NE 255, Fa16
Eigenvalue Formulation and Solutions
October 27, 2016

We're going to start talking about how to find the criticality state of a reactor, or the dominant eigenvalue-eigenvector pair. To turn the steady state Boltzmann transport equation with fission into an eigenvalue problem, we can use a mathematical "knob". There are two ways we do this:

1. Alter the effective cross section by adding an α term (what we have been doing), or
2. Alter the effective fission yield, ν, by scaling it with k

The α version, which is used more frequently at LLNL and LANL, looks like this:

$$\Big[\hat{\Omega}\cdot\nabla + \big(\Sigma_t + \underbrace{\frac{\alpha_0}{v}}_{\text{new}}\big)\Big]\psi(\vec{r},E,\hat{\Omega}) = \int_{4\pi} d\hat{\Omega}' \int_0^\infty dE'\,\Sigma_s(E',\hat{\Omega}'\to E,\hat{\Omega})\psi(\vec{r},E',\hat{\Omega}') + \frac{\chi(E)}{4\pi}\int_0^\infty dE'\,\nu(E')\Sigma_f(E')\int_{4\pi} d\hat{\Omega}'\,\psi(\vec{r},E',\hat{\Omega}')$$

+ homogeneous boundary conditions.
Some important notes about this

- In the α-evaluation problem, the total cross section is modified by a 1/v absorber.
- There is numerical difficulty if $\alpha_0 < 0$.
  If $\frac{-\alpha_0}{v} > \Sigma_t$ then total interaction becomes a "source" rather than a loss!
- If $\alpha_0 > 0$, absorption of **slow** neutrons is enhanced by an α_0/v absorber $\rightarrow$ harder spectrum.
- If $\alpha_0 < 0$, absorption of **fast** neutrons is enhanced by an α_0/v absorber $\rightarrow$ softer spectrum.
- These spectral effects are important because reaction rate is $\int_0^\infty dE\,\Sigma_j(E)\phi(E)$

The k-eigenvalue problem is a little bit different:

$$\big[\hat{\Omega}\cdot\nabla + \Sigma_t\big]\psi(\vec{r},E,\hat{\Omega}) = \int_{4\pi} d\hat{\Omega}' \int_0^\infty dE'\,\Sigma_s(E',\hat{\Omega}'\to E,\hat{\Omega})\psi(\vec{r},E',\hat{\Omega}') + \underbrace{\frac{1}{k}}_{\text{new}}\frac{\chi(E)}{4\pi}\int_0^\infty dE'\,\nu(E')\Sigma_f(E')\int_{4\pi} d\hat{\Omega}'\,\psi(\vec{r},E',\hat{\Omega}')$$

+ homogeneous boundary conditions.
Important notes about this version:

- we can see that $k = 1$ means $\alpha_0 = 0$ and vice versa.
- k as a multiplication factor: the ratio of neutron production in one generation to the neutron production in the previous generation.
- We often use the k version because we don't have the mathematical difficulties associated with the α form, and the physical interpretation is helpful.

Most material derived from my thesis, with support form Wolfram Mathworld and other sources (noted in tex comments).

Background

A right eigenvector is defined as a column vector x_R satisfying

$$\mathbf{A}x_R = \lambda_R x_R \, .$$

In many common applications, only right eigenvectors (and not left eigenvectors) need be considered. Hence the unqualified term "eigenvector" can be understood to refer to a right eigenvector.

A left eigenvector is defined as a row vector x_L satisfying

$$x_L\mathbf{A} = \lambda_L x_L \, .$$

The **generalized eigenvalue problem** takes the form $\mathbf{B}x = \mu\mathbf{C}x$ and can be transformed into an **ordinary eigenvalue problem**, $\mathbf{A}x = \lambda x$. Both forms have the same right eigenvectors. If $\mathbf{C}$ is non-singular then $\mathbf{A} = \mathbf{C}^{-1}\mathbf{B}$ and the problem $v = \mathbf{A}x$ can be solved in two steps:

1. $w = \mathbf{B}x$
2. Solve the system $\mathbf{C}v = w$.

Because the generalized form can be converted to the ordinary form, we will focus on the more common ordinary form without loss of applicability.

Recall this basic notation: let $\sigma(\mathbf{A}) \equiv \{\lambda \in \mathbb{C} : rank(\mathbf{A} - \lambda\mathbf{I}) \leq n\}$ be the spectrum of $\mathbf{A}$,

where the elements in the set are the eigenvalues and $\mathbb{C}$ is the set of complex numbers. The eigenvalues can be characterized as the n roots of the polynomial $p_{\mathbf{A}}(\lambda) \equiv det(\lambda \mathbf{I} - \mathbf{A})$. Each distinct eigenvalue in $\sigma(\mathbf{A})$ has a corresponding nonzero vector x such that $\mathbf{A}x_i = \lambda_i x_i$ for $i = 1, ..., n$. It will be assumed that the eigenvalues are ordered as $|\lambda_1| > |\lambda_2| \geq \cdots \geq |\lambda_n| \geq 0$.

Eigenvalue problems in the nuclear transport community are typically solved with iterative rather than direct methods. A variety of iterative solvers have been used to solve eigenvalue problems. These are ones that have been most widely used.

Power Iteration

Power iteration (PI) is an old and straightforward algorithm for finding an eigenvalue/vector pair. The basic idea is that any nonzero vector can be written as a linear combination of the eigenvectors of $\mathbf{A}$ because the eigenvectors are linearly independent, namely $v_0 = \gamma_1 x_1 + \gamma_2 x_2 + \cdots + \gamma_n x_n$ where x_j is the jth eigenvector and γ_j is some constant. This specific expression assumes a non-defective $\mathbf{A}$, though this assumption is not necessary for the method to work.

Another fact that is used to understand power iteration is that $\mathbf{A}^k x_i = \lambda_i^k x_i$. Thus

$$\mathbf{A}^k v_0 = \gamma_1 \lambda_1^k x_1 + \gamma_2 \lambda_2^k x_2 + \cdots + \gamma_n \lambda_n^k x_n \, . \tag{1}$$

Since $|\lambda_1| > |\lambda_i|, i \neq 1$, the first term in the expansion will dominate as $k \to \infty$ and $\mathbf{A}^k v_0$ therefore becomes an increasingly accurate approximation to x_1. In practice, it is desirable to avoid exponentiating a matrix and it is helpful to normalize v in order to avoid possible over or underflow. This leads to the power iteration algorithm:

Given $\mathbf{A}$ and v_0, $v = \frac{v_0}{||v_0||}$.

Until convergence do:

$w = \mathbf{A}v$

$v = \frac{w}{||w||}$

$\lambda = v^T \mathbf{A} v$

Using Equation (1) and the PI algorithm, PI's convergence behavior can be understood. After k

steps, the iteration vector will be:

$$v_k = \big(\frac{\lambda_1^k}{e_1^T \mathbf{A}^k v_0}\big)\big(\frac{1}{\lambda_1^k}\mathbf{A}^k v_0\big)\,. \tag{2}$$

If $\mathbf{A}$ has eigenpairs $\{(x_j, \lambda_j), 1 \leq j \leq n\}$ and v_0 has the expansion $v_0 = \sum_{j=1}^n x_j \gamma_j$ then

$$\frac{1}{\lambda_1^k}\mathbf{A}^k v_0 = \frac{1}{\lambda_1^k}\sum_{j=1}^n \mathbf{A}^k x_j \gamma_j = \sum_{j=1}^n x_j \big(\frac{\lambda_j}{\lambda_1}\big)\gamma_j\,. \tag{3}$$

From equation (3) it can be determined that the error is reduced in each iteration by a factor of $|\frac{\lambda_2}{\lambda_1}|$, which is called the dominance ratio. If λ_2 is close to λ_1, then this ratio will be close to unity and the method will converge very slowly. If λ_2 is far from λ_1, then convergence will happen much more quickly. Put simply, PI is better suited for problems where $\mathbf{A}$ has eigenvalues that are well separated.

Power iteration is very attractive because it only requires matrix-vector products and two vectors of storage space. Because of its simplicity and low storage cost, PI has been widely used in the transport community for criticality problems for quite some time. However, because of the slow convergence for many problems of interest, many current codes use an acceleration method with PI or have moved away from it altogether.

As an aside, it is interesting to point out the connection between Krylov methods and power iteration. The power method is a Krylov method that uses a subspace of dimension one. A Krylov subspace is built by storing the vectors created during each iteration of the power method. Krylov methods with subspaces larger than one take advantage of the information generated during each iteration that the power method discards.

Power Iteration in the TE

Recall the operator form of the steady state, eigenvalue transport equation:

$$\mathbf{L}\psi = \mathbf{MS}\phi + \frac{1}{k}\mathbf{MF}\phi\,.$$

This equation can be combined with $\phi = \mathbf{D}\psi$ and manipulated in the following ways:

$$\phi = \mathbf{DL}^{-1}\mathbf{MS}\phi + \mathbf{DL}^{-1}\mathbf{M}\frac{1}{k}\chi\mathbf{f}^T\phi\,. \tag{4}$$

Let $\mathbf{T} = \mathbf{D}\mathbf{L}^{-1}$, rearrange, and multiply both sides by f^T:

$$f^T\big(\mathbf{I} - \mathbf{TMS}\big)\phi = \frac{f^T}{k}\mathbf{TM}\chi f^T\phi\,, \tag{5}$$

$$\text{define } \Gamma = f^T\phi \text{ and rearrange ,}$$

$$k\Gamma = \underbrace{f^T(\mathbf{I} - \mathbf{TMS})^{-1}\mathbf{TM}\chi}_{\mathbf{A}}\,\Gamma\,. \tag{6}$$

All of that can be used to make the traditional form of power iteration where the eigenvalue iteration index is k:

$$\Gamma^{k+1} = \frac{1}{k^k}\mathbf{A}\Gamma^k\,. \tag{7}$$

Note that Equation (6) is the ordinary form of the eigenvalue transport equation. In this case the eigenvalue-vector pair are (Γ, k).

The application of $\mathbf{A}$ to Γ^k involves a multigroup solve that looks like a fixed source problem:

$$\big(\mathbf{I} - \mathbf{TMS}\big)y^k = \mathbf{TM}\chi\Gamma^k\,. \tag{8}$$

Thus, inside of one eigenvalue iteration we will have done inner, space-angle solves for each group and an outer iteration over all groups.

Shifted Inverse Iteration

One way to improve power iteration is to shift the matrix $\mathbf{A}$ and then, in a power iteration-type scheme, apply the inverse of the shifted matrix rather than the regular unshifted matrix. The method is called inverse iteration or shifted inverse iteration and the goal is to provide better convergence properties. Understanding why shifted inverse iteration is an improvement requires understanding spectral transformation.

The fundamental notion is that $\mathbf{A}$ can be shifted by a scalar without changing the eigenvectors. That is, for some shift μ, $(\mathbf{A} - \mu\mathbf{I})$ will have the same eigenvectors as $\mathbf{A}$. If $\mu \notin \sigma(\mathbf{A})$, then $(\mathbf{A} - \mu\mathbf{I})$ is invertible and $\sigma([\mathbf{A} - \mu\mathbf{I}]^{-1}) = \{1/(\lambda - \mu) : \lambda \in \sigma(\mathbf{A})\}$. Eigenvalues of $\mathbf{A}$ that are near the shift will be transformed to extremal eigenvalues that are well separated from the others. Such a spectral transformation can be added to the power method. Given an estimate, $\mu \approx \lambda_1$, the shifted inverse power method will usually converge more quickly than PI.

To see why this works consider:

$$\tau_1 = \frac{1}{\lambda_1 - \mu}\,, \tau_2 = \frac{1}{\lambda_2 - \mu}\,, \ \ldots\ , \tau_n = \frac{1}{\lambda_n - \mu}\,. \tag{9}$$

As $\mu \to \lambda_1$, $\tau_1 \to \infty$ and all the other eigenvalues go to finite quantities. If the τs are inserted into the convergence analysis done for PI, it can be shown that the shifted inverse power method will reduce error in every iteration by a factor of $\frac{|\lambda_1 - \mu|}{|\lambda_2 - \mu|}$. This is typically much faster than $|\frac{\lambda_2}{\lambda_1}|$, though the ultimate success of the method is dependent upon the quality of the shift.

The algorithm for shifted inverse iteration is much like that for power iteration, requiring only one change. Step 1 becomes "solve $(\mathbf{A} - \mu\mathbf{I})w = v$;" all other steps are the same. After convergence, the actual eigenvalue of interest is backed out using the eigenvector. Shifted inverse iteration is effectively the same as performing power iteration on $(\mathbf{A} - \mu\mathbf{I})^{-1}$.

Aside:
Conditioning is one way to express the perturbation behavior of the mathematical problem. A *well-conditioned* problem is one in which all small perturbations of x lead to only small changes in $f(x)$. An *ill-conditioned* problem is one in which some small perturbation of x leads to a large change in $f(x)$. The condition number is a quantity used to express how well-conditioned a matrix or problem is. A small condition number corresponds to a well-conditioned problem, and vice versa.

The relative condition number is used to measure the effect of relative perturbations. This is particularly useful in numerical analysis because computers introduce relative errors. Let δx be a small perturbation of x and $\delta f = f(x + \delta x) - f(x)$. With these terms, the relative condition number for some norm p is defined as

$$\kappa(x)_p = \lim_{\delta \to 0} \sup_{||\delta x||_p \leq \delta} \left(\frac{||\delta f||_p}{||f(x)||_p} / \frac{||\delta x||_p}{||x||_p} \right), \tag{10}$$

where $\sup$ is the supremum, the smallest real number that is greater than or equal to every number in the set in question. This definition holds for any norm. The norm subscript will excluded unless specifying a certain norm is pertinent.

The condition number of a matrix $\mathbf{A}$ is defined as

$$\kappa(\mathbf{A}) = ||\mathbf{A}||\ ||\mathbf{A}^{-1}||\,. \tag{11}$$

For an $n \times m$ matrix $\mathbf{A}$, the singular values (a generalization of eigenvalues for non-square matrices) are ordered such that $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n > 0$. If the two-norm is used, then $||\mathbf{A}||_2 = \sigma_1$, $||\mathbf{A}^{-1}||_2 = \frac{1}{\sigma_m}$, and $\kappa_2(\mathbf{A}) = \frac{\sigma_1}{\sigma_m}$; σ_m is the mth singular value of $\mathbf{A}$. If $\mathbf{A}$ is singular, its condition number is infinity.

On initial consideration it would seem shifted inverse methods might not work well when the shift is very good because the matrix becomes very ill-conditioned. If the shift is exact, i.e. when $\mu = \lambda_1$, the matrix is singular. It turns out this concern is unfounded. Peters and Wilkinson proved that ill-conditioning is not a problem for inverse iteration methods. Trefethen and Bau also assert that this is the case as long as the $(\mathbf{A} - \mu\mathbf{I})w = v$ portion is solved with a backwards stable algorithm.

Wielandt's method is a flavor of shifted inverse iteration that has been used widely in the neutron transport community, though it was originally developed in 1944 in an unrelated context. In the transport equation formulation, Wielandt's method changes the generalized form of the eigenvalue problem to $(\mathbf{I}-\mathbf{D}\mathbf{L}^{-1}\mathbf{M}(\mathbf{S}+\gamma_e\mathbf{F}))\phi = (\delta\gamma)\mathbf{D}\mathbf{L}^{-1}\mathbf{M}\mathbf{F}\phi$ where γ_e is an estimate for the eigenvalue $\gamma_1 = \frac{1}{k}$, ϕ is the corresponding eigenvector, and $(\delta\gamma) = \gamma_1 - \gamma_e$. The power method is applied to this, giving.

$$\phi^{i+1} = (\delta\gamma)^i(\mathbf{I} - \mathbf{D}\mathbf{L}^{-1}\mathbf{M}(\mathbf{S} + \gamma_e\mathbf{F}))^{-1}\mathbf{D}\mathbf{L}^{-1}\mathbf{M}\mathbf{F}\phi^i \,. \tag{12}$$

The estimate, γ_e, can be improved on each iteration by starting with an initial guess of 0 and setting $\gamma_e^i = (\delta\gamma)^{i-1} + \gamma_e^{i-1} + \alpha$. Here α is an optional small positive constant that is designed to keep roundoff error from dominating when γ_e becomes very close to γ_1.

Studies of reactor systems have found shifted inverse iteration to be faster than power iteration. In the general computational community, shifted inverse iteration has largely taken the place of power iteration when looking for eigenvectors associated with eigenvalues that are relatively well known at the outset of the calculation since this information allows for the selection of a good shift.

Rayleigh Quotient Iteration

Rayleigh quotient iteration is a variation of shifted inverse iteration that has a changing shift like Wielandt's method sometimes does. The key difference is that this method uses a specific formula, the Rayleigh quotient (RQ), that is designed to find the optimal the shift at every iteration.

The Rayleigh quotient for the ordinary eigenvalue problem, originally proposed by the third Baron

Rayleigh in the 1870s, is defined as:

$$\rho(x, \mathbf{A}) = \rho(x) = \frac{x^T \mathbf{A} x}{x^T x} \,. \tag{13}$$

If x is an eigenvector of $\mathbf{A}$, then the RQ is the corresponding eigenvalue. If x is close to an eigenvector, then the RQ will approximate the eigenvalue. The derivation of the RQ comes from asking the question "what α will minimize $||\mathbf{A}x - \alpha x||_2$?" Solving this using least squares will give $\alpha = \rho(x)$.

The RQ has a similar form for the generalized eigenvalue problem, mentioned here because it is the form that was implemented in Denovo. For the problem $\mathbf{A}x = \lambda \mathbf{B}x$, there is a right eigenpair (λ, x) for $x \neq 0$ and a left eigenpair $(\lambda, y) : y^T \mathbf{A} = \lambda y^T \mathbf{B}$ for $y \neq 0$. Let $\langle \alpha, \beta \rangle$ be a simple eigenvalue of pencil $(\mathbf{A}, \mathbf{B})$. If x and y are right and left eigenvectors corresponding to $\langle \alpha, \beta \rangle$, respectively, then $\langle \alpha, \beta \rangle = \langle y^T \mathbf{A} x, y^T \mathbf{B} x \rangle$. This means the ordinary form of the eigenvalue is:

$$\lambda = \frac{y^T \mathbf{A} x}{y^T \mathbf{B} x} \,, \tag{14}$$

which is the generalization of the RQ.

Recall from the previous section that choosing a shift close to the eigenvalue of interest controls the dominance ratio of shifted inverse iteration, and hence convergence behavior. The RQI algorithm uses a strategically selected shift, the Rayleigh quotient, as seen here.

Given $\mathbf{A}$ and v_0, $v = \frac{v_0}{||v_0||}$, and $\rho_0 = \rho(v) = v^T \mathbf{A} v$

Until convergence do:

Solve $(\mathbf{A} - \rho \mathbf{I}) w = v$

normalize $v = \frac{w}{||w||}$

form $\rho = v^T \mathbf{A} v$

This process generates a sequence, $\{\rho_k, v_k\}$, called the Rayleigh sequence generated by v_0 on $\mathbf{A}$. To more deeply understand why this method is optimal and useful for the purposes of this work, more properties of the Rayleigh quotient are highlighted:

- For $\alpha \neq 0$, $\rho(\alpha x, \mathbf{A}) = \rho(x, \mathbf{A})$, so using any multiple of x will produce the same sequence as x.

- The RQ has translational invariance, $\rho(x, \mathbf{A} - \alpha\mathbf{I}) = \rho(x, \mathbf{A}) - \alpha$, meaning the matrix $(\mathbf{A} - \alpha\mathbf{I})$ produces the same sequence as $\mathbf{A}$. This relationship can be used to relate eigenvalues and applied shifts. In fact, this is one of the ways to find the eigenvalue of interest for the standard shifted inverse iteration method.
- When x is an eigenvector of $\mathbf{A}$, $\rho(x)$ is stationary at x.
- When $x \neq 0$ the RQ gives the minimal residual, with equivalence holding only when $\beta = \rho(x)$:
$$||(\mathbf{A} - \beta\mathbf{I})x||^2 \geq ||\mathbf{A}x||^2 - ||\rho(x)x||^2 . \tag{15}$$
- x is orthogonal to $(\mathbf{A} - \rho(x))x$.

RQI has very good convergence properties for normal matrices. The minimal residual property of the RQ causes the global convergence behavior. The sequence of residuals $\{||(\mathbf{A} - \rho_k)v_k|| = ||r_k||, k = 0, 1, ...\}$ is monotonically decreasing for all starting v_0. When RQI is applied to normal matrices, the following has been proven as $k \to \infty$:

1. $\rho_k = \rho(v_k)$ converges, and either
2. (ρ_k, v_k) converges cubically to an eigenpair (λ, k), or
3. ρ_k converges linearly to a point equidistant from $s \geq 2$ eigenvalues of $\mathbf{A}$, and the sequence $\{v_k\}$ cannot converge.

This means that when RQI converges to the correct eigenpair it does so rapidly. However, there is a risk that if a bad starting vector is selected it will not converge at all.

The monotone sequence $\{||r_k||\}$ is bounded from below by 0. If the limit of the sequence is 0 as $k \to \infty$ then case 2 will be observed and convergence will be cubic. If the limit is greater than 0, case 3 is found. Unfortunately, it does not seem that this can be know *a priori*. In practice, however, users found that it was difficult to make the method fail for normal matrices.

For non-normal matrices the stationary property does not hold, which means the convergence is quadratic at best. The residual sequence is also not guaranteed to be monotonically decreasing and thus no global convergence properties can be proven. It has been found in practice that RQI will still converge for non-normal systems, just at a slower rate than for normal matrices. However, convergence cannot be guaranteed nor predicted in advance.