LA-UR-07-1194

Approved for public release;
distribution is unlimited.

Title:	Wielandt Acceleration for MCNP5 Monte Carlo Eigenvalue Calculations
Author(s):	Forrest Brown
Intended for:	M&C+SNA-2007, ANS Mathematics & Computation Topical Meeting Monterey, CA, 15-19 April 2007

Form 836 (7/06)

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)

WIELANDT ACCELERATION FOR MCNP5 MONTE CARLO EIGENVALUE CALCULATIONS

Forrest Brown
X-3 Monte Carlo Codes
Los Alamos National Laboratory
fbrown@lanl.gov

ABSTRACT

Monte Carlo criticality calculations use the power iteration method to determine the eigenvalue (k_{eff}) and eigenfunction (fission source distribution) of the fundamental mode. A recently proposed method for accelerating convergence of the Monte Carlo power iteration using Wielandt's method has been implemented in a test version of MCNP5. The method is shown to provide dramatic improvements in convergence rates and to greatly reduce the possibility of false convergence assessment. The method is effective and efficient, improving the Monte Carlo figure-of-merit for many problems. In addition, the method should eliminate most of the underprediction bias in confidence intervals for Monte Carlo criticality calculations.

Key Words: Wielandt method, Monte Carlo, criticality, MCNP

1. INTRODUCTION

Monte Carlo-based criticality calculations make use of the basic numerical method called power iteration [1,2]. Given a fission neutron source distribution and an estimate of k_{eff}, single-generation random walks are carried out for a "batch" of neutrons to estimate a new k_{eff} and source distribution. Iterations continue until both k_{eff} and the source distribution have converged. Upon convergence, tallies are started and iterations continued until statistical uncertainties become small enough.

The power iteration method may converge slowly for many criticality problems, especially those having a dominance ratio close to 1.0. Deterministic codes based on discrete-ordinates or diffusion theory methods also use the power method to determine k_{eff} and the fundamental mode source distribution, but apply acceleration methods to improve the convergence rate. Standard acceleration methods used in deterministic codes have not been successfully applied to Monte Carlo criticality calculations, however, due to difficulties in treating the statistical noise. Recently, a novel Monte Carlo technique was proposed [3] for applying Wielandt's method [1,2] to accelerate the convergence of Monte Carlo eigenvalue calculations. While the improvements in convergence rate reported in [3] were impressive, the increased computing costs appeared prohibitive. This paper presents the results of recent efforts to provide an efficient implementation of Wielandt's method into the MCNP5 Monte Carlo code [4]. Similar to [3], we have found that the method provides dramatic improvements in convergence rates and greatly reduces the possibility of false convergence assessment. The current implementation in MCNP5, however, is effective and efficient, improving the Monte Carlo figure-of-merit for some

problems and roughly matching the unaccelerated performance for other problems. In addition, the method has the potential to eliminate most of the underprediction bias in confidence intervals for Monte Carlo criticality calculations. This paper summarizes the recent work on implementing Wielandt's method and performing numerical tests of its effectiveness.

2. THEORY FOR WIELANDT'S METHOD

Wielandt's method, also called fractional iteration, was first applied to accelerate the solution of diffusion theory reactor calculations in the 1950s and 1960s. The method is analyzed in detail in [1,2], and is briefly described here. The k-eigenvalue transport equation in standard form is

$$\left[\Omega\cdot\nabla+\Sigma_T(\vec{r},E)\right]\Psi(\vec{r},E,\Omega)=\iint\Psi(\vec{r},E',\Omega')\Sigma_S(\vec{r},E'\to E,\Omega\cdot\Omega')d\Omega'dE' + \frac{1}{k_{eff}}\frac{\chi(E)}{4\pi}\iint\nu\Sigma_F(\vec{r},E')\Psi(\vec{r},E',\Omega')d\Omega'dE' \quad (1)$$

This can be written as

$$(\mathbf{L}+\mathbf{T})\Psi=\mathbf{S}\Psi+\frac{1}{k_{eff}}\mathbf{M}\Psi \quad (2)$$

Subtracting a fixed fission source from each side of the transport equation gives

$$(\mathbf{L}+\mathbf{T}-\frac{1}{k_e}\mathbf{M})\Psi=\mathbf{S}\Psi+\left(\frac{1}{k_{eff}}-\frac{1}{k_e}\right)\mathbf{M}\Psi \quad (3)$$

where k_e must be chosen to be greater than k_{eff}. The amazing insight from [3] which permits the method to be applied to standard Monte Carlo criticality calculations concerns the treatment of the additional fission terms on each side of Eq. (3): The $(1/k_e)\mathbf{M}\Psi$ term on the left side corresponds to fission sources which will be followed within the current iteration, while the same term on the right side corresponds to fission sources which will be banked for the next iteration. With this interpretation, a modified power iteration is obtained:

$$\Psi^{(n+1)}=\frac{1}{\tilde{k}^{(n)}}\tilde{\mathbf{F}}\Psi^{(n)},\qquad n=0,1,...,\qquad given\ \tilde{k}^{(0)}\ and\ \Psi^{(0)} \quad (4)$$

$$where\ \ \tilde{k}^{(n)}=\left[\frac{1}{k_{eff}^{(n)}}-\frac{1}{k_e}\right]^{-1}\qquad and\qquad \tilde{\mathbf{F}}=(\mathbf{L}+\mathbf{T}-\mathbf{S}-\frac{1}{k_e}\mathbf{M})^{-1}\mathbf{M}$$

While the eigenfunctions of the modified power iteration given by Eq. (4) are identical to those of standard power iteration, the eigenvalues are shifted and the convergence rate is increased. The dominance ratio of the modified power iteration is given by

$$\tilde{\rho}=\frac{k_e-k_0}{k_e-k_1}\cdot\rho \quad (5)$$

where $\rho=(k_1/k_0)$ is the dominance ratio for standard power iteration. The modified dominance ratio given by Eq. (5) is always less than ρ since $k_e>k_0>k_1$, so that Eq. (4) converges at a faster rate than standard power iteration.

3. IMPLEMENTATION IN MCNP5

For this work, implementing Wielandt's method into MCNP5 was done somewhat differently from [3] so that existing and efficient MCNP5 coding could be used for banking fission neutrons for the current generation and next generation. For conventional MCNP5 criticality calculations, the number of fission neutrons placed into the next-generation fission bank on a collision is given by

$$n_{next} = \left\lfloor wgt \cdot \frac{\nu\Sigma_F}{\Sigma_T} \cdot \frac{1}{k_{collision}^{(n-1)}} + \xi \right\rfloor \tag{6}$$

For Wielandt's method, a reduced number of fission neutrons are placed into the next-generation fission bank on each collision, and some additional fission neutrons are placed into the particle bank for the active neutron at each collision:

$$\tilde{n}_{next} = \left\lfloor wgt \cdot \frac{\nu\Sigma_F}{\Sigma_T} \cdot \frac{1}{\tilde{k}^{(n-1)}} + \xi \right\rfloor, \qquad \tilde{n}_{current} = \left\lfloor wgt \cdot \frac{\nu\Sigma_F}{\Sigma_T} \cdot \frac{1}{k_e} + \xi \right\rfloor \tag{7}$$

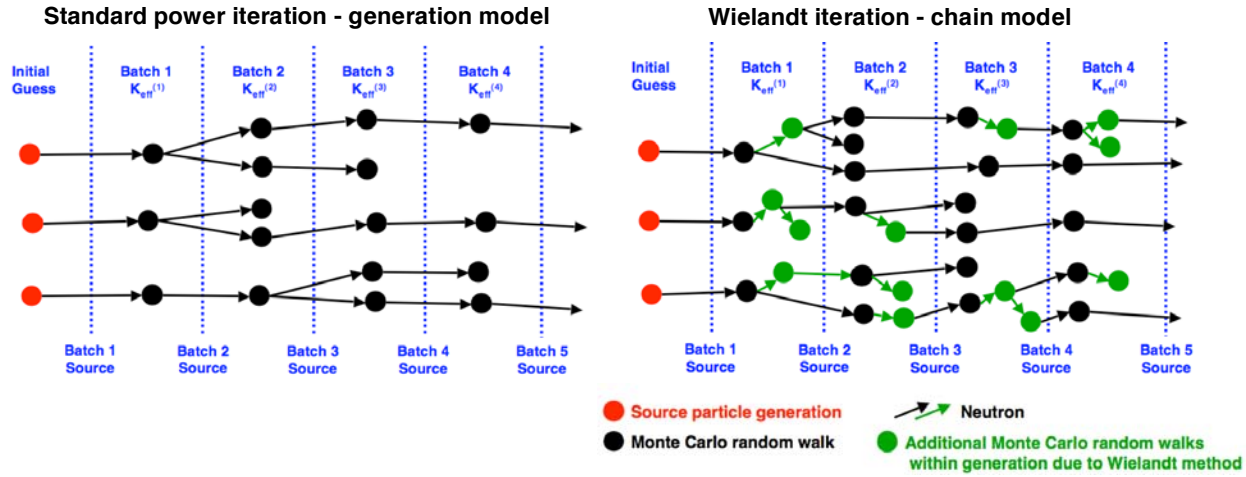
Note that $E(n_{next}) = E(\tilde{n}_{next}) + E(\tilde{n}_{current})$. With this approach, the fission neutrons for the current generation are banked away as secondary particles and then retrieved and followed as part of the current history. Thus, the existing banking mechanisms built into MCNP5 are used efficiently. Another important consideration is that the "chain" of current-generation neutrons followed in this manner will be treated as part of the same initial history in the iteration, so that correlation effects will be handled correctly between successive fissions. This improved statistical treatment should serve to reduce the underprediction bias in confidence intervals for criticality calculations.

The effect of using Eq. (7) for generating fission neutrons in both the current and next iterations, as opposed to using Eq. (6) to generate fission neutrons only in the next iteration, is shown schematically in Figure (1). The standard power iteration method corresponds to a "generation model," where there is a one-to-one correspondence between fission generations and power iterations. In Wielandt's method, each iteration contains (partial) fission chains, spanning more than one generation. There is a distribution of chain lengths, realized by sampling $n_{current}$ in Eq. (7) for successive collisions. The average length of these fission chains within an iteration can readily be shown to be

$$L \approx 1 + \frac{k_0}{\Delta}, \qquad where \;\; \Delta = k_e - k_0 \tag{8}$$

Thus for $k_0 \approx 1$, $L \approx 1$ for $\Delta = \infty$, $L \approx 2$ for $\Delta = 1$, $L \approx 11$ for $\Delta = 0.1$, $L \approx 101$ for $\Delta = 0.01$. As Δ approaches 0 (i.e., k_e approaches k_0), the chains within each iteration become longer, and there is correspondingly less correlation between different iterations. It is well-known that inter-iteration correlation causes an underestimation of the confidence intervals in Monte Carlo k-effective calculations [5]. The reduced correlation between iterations in Wielandt's method should eliminate much of the underprediction and result in more accurate (but larger) confidence intervals. (See Section 8 of [5] for discussion. Although the discussion in [5] was directed toward the superhistory method, it applies as well to Wielandt's method. In rough terms, the superhistory method uses a fixed number of generations per iteration, a fixed value of L, whereas

Figure 1. Comparison of standard power iteration & Wielandt iteration



Wielandt's method determines the number of generations for each chain stochastically according to Eqs. (7), with the average L given by Eq. (8).) In addition, longer fission chains within an iteration permit the source distribution to "spread out" more in a single iteration, leading to the faster convergence rate for Wielandt's method. The average computing cost for each iteration is roughly proportional to L, however, so that the faster convergence is offset by a higher computing cost per iteration.

The initial implementation of Wielandt's method into MCNP5 used a fixed value of k_e or Δ. Numerical testing for several problems showed that this approach often led to large fluctuations in the neutron population in the early stages of the iterations, especially for small values of Δ. To avoid this difficulty, Δ=1 was used for the first 2 iterations, and then Δ was reduced on subsequent iterations according to the ad hoc prescription in Table (1), until it reached a user-specified value. Following that, Δ was held constant at the user-specified value.

Table 1. Variation in Δ for the initial iterations, until Δ is reduced to user-specified value

Iteration	Δ	Iteration	Δ	Iteration	Δ
1	1.0	7	0.20	13	0.035
2	1.0	8	0.15	14	0.025
3	0.75	9	0.125	15	0.020
4	0.50	10	0.100	16	0.015
5	0.35	11	0.075	17	0.0125
6	0.25	12	0.050	≥18	0.0100

Using this scheme for varying Δ, there were no subsequent instabilities in the neutron population during the iterations.

Figure 2. Convergence of H_{src} vs iteration for 2D PWR test

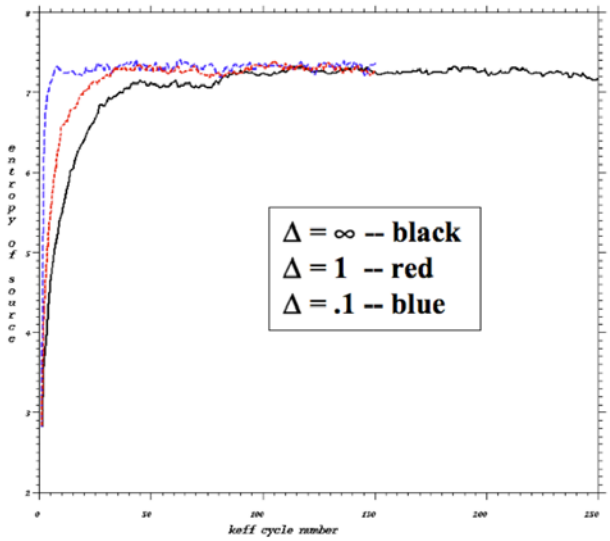
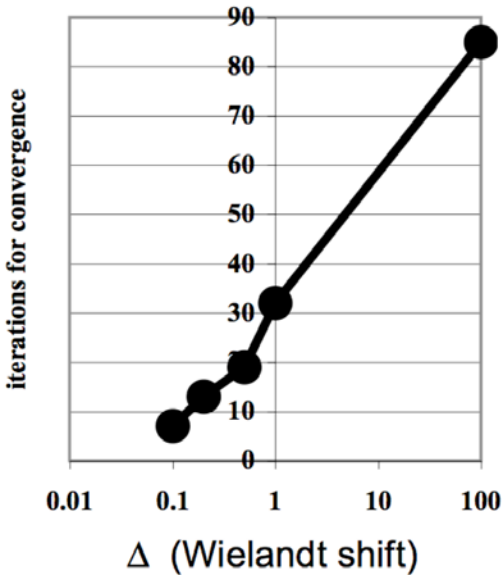


Figure 3. Iterations required for convergence, for different values of Δ



3. NUMERICAL RESULTS

Figure (2) shows the convergence of the Shannon entropy of the fission source distribution for a typical PWR reactor calculation in 2D for standard power iteration and for Wielandt's method using Δ=1 and Δ=0.1. A dramatic improvement in convergence can be seen for Δ=0.1, with the source distribution converging in about 5 iterations vs about 80 iterations for standard power iteration. Figure (3) shows the number of iterations required for source convergence for different choices of the shift parameter Δ for the same calculation. For this calculation, the Figure-Of-Merit ($FOM=1/(\sigma^2 T)$, where σ^2 is the variance in k_{eff} and T is the cpu-time) provided by MCNP5 for its final iterations was 168×10^3 for standard power iteration, 188×10^3 for Wielandt's method using Δ=1, and 184×10^3 using Δ=.1, indicating slightly improved code performance when using Wielandt's method.

A number of other test problems were calculated using Wielandt's method, including a 3D ¼ core PWR, 1D slab problems, and a large loosely-coupled fuel storage vault. The improvements in convergence in all cases were comparable to that for the problem discussed above. The FOM varied, showing improvement for some tests and moderate (~10-20%) degradation for others. The cause for the variation in FOM behavior is currently unknown and is the subject for continuing investigation.

4. WORK IN PROGRESS

While Wielandt's method appears to be very effective in accelerating the convergence of k_{eff} and the fission source distribution, there are many questions remaining before a robust, automated implementation is made available in a production version of MCNP5:

- The optimal choice for Δ is not known. Smaller values of Δ result in faster convergence, at the expense of increased computing cost per iteration.
- The scheme for varying Δ as a function of iteration number (cf. Table 1) works well for the problems tested, but may need modification for other cases.
- To date, numerical testing has focused on the convergence of k_{eff} and the Shannon entropy of the source distribution. Further examination into the convergence of local reaction rates (e.g., assembly or pin powers) is needed.
- Because Wielandt's method changes the total number of neutrons followed during an iteration, there are questions concerning the proper weighting of tally data. For example, should variance be based on the scores for fission chains in an iteration, or on the individual scores for each neutron?
- It was stated above and in [5] that following more than 1 fission generation in each iteration should reduce the inter-generation correlation and thus eliminate the underprediction bias in confidence intervals. It remains to be proven that these assertions are correct, and to determine the practical effects of computing accurate (but larger) confidence intervals.

All of these questions (and more) are under investigation. In addition, there is a larger question to be considered: If Wielandt's method is effective in accelerating the convergence of Monte Carlo criticality calculations, what other acceleration schemes might be adapted for use with Monte Carlo, using techniques similar to those described above?

Despite the questions noted here, it is expected that an initial implementation of Wielandt's method into the standard MCNP5 distribution will be made during 2007.

5. CONCLUSIONS

Wielandt's method has been implemented into a test version of the MCNP5 Monte Carlo code. Numerical testing has shown that the method provides dramatic improvements in convergence rates and greatly reduces the possibility of false convergence assessment. The current implementation in MCNP5 is effective and efficient, improving the Monte Carlo figure-of-merit for many problems, and roughly matching the unaccelerated performance for other problems. In addition, the method has the potential to eliminate most of the underprediction bias in confidence intervals for Monte Carlo criticality calculations.

ACKNOWLEDGMENTS

This work was performed in support of the US DOE Nuclear Criticality Safety Program.

REFERENCES

1. F. B. Brown, "Fundamentals of Monte Carlo Particle Transport", LA-UR-05-4983, Los Alamos National Laboratory, http://mcnp.lanl.gov/publication/mcnp_publications.html (2005).

2. S. Nakamura, *Computational Methods in Engineering and Science*, R.E. Krieger Pub. Company, Malabar, FL (1986).
3. T. Yamamoto and Y. Miyoshi, "Reliable Method for Fission Source Convergence of Monte Carlo Criticality Calculation with Wielandt's Method," *J. Nucl. Sci. Technol.* **41**, No. 2, 99-107 (2004).
4. X-5 Monte Carlo Team, "MCNP – A General N-Particle Transport Code, Version 5 – Volume I: Overview and Theory", LA-UR-03-1987, Los Alamos National Laboratory (April, 2003).
5. R.J. Brissenden and A.R. Garlick, "Biases in the Estimation of Keff and Its Error by Monte Carlo Methods," *Ann. Nucl. Energy*, Vol 13, No. 2, 63-83 (1986).
6. F. B. Brown, "On the Use of Shannon Entropy of the Fission Distribution for Assessing Convergence of Monte Carlo Criticality Calculations", *Proceedings PHYSOR-2006*, Vancouver, Canada, Sept 10-14 (2006).