LA-UR-12-00577

Title:	Evaluation of Computing c-Eigenvalues with Monte Carlo
Author(s):	Brian C. Kiedrowski
Intended for:	2012 ANS Annual Meeting June 24-28, 2012 Chicago, IL

Evaluation of Computing c-Eigenvalues with Monte Carlo

Brian C. Kiedrowski

Los Alamos National Laboratory, P.O. Box 1663 MS A143, Los Alamos, NM 87545,
bckiedro@lanl.gov

INTRODUCTION

Typically, criticality is determined by the computation of a system's k-eigenvalue that is the ratio of the production of fission neutrons to the losses from capture and leakage. Alternative eigenvalues exist, but are usually computed only for specialized applications. The second most commonly computed eigenvalue is the α-eigenvalue that measures how much of a $1/v$ (v is the neutron speed) absorber must be inserted to achieve criticality; this eigenvalue is often used quasi-static analysis of transients because it arises from the separation of time from the other dimensions of phase space.

Another eigenvalue, which is used for theoretical studies but very rarely used for applications, is the c-eigenvalue (sometimes this is referred to as the γ-eigenvalue) that is the ratio of production of secondaries from all sources (fission, scattering, n,xn) to losses. This eigenvalue was proposed by Davison [1], and has been studied theoretically by Sahni and others [2]. Ronen, et. al. [3] also showed for fast plutonium systems that the c-eigenvalue may be less sensitive to distortions in the energy spectrum when a system departs from criticality.

There is, perhaps, some motivation to study the c-eigenvalue for continuous energy. The ability to calculate of the c-eigenvalue is implemented in a research version of MCNP6 [4] and validated with some simple problems. Preliminary results show that the c-eigenvalue is often less variant and is sometimes computed more efficiently than the k-eigenvalue. Since the c- and k-eigenvalue equations are identical at criticality, the c-eigenvalue may be a more efficient quantity to use for criticality searches.

METHOD

Eigenvalue Equations

The c-eigenvalue relation is derived in a very similar fashion to that of the k-eigenvalue. The time derivative in the neutron transport equation is arbitrarily set to zero, and, to balance the equation, an artificial factor is applied somewhere. The formulation of the k-eigenvalue form of the neutron transport equation is

$$(L + T - S)\psi_k = \frac{1}{k} F\psi_k. \tag{1}$$

Here ψ_k is the neutron angular flux or k-eigenfunctions, and the operators are L for leakage, T for collision, S for emergence of neutrons from scattering or multiplicity reactions, and F for fission. The c-eigenvalue equation is very similar except that an artificial factor of $1/c$ is applied to both the scattering and fission terms as

$$(L + T)\psi_c = \frac{1}{c}(S + F)\psi_c. \tag{2}$$

By solving for k or c, physical meanings of the eigenvalues as being ratios of production from either fission (for k) or fission plus scatter (for c) to the losses from leakage and absorption. Like with k, the criticality condition for c is such that $c = 1$ denotes a critical system, $c < 1$ is subcritical, and $c > 1$ is supercritical. The off-critical values of k and c differ, as do their corresponding eigenfunctions – there is no known general relationship between the eigenfunctions. Note that, unlike k, c (like α) is defined for configurations with no fissile material.

Iterative Method Adaptation

Both the k- and c-eigenvalue equations have terms on the left- and right-hand sides of the equations that are treated differently in terms of an iterative scheme. The terms on the right-hand side are assumed to be known for the duration of the iteration, and a solution to the function ψ on the left-hand side is desired. Mathematically, the c-eigenvalue equation for an iteration (denoted by superscript (n)) is

$$(L + T)\psi_c^{(n+1)} = \frac{1}{c^{(n)}}(S + F)\psi_c^{(n)}. \tag{3}$$

Formally, the operators of the equation are inverted such that a new value of the eigenfunction is obtained. This "inversion" can be done via solving the equation via numerical techniques or direct simulation by Monte Carlo. For the latter, the terms on the right-hand side define the source neutrons for the iteration and the terms on the left are simulated via particle transport. To compute a new estimate of the right-hand side, tallies or estimators are employed during the particle transport.

In the familiar k-eigenvalue equation, neutron streaming (L), collisions (T), and scattering (S) are simulated via transport with particles obtained from the fission source. A new estimate of k and the fission source are obtained by some combination of track-length or reaction-rate estimators; in the case of MCNP, k is obtained by a combination

of track-length, collision, and absorption estimators and the fission source is obtained via a collision estimator of fission neutron production.

The analogous simulation for the c-eigenvalue is a straightforward modification. Within an iteration, the only physics simulated is streaming and collisions. The source now consists of neutrons that arise from both scattering and fission, which is obtained from estimators. The reaction-rate multiplier for the estimators is the macroscopic secondary-production "cross section" Σ_P that is defined as

$$\Sigma_P = \nu\Sigma_f + \sum_{x=1}^{\infty} x\Sigma_{n,xn}, \tag{4}$$

where ν is the average number of neutrons per fission, Σ_f is the macroscopic fission cross section, x is the multiplicity of the reaction, and $\Sigma_{n,xn}$ is the macroscopic multiplicity reaction cross section where $x = 1$ is the sum of the elastic and inelastic scatter cross sections. For the moment, only collision estimators are used for the estimators of c and the secondary-neutron source.

Algorithm Description

The algorithm for computing c is very similar to that of computing k. Prior to the calculation, the user must specify an initial guess for c, define an initial source, and provide the total weight of each iteration M. M source neutrons are generated from this source guess with weight w of unity, and neutrons are followed until they either leak out of the system or have a collision. When a neutron has a collision, Σ_P is calculated, and an estimate for secondary production P is made:

$$P = \frac{w\Sigma_P}{\Sigma_t}, \tag{5}$$

where Σ_t is the total cross section for the material. The estimate of the eigenfunction is done by banking (the position, energy, and direction arising from the collision are recorded) some number of neutrons B given by

$$B = \left\lfloor \frac{P}{c} + \xi \right\rfloor, \tag{6}$$

where the c used is either the initial guess or the estimate from the previous iteration and ξ is a uniform random variable from zero to one. The nuclide for computing outgoing energies and directions is selected based on a histogram of ratios of Σ_P for each isotope to the total Σ_P for the material, and collision mechanics is performed according to randomly selected nuclear reactions. Once all histories within the iteration are run, the banked secondary neutrons become the new source for the next iteration. The statistical weight w is adjusted such that each particle has the same weight and the total weight sums to M.

With the new source, the process repeats until some user defined cutoff is reached. Like with k-eigenvalue calculations, since the secondary-production source is typically unknown, some iterations must be spent reaching a converged secondary-production source. Once the converged source is reached, since all that is available is a random realization of it, additional iterations must be done to converge mean values of any desired responses such as c-eigenvalue. In the MCNP parlance, the iterations for source convergence are referred to as inactive cycles, and the iterations for statistical convergence of responses are the active cycles.

VERIFICATION & TESTING

Analytic Verification

To show that the method and implementation in MCNP is correct, c is computed for very simple systems where an analytic result may be obtained. Test case 1 is a critical ($k = c = 1$) bare sphere of one-group plutonium-239, which is test problem 8 in Ref. [5]. Test case 2 is an two-group infinite medium problem with cross sections given in Table I and all neutrons are produced in group 1. The value for $c = (7 + \sqrt{145})/24 \approx 0.79340$). Results from MCNP agree to five decimal digits of accuracy.

Table I. Cross sections (cm^{-1}) for test case 2.

g	Σ_t	Σ_c	Σ_f	ν	$\Sigma_{g\to1}$	$\Sigma_{g\to2}$
1	4	1	0	–	1	2
2	6	3	1	3	0	2

Eigenvalue Convergence

The eigenvalue trend must be converged before it may be tallied. Convergence of the eigenfunction must also be completed before any other responses may be tallied, but for now, the attention is solely on the eigenvalue. The method for c differs from k in that only one path to collision is sampled per history, as opposed to following the neutron over potentially many collisions until it causes fission or is removed by leakage or capture. Because less transport is done per iteration, convergence of c is expected to require more iterations than for k.

To test convergence, test problem consisting of a 3×2 array of cans of plutonium-nitrate solution from Sec. 5.2 of the MCNP Criticality Primer [6] is used. Source neutrons begin in the center of one of the corner cans at an energy of 1 MeV, which is not a very good initial guess, but it makes convergence trends more apparent. The c-eigenvalue as a function of iteration with a curve fit is given in Fig. 1, and

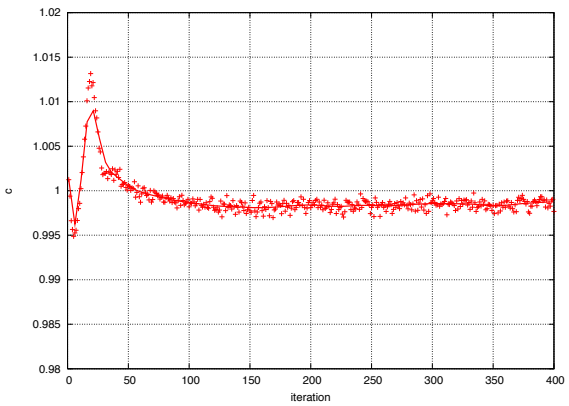

Fig. 1. Convergence of c-eigenvalue for the array of plutonium nitrate solution cans.

it shows that about 150 iterations are required for convergence of the trend. The convergence in k for an equivalent calculation takes about 20 iterations. In terms of wall-clock time, it takes 4.7 minutes for convergence in c and 1.6 minutes for convergence in k (for $M = 10{,}000$). Placing a source point in each of the six cans (energy still 1 MeV) halves the time for convergence of the eigenvalues. This relatively large difference in time required suggests that c is more sensitive to spatial and spectral departures from the fundamental mode than k, and a good starting guess in both space and energy for a c-eigenvalue calculation is more important for efficient convergence.

The value of c computed for this system using 300 active iterations with $M = 100{,}000$ is 0.99852(3); the value of k with the same number of iterations and M is 0.98792(59). As expected c and k are different; however, the statistical uncertainty for the same number of histories in k is about twenty times higher than that of c. This finding is discussed in greater detail later.

Neutrons in Thermal Equilibrium

The c-eigenvalue, unlike k, is defined for systems with no fission. The steady eigenfunction solution for ψ_c is a system of neutrons in thermal equilibrium with its medium – neutrons that are lost are replaced on average by the amplification factor of $1/c$ each collision. For continuous-energy Monte Carlo, the effect of upscatter is captured via either free-gas or $S(\alpha, \beta)$ scattering laws in the collision mechanics.

A test of this feature in a c-eigenvalue calculation is performed by evaluating a system with no fissionable material. The test problem used is a sphere of water at 1.0 g/cc with a diameter of 5 cm. Two cases are run: the first is with only free-gas scattering, and the second includes the $S(\alpha, \beta)$ scattering law for hydrogen bonded to a water molecule.

The value of c for the free-gas case is 0.84430(6), and is 0.94568(3) with $S(\alpha, \beta)$. This makes sense as the effect of $S(\alpha, \beta)$ increases the elastic scattering cross section of hydrogen, so more secondary neutrons should be produced relative to the loss rate of neutrons from leakage or capture. The spectra for both cases show a stable Maxwellian shape; had no upscattering treatment been included, the neutrons would downscatter to zero energy.

EIGENVALUE CALCULATION EFFICIENCY

In the convergence test case for a 3×2 array of cans of plutonium nitrate solution, the c-eigenvalue estimation is less variant than that for the k-eigenvalue. A standard metric for assessing efficiency of a Monte Carlo calculation is the Figure of Merit:

$$FOM = \frac{1}{R^2 \tau}. \tag{7}$$

Here R is the relative uncertainty of the response and τ is the computation time during the active cycles. The drawback of the FOM is that it does not take into account time spent in the inactive iterations, which is an added cost onto a calculation that will generally be higher for c than will be for k. Nonetheless, this serves as a crude way to compare the efficiency of computing the eigenvalues. Since the values of the FOM depend on the speed of the computer used to run the problem, the ratio of the FOMs for c and k is perhaps more relevant; this ratio is called the gain G.

The test cases used are (1) a beryllium-reflected sphere of highly-enriched uranium [7], (2) the 3×2 array of cans discussed previously, (3) and a 3-D full core pressurized water reactor (PWR) [8]. For each case, the following is computed: k, c, the wall-clock time (in minutes) of inactive cycles required for each eigenvalue (W_k and W_c respectively), and the gain G; these results are given in Table II. All calculations are run sequentially to ensure consistent timing, use $M = 10{,}000$ and 200 active iterations. Initial source guesses are chosen in accordance with the best practices for criticality calculations [9].

In all cases, the gain G is greater than one, implying that during the active iterations c is estimated more efficiently than k. This is not to say, however, that it is necessarily faster to compute c because, in all cases, it takes

Table II. Performance data for three test cases.

case	k	W_k	c	W_c	G
1	0.9955(4)	0.3	0.9954(3)	0.8	3.1
2	0.9866(7)	0.8	0.9989(1)	2.4	15.2
3	0.9992(5)	3.4	0.9986(1)	7.4	51.4

more wall-clock time to converge the trend in the eigenvalue before active iterations may even start. It appears that the biggest gains for using c are for systems with a significant amount of scattering, where the PWR (case 3) has the largest gains of the three. This suggests the possibility that, if the convergence of test case 3 is typical of PWRs, the c-eigenvalue may be a possible alternative for criticality searches for reactor applications. Note that, as for the eigenvalues themselves, cases 1 and 3 appear to have matching k and c, whereas case 2 has a difference of over 0.01.

CONCLUSIONS & OUTLOOK

The c-eigenvalue can be computed via Monte Carlo using a straightforward modification of the k-eigenvalue iteration procedure. The MCNP6 implementation produces results that match analytic solutions for multigroup and expected behavior for continuous energy. The c-eigenvalue typically takes longer to converge than the k-eigenvalue, but, once converged, it can be evaluated more efficiently (for the test problems presented). The case with the greatest gain in efficiency is a 3-D, full core PWR, which suggests the c-eigenvalue may be useful for doing criticality searches on these types of systems.

Issues still remain open. First, it is not entirely clear what applications the c-eigenvalue is useful for, and it is likely k will remain the quantity of interest for fields such as criticality safety, since k deals directly with the addition and removal of fissile material, reflectors, etc. Nonetheless, since both are the same at criticality, there may be advantages to employing c over k. For instance, claims by Ronen, et. al, about greater spectral preservation away from criticality warrants further investigation, as it may be possible to obtain more accurate estimates of the neutron spectrum for the critical configuration despite the fact that the model indicates the system is not critical. Furthermore, the issue of convergence of the c-eigenfunction and how it may be assessed or accelerated remains an open topic, especially considering energy (and possibly directional) dependence seems important.

ACKNOWLEDGMENTS

Funding for this work was provided by the US DOE/NNSA Nuclear Criticality Safety Program.

REFERENCES

1. B. DAVISON, *Neutron Transport Theory*, Oxford University Press, London, UK (1958).
2. D. C. SAHNI, S. J. SJÖSTRAND, "Criticality and Time Eigenvalues in One-Speed Neutron Transport", *Prog. Nucl. Engry.*, **23**, 241-289 (1990).
3. Y. RONEN, D. SHVARTS, J. J. WAGSCHAL "A Comparison of Some Eigenvalues in Reactor Theory", *Nucl. Sci. Eng.*, **60**, 97-101 (1976).
4. X-5 MONTE CARLO TEAM, "MCNP – A General Monte Carlo N-Particle Transport Code, Volume I: Overview and Theory", LA-UR-03-1987, Los Alamos National Laboratory (2003).
5. A. SOOD, R. A. FORSTER, D. K. PARSONS, "Analytical Benchmark Test Set for Criticality Code Verification", *Prog. Nucl. Engry.*, **42**, 55-106 (2003).
6. R. W. BREWER (Ed.), "Criticality Calculations with MCNP5: A Primer", Los Alamos National Laboratory Report, LA-UR-09-00380 (2009).
7. J. B. BRIGGS (Ed.), "International Handbook of Evaluated Criticality Safety Benchmark Experiments", heu-met-fast-009, Nuclear Energy Agency, NEA/NSC/DOC(95)03/I, Paris, France (2004).
8. J. E. HOOGENBOOM, W. R. MARTIN, B. PETROVIC, "Monte Carlo Performance Benchmark for Detailed Power Density Calculation in a Full Size Reactor Core", OECD/NEA, (2009).
9. F. B. BROWN, "A Review of Best Practices for Monte Carlo Criticality Calculations", *Proc. 2009 ANS Nuclear Criticality Safety Division Topical Mtg.*, Richland, WA, Sept. 13-17 (2003).