3.154x10^{1}$	$3.058x10^{1}$	0.96
$1.1x10^{-8}$	$4.434x10^{1}$	$4.303x10^{1}$	1.31
$1.2x10^{-8}$	$6.279x10^{1}$	$6.057x10^{1}$	2.22
$1.3x10^{-8}$	$8.828x10^{1}$	$8.544x10^{1}$	2.84
$1.4x10^{-8}$	$1.269x10^{2}$	$1.204x10^{2}$	6.5
$1.5x10^{-8}$	$1.795x10^{2}$	$1.694x10^{2}$	10.1
$1.6x10^{-8}$	$2.513x10^{2}$	$2.381x10^{2}$	13.2
$1.7x10^{-8}$	$3.550x10^{2}$	$3.359x10^{2}$	19.1
$1.8x10^{-8}$	$5.006x10^{2}$	$4.729x10^{2}$	27.7
$1.9x10^{-8}$	$6.940x10^{2}$	$6.659x10^{2}$	28.1

4.2 Two-Group Infinite Homogeneous Problem

Two-group infinite homogeneous problem with speeds, υ_1 (fast) and υ_2 (thermal) is considered as a simplest non-trivial example. It is assumed that neutrons are transported in an infinite media with no up-scattering, that there is an emission of prompt neutrons only in group $g = 1$. If there is no external source then the time-dependent two group neutron transport equations for flux distribution can be interpreted as [10];

$$\begin{aligned} \frac{1}{\upsilon_1}\frac{\partial \phi_1}{\partial t} + \Sigma_{t1}\phi_1 &= \Sigma_{s11}\phi_1 + \Sigma_{s12}\phi_2 \\ \frac{1}{\upsilon_2}\frac{\partial \phi_2}{\partial t} + \Sigma_{t2}\phi_2 &= \Sigma_{s22}\phi_2 + \nu_1\Sigma_{f1}\phi_1 + \nu_2\Sigma_{f2}\phi_2 \end{aligned} \quad (16)$$

The solution of above system of equations gives group flux. To calculate the group neutron density, group flux is divided by the corresponding group neutron speed. The reference physical parameters for a super-critical infinite homogeneous problem with $k_\infty = 1.10075$ are given in Table 4.

Table 4: Two-Group Cross Sections for Infinite Homogeneous Problem.

Cross Section	**Thermal Group (g=1)**	**Fast Group (g=2)**
ν_g	2.50	2.70
Σ_{fg}	0.06912	0.06192
Σ_{ag}	0.13862	0.16142
Σ_{sgg}	0.26304	0.078242
$\Sigma_{sg'g}$ (g′≠g)	0.0	0.072
Σ_{tg}	0.40166	0.31166
χ_g	0.0	1.0
$1/\upsilon_g$ [sec/cm]	9.14505×10^{-7}	7.22980×10^{-7}

The results of the Monte Carlo time-dependent simulations for 10,000 neutron histories and 200 time steps with time step size 10.0μsec are interpreted in Table 5 and compared with the solution found by MATLAB. Figure 7 and Figure 8 represent the neutron density in thermal and fast energy groups respectively along with the variation in relative standard deviation.

Numerical results for standard deviation and real standard deviation computed for 100 independent simulations for the neutron density are shown in Table 6. It is observed that the results are within 95% confidence intervals. Figure 9 represents the propagation in standard deviation in neutron density with time by using proposed uncertainty propagation model compared with the real standard deviation (RSD) in both thermal and fast energy groups.

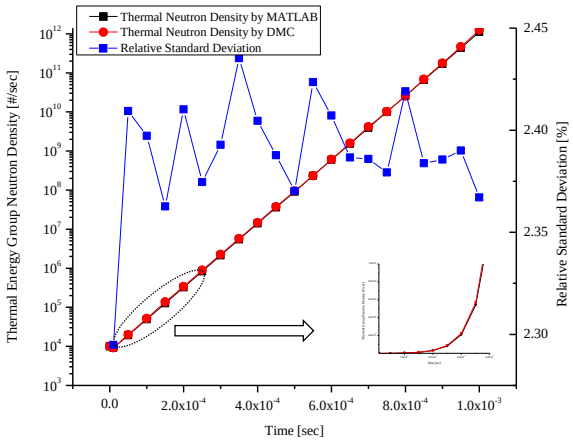

Figure 7: Neutron Density in Thermal Energy Group.

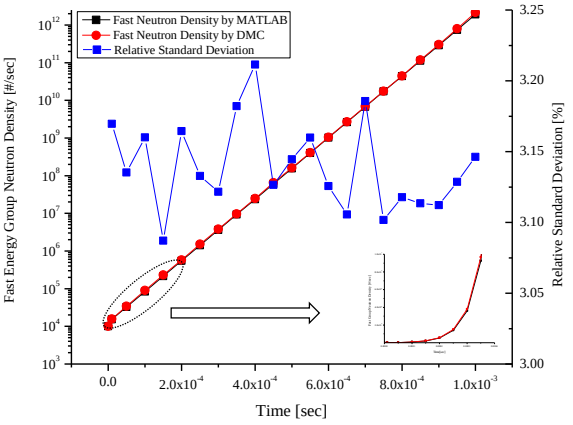

Figure 8: Neutron Density in Fast Energy Group.

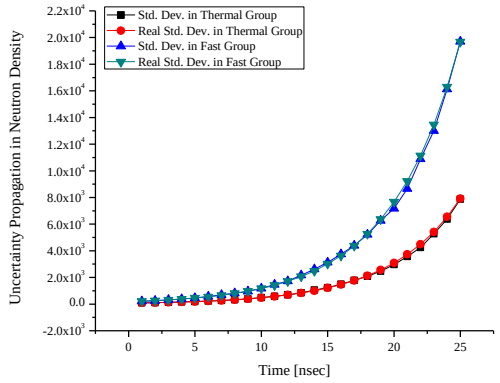

Figure 9: Standard Deviation vs Real Standard Deviation.

Table 5: Comparison of Neutron Densities for Two-Group Infinite Homogeneous Problem.

Time (sec)	Neutron Density (MATLAB Solution)		Neutron Density (TDMC Solution)		Relative Std. Dev. (RSD) [%]	
	Thermal	Fast	Thermal	Fast	Thermal	Fast
1.0×10^{-5}	9.22414×10^{3}	1.52169×10^{4}	9.26600×10^{3}	1.56180×10^{4}	2.29	3.17
5.0×10^{-5}	1.91845×10^{4}	3.27403×10^{4}	1.99110×10^{4}	3.42428×10^{4}	2.41	3.14
1.0×10^{-4}	4.91904×10^{4}	8.38535×10^{4}	5.18764×10^{4}	8.95993×10^{4}	2.40	3.16
1.5×10^{-4}	1.26036×10^{5}	2.14864×10^{5}	1.36989×10^{5}	2.30874×10^{5}	2.36	3.09
2.0×10^{-4}	3.22808×10^{5}	5.50701×10^{5}	3.41171×10^{5}	5.77859×10^{5}	2.41	3.16
2.5×10^{-4}	8.26774×10^{5}	1.41147×10^{6}	8.95560×10^{5}	1.51842×10^{6}	2.37	3.13
3.0×10^{-4}	2.11904×10^{6}	3.61598×10^{6}	2.26043×10^{6}	3.80371×10^{6}	2.39	3.12
3.5×10^{-4}	5.43277×10^{6}	9.26183×10^{6}	5.70027×10^{6}	9.67843×10^{6}	2.44	3.18
4.0×10^{-4}	1.39216×10^{7}	2.37304×10^{7}	1.47221×10^{7}	2.48614×10^{7}	2.40	3.21
4.5×10^{-4}	3.56597×10^{7}	6.08179×10^{7}	3.77051×10^{7}	6.54494×10^{7}	2.39	3.13
5.0×10^{-4}	9.13504×10^{7}	1.55858×10^{8}	9.44370×10^{7}	1.61481×10^{8}	2.37	3.14
5.5×10^{-4}	2.34109×10^{8}	3.99311×10^{8}	2.34241×10^{8}	4.06923×10^{8}	2.42	3.16
6.0×10^{-4}	5.99865×10^{8}	1.02316×10^{9}	6.10370×10^{8}	1.05023×10^{9}	2.41	3.13
6.5×10^{-4}	1.53744×10^{9}	2.62121×10^{9}	1.57753×10^{9}	2.69890×10^{9}	2.39	3.11
7.0×10^{-4}	3.93582×10^{9}	6.72033×10^{9}	4.17350×10^{9}	6.88929×10^{9}	2.39	3.19
7.5×10^{-4}	1.00854×10^{10}	1.72189×10^{10}	1.02135×10^{10}	1.74681×10^{10}	2.38	3.10
8.0×10^{-4}	2.58551×10^{10}	4.41058×10^{10}	2.53491×10^{10}	4.45106×10^{10}	2.42	3.12
8.5×10^{-4}	6.63131×10^{10}	1.12942×10^{11}	6.85577×10^{10}	1.17610×10^{11}	2.38	3.11
9.0×10^{-4}	1.69819×10^{11}	2.89499×10^{11}	1.76957×10^{11}	3.04030×10^{11}	2.39	3.11

Table 6: Comparison of Standard Deviation and Real Standard Deviation in Neutron Densities.

Time (sec)	Standard Deviation		Real Standard Deviation		Difference	
	Thermal	Fast	Thermal	Fast	Thermal	Fast
1.0×10^{-5}	8.620×10^{1}	2.131×10^{2}	8.609×10^{1}	2.145×10^{2}	0.11	-1.43
5.0×10^{-5}	1.064×10^{2}	2.577×10^{2}	1.044×10^{2}	2.585×10^{2}	2.08	-0.87
1.0×10^{-4}	1.281×10^{2}	3.177×10^{2}	1.262×10^{2}	3.133×10^{2}	1.88	4.37
1.5×10^{-4}	1.552×10^{2}	3.811×10^{2}	1.527×10^{2}	3.783×10^{2}	2.50	2.88
2.0×10^{-4}	1.824×10^{2}	4.646×10^{2}	1.835×10^{2}	4.552×10^{2}	-1.14	9.38
2.5×10^{-4}	2.270×10^{2}	5.695×10^{2}	2.215×10^{2}	5.482×10^{2}	5.50	21.32
3.0×10^{-4}	2.753×10^{2}	6.950×10^{2}	2.674×10^{2}	6.612×10^{2}	7.87	33.87
3.5×10^{-4}	3.247×10^{2}	8.209×10^{2}	3.223×10^{2}	8.013×10^{2}	2.38	19.67
4.0×10^{-4}	3.944×10^{2}	9.570×10^{2}	3.897×10^{2}	9.672×10^{2}	4.73	-10.20
4.5×10^{-4}	4.903×10^{2}	1.182×10^{3}	4.722×10^{2}	1.171×10^{3}	18.06	11.56
5.0×10^{-4}	5.888×10^{2}	1.462×10^{3}	5.690×10^{2}	1.416×10^{3}	19.79	45.85
5.5×10^{-4}	7.001×10^{2}	1.726×10^{3}	6.890×10^{2}	1.708×10^{3}	11.08	18.43
6.0×10^{-4}	8.406×10^{2}	2.172×10^{3}	8.319×10^{2}	2.067×10^{3}	8.65	104.56
6.5×10^{-4}	1.057×10^{3}	2.626×10^{3}	1.005×10^{3}	2.490×10^{3}	52.70	135.49
7.0×10^{-4}	1.230×10^{3}	3.136×10^{3}	1.212×10^{3}	3.022×10^{3}	17.51	113.70
7.5×10^{-4}	1.492×10^{3}	3.747×10^{3}	1.465×10^{3}	3.627×10^{3}	26.12	120.39
8.0×10^{-4}	1.788×10^{3}	4.392×10^{3}	1.764×10^{3}	4.372×10^{3}	23.90	20.30
8.5×10^{-4}	2.087×10^{3}	5.227×10^{3}	2.120×10^{3}	5.266×10^{3}	-32.19	-39.08
9.0×10^{-4}	2.476×10^{3}	6.260×10^{3}	2.561×10^{3}	6.358×10^{3}	-84.46	-98.31

5. Conclusions

In this paper, a conventional method to control the neutron population for super-critical systems is implemented. Instead of considering the cycles, the simulation is divided in time intervals. At the end of each time interval, neutron population control is applied on the banked neutrons. Randomly selected neutrons are discarded, until the size of neutron population matches the initial neutron histories at the beginning of time simulation. The desired neutron density is tallied by the introduction of scale factor. The uncertainty propagated in neutron density resulting from the uncertainty in scale factor at the end of each time interval can be calculated. Two test problems are demonstrated for verification of TDMC algorithm and it is seen that the results agree well with the results calculated from analytical solution within 95% confidence intervals. It is also analyzed that standard deviation and the real standard deviation are in good agreement.

6. Future Work

The next development will be the addition of controlling neutron population mechanism for sub-critical systems at the end of each time boundary. In this case, uncertainty propagation analysis will also be performed.

REFERENCES

[1] B. L. Sjenitzer, J.E. Hoogenboom, A Monte Carlo Method for Calculation of the Dynamic Behaviour of Nuclear Reactors, Nuclear Science and Technology, Vol. 2, pp.716-721, 2011.
[2] B. L. Sjenitzer, J.E. Hoogenboom, General Purpose Dynamic Monte Carlo with Continuous Energy for Transient Analysis, PHYSOR 2012 Advances in Reactor Physics Linking Research, Industry, and Education, American Nuclear Society, April 15-20, 2012.
[3] B. L. Sjenitzer, J.E. Hoogenboom, Implementation of the Dynamic Monte Carlo Method for Transient Analysis in the General Purpose Code Tripoli, Joint International Conference on Mathematics and Computation (M&C), Rio de Janeiro, RJ, Brazil, May 8-12, 2011.
[4] B. L. Sjenitzer, J.E. Hoogenboom, Dynamic Monte Carlo Method for Nuclear Reactor Kinetics Calculations, Nuclear Science and Engineering, Vol. 175, pp. 94-107, 2013.
[5] J. F. Briesmeister, Manual of MCNP-A General Monte Carlo N-Particle Transport Code: CCC7000, 2000.
[6] J. Leppanen, Developmet of a Dynamic Simulation Mode in SERPENT 2 Monte Carlo Code, Joint International Conference on Mathematics and Computation (M&C), Sun Valley, Idaho, May 5-9, 2013.
[7] D. E. Cullen, C. J. Clouse, R. Procassini, R. C. Little, Static and Dynamic Criticality are They Different?, Report: UCRL-TR-201506, U.S. Department of Energy, LLNL, November 22, 2003.
[8] H. J. Shim, MCCARD: Monte Carlo Code for Advanced Reactor Design and Analysis, Methodology Manual, Ver. 1.0, SNUMCL/TR001/2010(01), April 30, 2012.
[9] V. Valtavirta, Uncertainty Underprediction in Coupled Time-Dependent Monte Carlo Simulations with SERPENT 2, Joint International Conference on Mathematics and Computation (M&C), Nashville, Tennessee, April 19-23, 2015.
[10] A. Zoia, E. Brun, F. Malvagi, A Monte Carlo Method for Prompt and Delayed Alpha Eigenvalue Calculations, PHYSOR 2014, The West Miyako, Kyoto, Japan, September 28 - October 3, 2014.