Arnoldi's Method for Monte Carlo Criticality Calculations

Jeremy Lloyd Conlin
jlconlin@umich.edu

University of Michigan, Ann Arbor, Michigan

Oak Ridge National Laboratory

Outline

1. Krylov Subspace Methods
2. Power Method
3. Arnoldi's Method
   - Restarted Arnoldi's Method
   - Monte Carlo Implementation
     - Spatial Discretization
     - Negative Sources
4. Numerical Results
   - Discretization
   - Relaxed Matrix-vector Products
   - Relaxed Arnoldi's Method
   - Discussion
5. Future Work
   - Implicitly Restarted Arnoldi's Method

Krylov Subspace Methods

Krylov Subspace:

$$\mathcal{K}_m \;=\; \mathrm{span}\left\{v, \mathbf{A}v, \mathbf{A}^2 v, \ldots, \mathbf{A}^{m-1} v\right\}$$

- Constructed iteratively
- Don't require explicit knowledge of linear operator
- Perfect for Monte Carlo criticality applications

Particle Transport Equation

- Boltzmann Equation:

$$\boldsymbol{\Omega} \cdot \nabla \psi(\mathbf{r}, \boldsymbol{\Omega}) + \Sigma_t \psi(\mathbf{r}, \boldsymbol{\Omega}) - \frac{\Sigma_s}{4\pi} \int \psi(\mathbf{r}, \boldsymbol{\Omega}) \, d\Omega = \frac{1}{k} \frac{\nu \Sigma_f}{4\pi} \int \psi(\mathbf{r}, \boldsymbol{\Omega}) \, d\Omega$$

- Operator Form:

$$\begin{aligned} \left(\mathbf{L} + \mathbf{C} - \mathbf{S}\right) \psi &= \frac{1}{k} \mathbf{F} \psi \\ \mathbf{T} \psi &= \frac{1}{k} \mathbf{F} \psi \end{aligned}$$

- Define:

$$q \equiv \mathbf{F} \psi \qquad\qquad \mathbf{A} \equiv \mathbf{F} \mathbf{T}^{-1}$$

- Finally:

$$\mathbf{A} q = k q$$

Monte Carlo Application of **A**

$$\mathbf{A}q = kq$$

$$q_{m+1} = \frac{1}{k_m}\mathbf{A}q_m$$

- q_m is a fission source
- Neutron is sampled from q_m and transported
- New fission sites are generated at points where neutron causes fission

Krylov Subspace—Power Method

$$\mathcal{K}_m = \mathrm{span}\left\{v, \mathbf{A}v, \mathbf{A}^2 v, \ldots, \mathbf{A}^{m-1} v\right\}$$

- Straightforward implementation of a Krylov Subspace
- Subspace becomes "not-helpful" because Krylov vectors are all nearly the same

Krylov Subspace—Power Method

Power Iteration:

$$q_{m+1} = \frac{1}{k_m}\mathbf{A}q_m,$$

$$k_{m+1} = k_m \frac{\int \mathbf{A}q_m}{\int q_m} = k_m \frac{\sum_{i=1}^{N} \omega_{m+1}^{(i)}}{\sum_{i=1}^{N} \omega_{m}^{(i)}}$$

- As *m* increases we are guaranteed to converge to the true eigenvector and eigenvalue (assuming no bias)
- Convergence is slow—proportional to dominance ratio (λ_1/λ_0)
- Forgets/ignores information from previous iterations

Arnoldi's Method

$$\mathcal{K}_m = \mathrm{span}\left\{v, \mathbf{A}v, \mathbf{A}^2 v, \ldots, \mathbf{A}^{m-1} v\right\}$$
$$\mathcal{K}_m = \mathrm{span}\left\{q_0, q_1, q_2, \ldots, q_{m-1}\right\}$$

- Uses all previously calculated vectors
- q_i's are called Arnoldi vectors
- Linear operator is applied in the same way as for the power method

Arnoldi's Method

Arnoldi Iteration:

- Application of operator

$$\tilde{q}_{m+1} = \mathbf{A} q_m$$

- Orthogonalized against previous Arnoldi vectors

$$\tilde{q}_{m+1} = \mathbf{A} q_m - \sum_{j=1}^{m} q_j h_{jm}$$

- Normalized

$$q_{m+1} = \frac{\tilde{q}_{m+1}}{h_{m+1,m}}$$

Arnoldi Iteration

Arnoldi iteration:

$$\tilde{q}_{k+1} = \mathbf{A} q_k - \sum_{j=1}^{k} q_j h_{jk} \qquad q_{k+1} = \frac{\tilde{q}_{k+1}}{h_{k+1,k}}$$

$$h_{jk} = \langle A q_k, q_j \rangle = \langle q_{k+1}, q_j \rangle.$$

Arnoldi Iteration

Arnoldi iteration:

$$\tilde{q}_{k+1} = \mathbf{A} q_k - \sum_{j=1}^{k} q_j h_{jk} \qquad q_{k+1} = \frac{\tilde{q}_{k+1}}{h_{k+1,k}}$$

$$h_{jk} = \langle A q_k, q_j \rangle = \langle q_{k+1}, q_j \rangle.$$

After m iterations

$$\mathbf{A} Q_m = Q_m H_m + q_{m+1} h_{m+1,m} e_m^T$$

Eigenvalues from Arnoldi's Method

$$\mathbf{A}Q_m = Q_m H_m + q_{m+1} h_{m+1,m} e_m^T$$

The eigenvalues of H_m can be calculated

$$H_m x = \lambda x$$

Eigenvalues from Arnoldi's Method

$$\mathbf{A}Q_m = Q_m H_m + q_{m+1} h_{m+1,m} e_m^T$$

The eigenvalues of H_m can be calculated

$$H_m x = \lambda x$$

Arnoldi factorization can gives us the Ritz pairs of $\mathbf{A}$:

$$\mathbf{A}Q_m x_m = Q_m H_m x_m + q_{m+1} h_{m+1,m} e_m^T x_m$$

$$\mathbf{A}Q_m x_m = \lambda Q_m x_m + q_{m+1} h_{m+1,m} e_m^T x_m$$

$$\mathbf{A}y_m = \lambda y_m + q_{m+1} h_{m+1,m} e_m^T x_m$$

Eigenvalues from Arnoldi's Method

$$\mathbf{A}Q_m = Q_m H_m + q_{m+1} h_{m+1,m} e_m^T$$

The eigenvalues of H_m can be calculated

$$H_m x = \lambda x$$

Arnoldi factorization can gives us the Ritz pairs of $\mathbf{A}$:

$$\mathbf{A}Q_m x_m = Q_m H_m x_m + q_{m+1} h_{m+1,m} e_m^T x_m$$

$$\mathbf{A}Q_m x_m = \lambda Q_m x_m + q_{m+1} h_{m+1,m} e_m^T x_m$$

$$\mathbf{A}y_m = \lambda y_m + q_{m+1} h_{m+1,m} e_m^T x_m$$

The Ritz pair (λ, y_m) is an estimate of an eigenpair of $\mathbf{A}$.

Residual

$$\mathbf{A} y_m = \lambda y_m + q_{m+1} h_{m+1,m} e_m^T x_m$$

Residual:

$$\begin{aligned} r_m &= \left\| \mathbf{A} y_m - \lambda y_m \right\| \\ &= \left| q_{m+1} h_{m+1,m} e_m^T x_m \right| \\ &= \left| h_{n+1,n} \right| \left| e_n^T x_n \right| \end{aligned}$$

If r_m is small, (λ, y_n) is a good approximation of an eigenpair of $\mathbf{A}$.

Restarted Arnoldi's Method

What is a good starting vector?

Restarted Arnoldi's Method

What is a good starting vector?

$$q_0 = a_0 x_0 + a_1 x_1 + \cdots + a_n x_n$$

x_i's are eigenvectors from desired portion of spectrum of $\mathbf{A}$.

Restarted Arnoldi's Method

What is a good starting vector?

$$q_0 = a_0 x_0 + a_1 x_1 + \cdots + a_n x_n$$

x_i's are eigenvectors from desired portion of spectrum of $\mathbf{A}$.

Restarting Arnoldi:

- Improves restart vector
- Reduces size of Krylov Subspace
- Reduces computational expense of orthogonalization

Spatial Discretization

What is the inner product of two fission sources?

$$h_{jk} = \langle q_k, q_j \rangle = \int q_k(x) q_j(x) dx$$

Spatial Discretization

What is the inner product of two fission sources?

$$h_{jk} = \langle q_k, q_j \rangle = \int q_k(x) q_j(x) dx$$

Must discretize fission source

$$q(x) = \sum_{i=1}^{B} a_i \Pi_i(x),$$

$$\Pi_i(x) = \begin{cases} 1, & x_i \leq x < x_{i+1} \\ 0, & \text{otherwise.} \end{cases}$$

Spatial Discretization

Discretized fission source

$$q(x) = \sum_{i=1}^{B} a_i \Pi_i(x), \qquad \Pi_i(x) = \begin{cases} 1, & x_i \leq x < x_{i+1} \\ 0, & \text{otherwise.} \end{cases}$$

Spatial Discretization

Discretized fission source

$$q(x) = \sum_{i=1}^{B} a_i \Pi_i(x), \qquad \Pi_i(x) = \begin{cases} 1, & x_i \le x < x_{i+1} \\ 0, & \text{otherwise.} \end{cases}$$

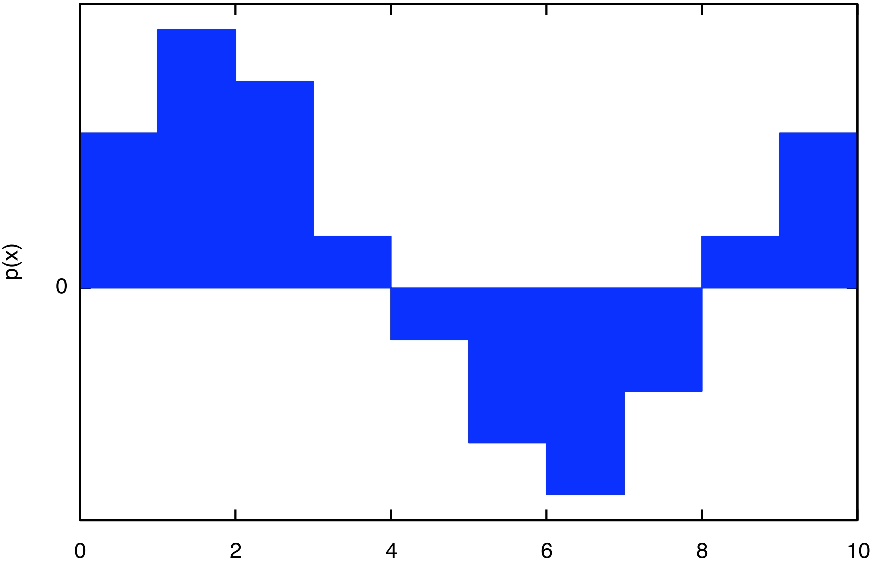
Vector representation

$$q_k = \left[a_1^{(k)} \; a_2^{(k)} \; \dots \; a_B^{(k)} \right]^T$$

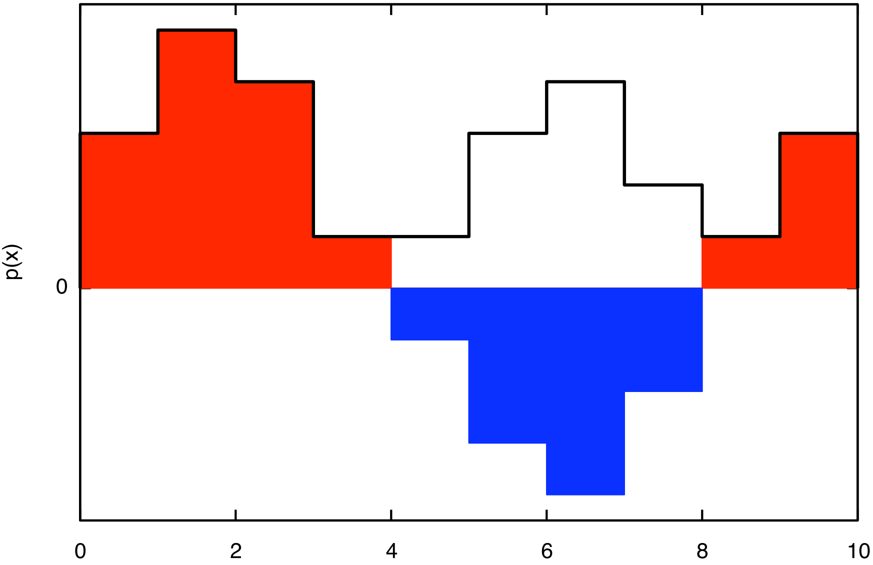
Inner product becomes dot product between two vectors:

$$h_{jk} = \int q_k(x) q_j(x) dx = \sum_{i=1}^{B} a_i^{(k)} a_i^{(j)}$$

Negative Sources



Negative Sources



Negative Sources

$$
\begin{aligned}
q^{(+)}(x) &= \begin{cases} q(x), & q(x) \geq 0 \\ 0, & \text{else} \end{cases} \\
q^{(-)}(x) &= \begin{cases} q(x), & q(x) < 0 \\ 0, & \text{else} \end{cases}
\end{aligned}
$$

$$q(x) \equiv q^{(+)}(x) + q^{(-)}(x)$$

$$\mathbf{A}q(x) = \mathbf{A}q^{(+)}(x) - \mathbf{A}q^{(-)}(x)$$

Negative Sources

$$\begin{aligned} q^{(+)}(x) &= \begin{cases} q(x), & q(x) \geq 0 \\ 0, & \text{else} \end{cases} \\ q^{(-)}(x) &= \begin{cases} q(x), & q(x) < 0 \\ 0, & \text{else} \end{cases} \end{aligned}$$

$$\begin{aligned} p(x) &\equiv \frac{|q(x)|}{\int |q(x)|\, dx} \\ &= \frac{q^{(+)}(x) + |q^{(-)}(x)|}{\int |q(x)|\, dx} \end{aligned}$$

Problem Parameters

- Homogeneous slab geometry
- $\Sigma_s = 0.5,\ \nu\Sigma_f = 0.5,\ \Sigma_t = 1.0$
- Both methods tracked the same number of particles

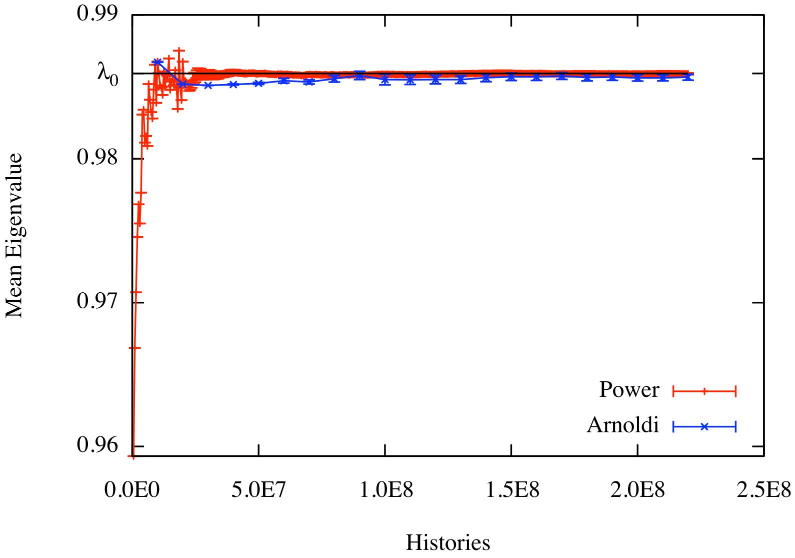
Width (mfp)	Spatial Bins	Particles	Iterations	Inactive Restarts	Active Restarts
20	75	500,000	20	2	20
			40/400 power cycles		
50	75	500,000	30	10	30
			300/900 power cycles		
100	100	10,000,000	30	5	25
			150/750 power cycles		

Numerical Results

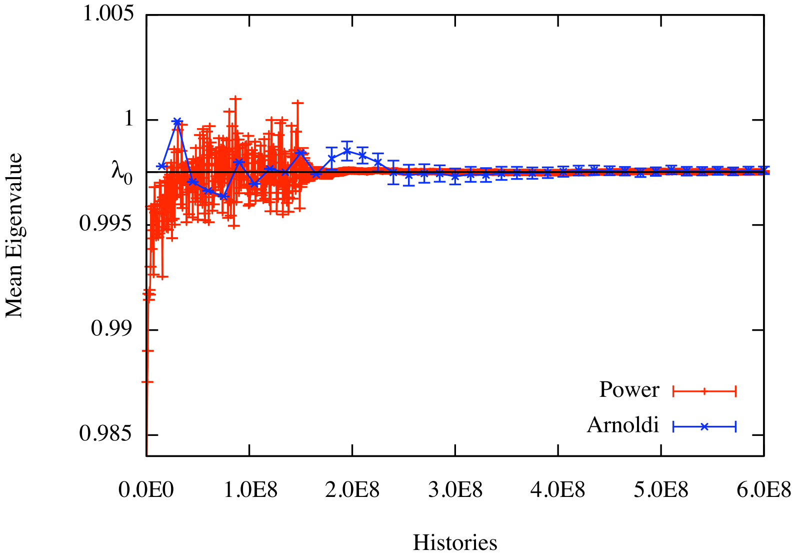
Width (mfp)	λ_0	Method	Eigenvalue	Runtime (s)
20	0.985928	Power	0.98590 ± 0.00006	833
		Arnoldi	0.9857 ± 0.0002	767
50	0.99752	Power	0.99753 ± 0.00004	2307
		Arnoldi	0.9976 ± 0.0002	2156
100	0.99933	Power	0.99929 ± 0.00001	42820
		Arnoldi	0.99926 ± 0.00007	35240

Jeremy Lloyd Conlin and James Paul Holloway, *Arnoldi's method of minimized iterations for Monte Carlo criticality calculations*, PHYSOR 2008, Interlaken, Switzerland

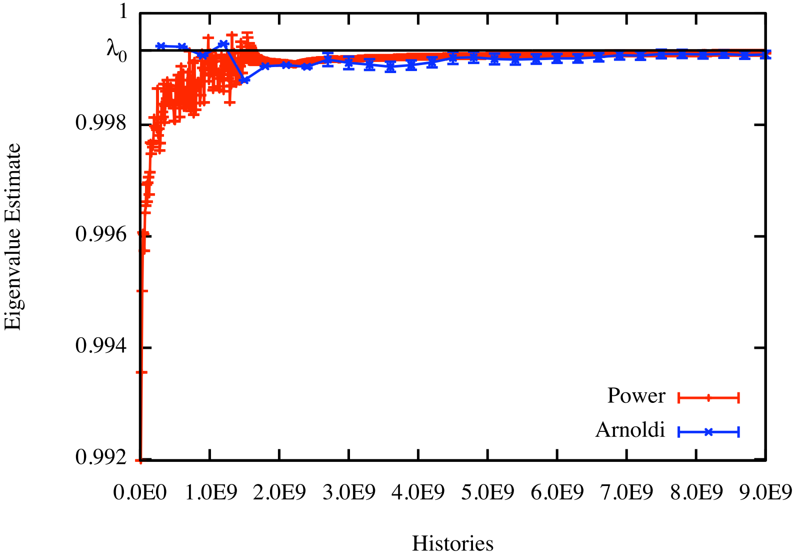
Eigenvalue Convergence—20 mfp



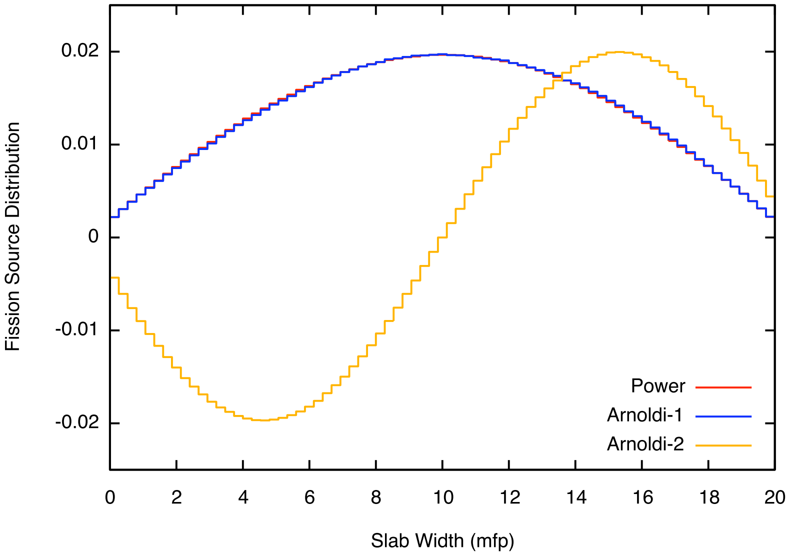
Eigenvalue Convergence—50 mfp



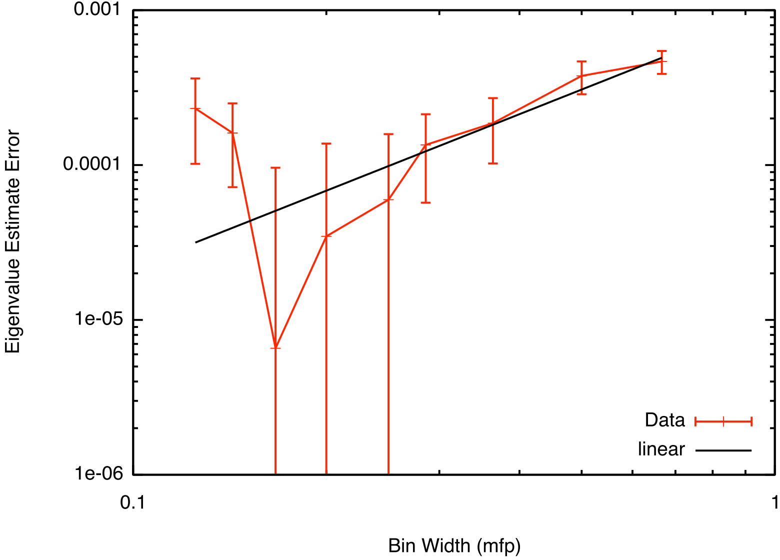
Eigenvalue Convergence—100 mfp



Eigenvectors



Discretization Bias



Relaxed Arnoldi's Method

$$r_m = \|\mathbf{A} y_m - \lambda y_m\| = \left| q_{m+1} h_{m+1,m} e_m^T x_m \right|$$

Relaxed Arnoldi's Method

$$r_m = \|\mathbf{A} y_m - \lambda y_m\| = \left| q_{m+1} h_{m+1,m} e_m^T x_m \right|$$

Bouras and Frayssé:

- Can *relax* application of linear operator
- Maintain convergence of Arnoldi's method
- Save computational expense

Relaxed Arnoldi's Method

$$\hat{q}_{m+1} = \left(\mathbf{A} + \mathbf{\Delta A}\right) q_m$$

Relaxed Arnoldi's Method

$$\hat{q}_{m+1} = \left(\mathbf{A} + \mathbf{\Delta A}\right) q_m$$

Magnitude of $\mathbf{\Delta A}$ is prevented from growing too large

$$\alpha_k = \frac{1}{\min\left(\left\| r_{k-1} \right\|, 1\right)}$$

$$\left\| \mathbf{\Delta A}_k \right\| = \varepsilon_k \left\| \mathbf{A} \right\|$$

where $\varepsilon_k = \min\left(\alpha_k \eta, 1\right)$.

Relaxed Arnoldi's Method

$$\hat{q}_{m+1} = (\mathbf{A} + \mathbf{\Delta A})\, q_m$$

Magnitude of $\mathbf{\Delta A}$ is prevented from growing too large

$$\alpha_k = \frac{1}{\min\left(\|r_{k-1}\|, 1\right)}$$

$$\|\mathbf{\Delta A}_k\| = \varepsilon_k \|\mathbf{A}\|$$

where $\varepsilon_k = \min\left(\alpha_k \eta, 1\right)$.

As residual decreases, precision of application of linear operator decreases.

Monte Carlo Relaxation

$$\hat{q}_{m+1} = (\mathbf{A} + \mathbf{\Delta A})\, q_m$$

$$\mathbf{\Delta A} \propto \frac{1}{\sqrt{N}}$$

More than 80% of time spent on tracking particles (application of **A**)

Monte Carlo Relaxation

$$\hat{q}_{m+1} = \left(\mathbf{A} + \mathbf{\Delta A}\right) q_m$$
$$\mathbf{\Delta A} \propto \frac{1}{\sqrt{N}}$$

More than 80% of time spent on tracking particles (application of **A**)

How to relax Monte Carlo

Monte Carlo Relaxation

$$\hat{q}_{m+1} = \left(\mathbf{A} + \mathbf{\Delta A}\right) q_m$$
$$\mathbf{\Delta A} \propto \frac{1}{\sqrt{N}}$$

More than 80% of time spent on tracking particles (application of $\mathbf{A}$)

How to relax Monte Carlo—track fewer particles when residual is small

$$N_k = \begin{cases} N_0 & , \quad \eta < r_{k-1} \\ N_0 \left(r_{k-1}/\eta\right) & , \quad \eta \geq r_{k-1}, \end{cases}$$

Problem Parameters

- Homogeneous slab geometry
- 20mfp thick
- 1E6 particles per iteration
- $\eta = 0.1$ for relaxed Arnoldi

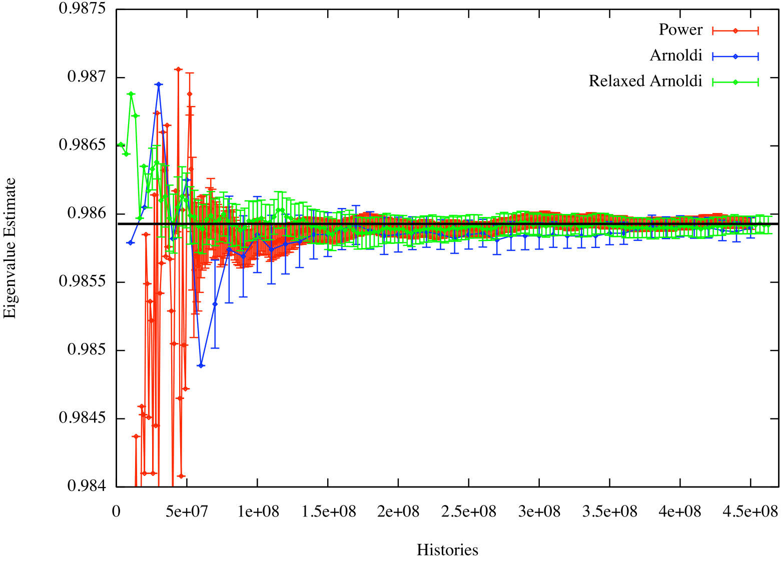
Material	Method	Iterations	Inactive Restarts	Active Restarts
Absorbing	Power	50/280 power cycles		
	Arnoldi	10	5	28
	Relaxed Arnoldi	10	5	150
Scattering	Power	150/250 power cycles		
	Arnoldi	10	15	25
	Relaxed Arnoldi	10	15	150

Numerical Results

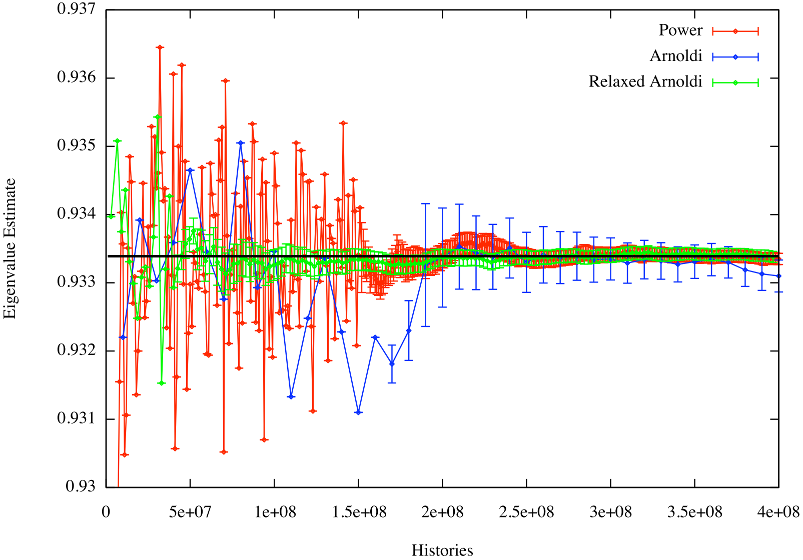
Material	λ_0	Method	Eigenvalue
Absorbing	0.98593	Power	0.98593 ± 0.00004
		Arnoldi	0.98590 ± 0.00008
		Relaxed Arnoldi	0.98592 ± 0.00006
Scattering	0.93339	Power	0.93335 ± 0.00008
		Arnoldi	0.93331 ± 0.00023
		Relaxed Arnoldi	0.93338 ± 0.00008

Jeremy Lloyd Conlin and James Paul Holloway, *Relaxation scheme for Monte Carlo explicitly restarted Arnoldi's method for criticality calculations*, M&C 2009, Saratoga Springs, NY

Eigenvalue Convergence—Absorbing Material



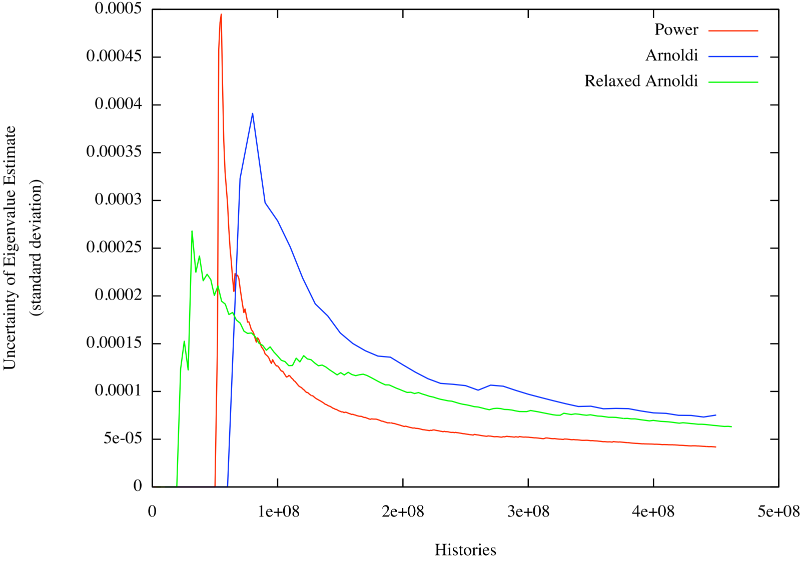
Eigenvalue Convergence—Scattering Material



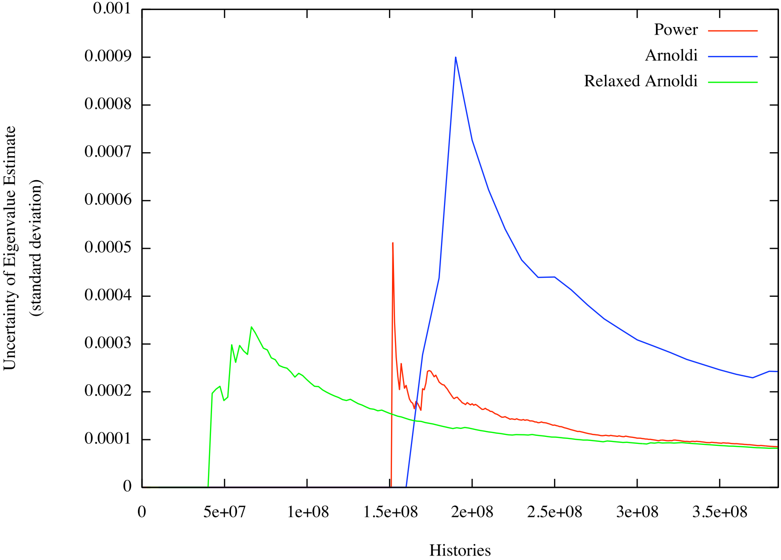
Relaxed Timing Results

Material	Method	Runtime (s)	Particles/sec	Cost
Absorbing	Power	1742	2.58×10^5	1
	Arnoldi	1669	2.70×10^5	0.956
	Relaxed Arnoldi	1972	2.34×10^5	1.100
Scattering	Power	2310	1.73×10^5	1
	Arnoldi	2340	1.71×10^5	1.01
	Relaxed Arnoldi	2272	1.75×10^5	0.989

Uncertainty—Absorbing Material



Uncertainty—Scattering Material



Discussion

Arnoldi's method:

- Can be used in Monte Carlo criticality applications
- Can calculate higher order eigenmodes
- Should be relaxed to reduce uncertainty and possibly save time

Outline

1. Krylov Subspace Methods
2. Power Method
3. Arnoldi's Method
   - Restarted Arnoldi's Method
   - Monte Carlo Implementation
     - Spatial Discretization
     - Negative Sources
4. Numerical Results
   - Discretization
   - Relaxed Matrix-vector Products
   - Relaxed Arnoldi's Method
   - Discussion
5. Future Work
   - Implicitly Restarted Arnoldi's Method

Restarted Arnoldi

$$\mathbf{A} Q_m = Q_m H_m + r_m$$

Restart vector:

$$q_0 = a_0 y_0 + a_1 y_1 + \cdots + a_{n-1} y_{n-1}$$

Restarted Arnoldi

$$\mathbf{A} Q_m = Q_m H_m + r_m$$

Restart vector:

$$q_0 = a_0 y_0 + a_1 y_1 + \cdots + a_{n-1} y_{n-1}$$

- Desire j eigenpairs

Restarted Arnoldi

$$\mathbf{A}Q_m = Q_m H_m + r_m$$

Restart vector:

$$q_0 = a_0 y_0 + a_1 y_1 + \cdots + a_{n-1} y_{n-1}$$

- Desire j eigenpairs
- Want $a_j, \cdots, a_{n-1}$ to be small

Restarted Arnoldi

$$\mathbf{A}Q_m = Q_m H_m + r_m$$

Restart vector:

$$q_0 = a_0 y_0 + a_1 y_1 + \cdots + a_{n-1} y_{n-1}$$

- Desire j eigenpairs
- Want $a_j, \cdots, a_{n-1}$ to be small
- Explicit restarts force $a_j = \cdots = a_{n-1} = 0$

Implicit Restarts

Replace Arnoldi Iterations with shifted QR-algorithm iterations on H_m

$$(H_m - \mu_1 I) = V_1 R_1$$

$$\hat{H}_1 = V_1^* H_m V_1$$

After j iterations

$$\hat{H}_p = \hat{V}_p^* H_m \hat{V}_p$$

where

$$\hat{V}_p = V_1 V_2 \cdots V_p$$

Implicit Restarts

Substitute $H_m = \hat{V}_p H_m \hat{V}_p^*$ into Arnoldi factorization

$$\begin{aligned}
\mathbf{A} Q_m &= Q_m H_m + r_m \\
\mathbf{A} Q_m &= Q_m \hat{V}_p H_m \hat{V}_p^* + r_m \\
\mathbf{A} Q_m \hat{V}_p &= Q_m \hat{V}_p H_m \hat{V}_p^* \hat{V}_p + r_m \hat{V}_p \\
\mathbf{A} \hat{Q}_m &= \hat{Q}_m H_m + r_m \hat{V}_p
\end{aligned}$$

where $\hat{Q}_m = Q_m \hat{V}_p$.

Implicit Restarts

$$\mathbf{A}\hat{Q}_m = \hat{Q}_m H_m + r_m \hat{V}_p$$

- Arnoldi factorization
- Restarted with p iterations of shifted QR-algorithm

Implicit Restarts

$$\mathbf{A}\hat{Q}_m = \hat{Q}_m H_m + r_m \hat{V}_p$$

- Arnoldi factorization
- Restarted with p iterations of shifted QR-algorithm
- If shifts are chosen judiciously:
  - Implicit restart equivalent to explicit restart
  - Suppresses unwanted portion of spectrum of $\mathbf{A}$
  - Implicit restarts can then proceed at iteration $p + 1$

Implicit Restarts

$$\mathbf{A}\hat{Q}_m = \hat{Q}_m H_m + r_m \hat{V}_p$$

- Arnoldi factorization
- Restarted with p iterations of shifted QR-algorithm
- If shifts are chosen judiciously:
    - Implicit restart equivalent to explicit restart
    - Suppresses unwanted portion of spectrum of **A**
    - Implicit restarts can then proceed at iteration $p+1$
- Implicit restarts trade applications of **A** for shifted QR-algorithm iterations

Questions