Acceleration of Source Iteration using the Dynamic Mode Decomposition

Ryan G. McClarren\textsuperscript{1} & Terry S. Haut\textsuperscript{2}
rmcclarr@nd.edu haut3@llnl.gov

1. Department of Aerospace & Mechanical Engineering, University of Notre Dame
2. Lawrence Livermore National Laboratory

August 28, 2019
Disclaimer This document was prepared as an account of work sponsored by an agency of the United States government. Neither the United States government nor Lawrence Livermore National Security, LLC, nor any of their employees makes any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States government or Lawrence Livermore National Security, LLC. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States government or Lawrence Livermore National Security, LLC, and shall not be used for advertising or product endorsement purposes.

LLNL-PRES-787899 This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under contract DE-AC52-07NA27344. Lawrence Livermore National Security, LLC
Outline

1 Introduction

2 The Dynamic Mode Decomposition

3 DMD for Accelerating Source Iteration
   3.1 Slab geometry examples

4 Conclusions
In scientific computing we are used to taking a known operator and making approximations to it.

It is possible to use the action of an operator and use just the action of the operator to generate approximations to it.

- This is the basis for many Krylov methods.

In this talk I will detail how we can use the action of radiation transport operators to compute the slowly converging modes in source iteration to accelerate convergence without the need for diffusion-based preconditioning.

The basis for this work is the dynamic mode decomposition (DMD). This method can

- Estimate time eigenvalues present in a subcritical system (Ryan G. McClarren (2019) “Calculating Time Eigenvalues of the Neutron Transport Equation with Dynamic Mode Decomposition”, Nuclear Science and Engineering, 193:8, 854-867), and

- Be used to produce an inexpensive reduced-order model (Zachary K. Hardy, Jim E. Morel Cory Ahrens (2019) “Dynamic Mode Decomposition for Subcritical Metal Systems”, Nuclear Science and Engineering)
We begin with a sequence of vectors related by an operator.

- Consider a sequence of vectors \( \{y_0, y_1, \ldots, y_K\} \) where \( y_k \in \mathbb{R}^N \).
- The vectors are related by a potentially unknown linear operator of size \( N \times N \), \( A \), as
  \[
  y_{k+1} = Ay_k.
  \]
- If we construct the \( N \times K \) data matrices \( Y_+ \) and \( Y_- \),
  \[
  Y_+ = \begin{pmatrix}
  y_1 & y_2 & \cdots & y_K \\
  \vdots & \vdots & \ddots & \vdots \\
  y_0 & y_1 & \cdots & y_K-1
  \end{pmatrix}
  \hspace{1cm}
  Y_- = \begin{pmatrix}
  y_0 & y_1 & \cdots & y_K-1
  \end{pmatrix}
  \]
  we can write
  \[
  Y_+ = AY_-.\]
- At this point we only need to know the data vectors \( y_k \), they could come from a calculation, measurement, etc.
- As \( K \to \infty \) we could hope to infer properties about \( A \).
The SVD gives a representation of the data matrices.

- We take the thin singular value decomposition (SVD) of $Y_-$ to write
  \[ Y_- = U\Sigma V^T, \]
  where $U$ is a $N \times K$ orthogonal matrix, $\Sigma$ is a diagonal $K \times K$ matrix with non-negative entries on the diagonal, and $V$ is a $K \times K$ orthogonal matrix.
- The SVD requires $O(NK^2)$ operations to compute.
- Later, we will want $K \ll N$, if, for example, $N$ is the number of unknowns in a transport calculation.
- Also, if the column rank of $Y_- < K$, then there is a further reduction in the SVD size.
- The matrix $U$ has columns that forms an orthonormal basis for the row space of $Y_- \subset \mathbb{R}^N$.
- Using the SVD we get
  \[ Y_+ = AU\Sigma V^T. \]
- If there are only $r < K$ non-zero singular values in $\Sigma$, we use the compact SVD where $U$ is $N \times r$, $\Sigma$ is $r \times r$, and $V$ is $K \times K$. 

We can rearrange the relationship between $Y_+$ and $Y_-$ to be

$$Y_+ = A U \Sigma V^T \quad \rightarrow \quad U^T A U = U^T Y_+ V \Sigma^{-1}.$$ 

Define $\tilde{A} = U^T A U = U^T Y_+ V \Sigma^{-1}$. This is a rank $K$ approximation to $A$.

Using the approximate operator $\tilde{A}$, we can now find out information about $A$.

The eigenvalues/vectors of $\tilde{A}$,

$$\tilde{A} w = \lambda w,$$

are used to define the dynamic modes of $A$:

$$\varphi = \frac{1}{\lambda} U^T Y_+ V \Sigma^{-1} w.$$ 

The dynamic mode decomposition (DMD) of the data matrix $Y_+$ is then the decomposition of into vectors $\varphi$. The mode with the largest norm of $\lambda$ is said to be the dominant mode.
Consider the sequence

\[ z_{k+1} = az_k + n_k, \]

where \( a = 0.5 \), and \( n_k \sim \mathcal{N}(0, 10^2) \).

Using \( K = 500 \), we estimate \( a = 0.506552 \) from the data below.
Evolution without an operator is possible: DMD infers the operator from the data.

Left: Data generated by moving a circle in a periodic motion with added noise. The data has two periods of motion.

Right: Reconstruction generated by approximating $\tilde{A}$ using one period of frames and starting from frame 1.
The DMD modes of the inferred operator are what we would expect.

Dominant DMD mode: $U\varphi_1$

Second DMD mode: $U\varphi_2$

Third DMD mode: $U\varphi_3$

Fourth DMD mode: $U\varphi_4$
The approximate operator can be used to find slowly converging modes in an iterative method.

- The discrete ordinates method for transport is typically solved using source iteration (Richardson iteration) and diffusion-based preconditioning/acceleration.
- Source iterations converge quickly for problems with a small amount of particle scattering.
- For strongly scattering media, the transport operator has a near nullspace that can be handled using a diffusion preconditioner.
- However, the question of efficiently preconditioning/accelerating transport calculation on high-order meshes with discontinuous fine elements is an open area of research.
- The approximate operator found from DMD can be used to remove this same near nullspace and improve iterative convergence without the need for a separate preconditioner or diffusion discretization/solve.
The steady, single group transport equation with isotropic scattering can be written as
\[ L\psi = \frac{c}{4\pi}\phi + \frac{Q}{4\pi}, \]
where \( c \) is the scattering ratio, \( Q \) is a prescribed source, and the streaming and removal operator is
\[ L = (\Omega \cdot \nabla + 1). \]

- \( \psi(\mathbf{x}, \Omega), \Omega \in S_2, \)
  \[ \phi(\mathbf{x}) = \int_{4\pi} \psi \, d\Omega = \langle \psi \rangle. \]
- Source iteration solves this problem using the iteration strategy
  \[ \phi^\ell = \left\langle L^{-1} \left( \frac{c}{4\pi} \phi^{\ell-1} + \frac{Q}{4\pi} \right) \right\rangle, \]
  where \( \ell \) is an iteration index.
- One iteration is often called a "transport sweep".
- A benefit of source iteration is that the angular flux, \( \psi \) does not have to be stored.
- As \( c \to 1 \), the convergence of source iteration can be arbitrarily slow. 

We consider source iteration for a simple transport problem.
We can write source iteration as a sequence of vectors related by a linear operator.

- Rearranging the transport equation we see that source iteration is an iterative procedure for solving

\[
\phi - \left< \frac{L^{-1} C}{4\pi} \phi \right> = L^{-1} Q,
\]

or

\[(I - A)\phi = b.\]

- Therefore, the source iteration vectors are

\[\phi^{\ell+1} = A\phi^{\ell} + b,\]

or

\[\phi^{\ell+1} - \phi^{\ell} = A(\phi^{\ell} - \phi^{\ell-1}).\]

- Therefore, we can cast the difference between iterates in a form that is amenable to the approximation of \( A \) using DMD, \( Y_+ = AY_- \),

\[
Y_+ = \begin{bmatrix}
\phi^2 - \phi^1, & \phi^3 - \phi^2, & \ldots, & \phi^K - \phi^{K-1}
\end{bmatrix},
\]

\[
Y_- = \begin{bmatrix}
\phi^1 - \phi^0, & \phi^2 - \phi^3, & \ldots, & \phi^{K-1} - \phi^{K-2}
\end{bmatrix}.
\]
As before we define an approximate $A$ as the $K \times K$ matrix:

$$\tilde{A} = U^T A U = U^T Y V \Sigma^{-1},$$

We can use $\tilde{A}$ to construct the operator $(I - \tilde{A})^{-1}$ and use this to approximate the solution:

$$(I - A)(\phi - \phi^{K-1}) = b - (I - A)\phi^{K-1} = b - \phi^{K-1} + (\phi^K - b) = \phi^K - \phi^{K-1}.$$

The difference $\phi - \phi^{K-1}$ is the difference between step $K - 1$ and the converged answer. We define a new vector $\Delta y$ as the length $K$ vector that satisfies

$$\phi - \phi^{K-1} = U \Delta y.$$  \hfill (1)

We then substitute and multiply by $U^T$ to get

$$(I - \tilde{A}) \Delta y = U^T (\phi^K - \phi^{K-1}).$$  \hfill (2)

This is a linear system of size $K$ that we can solve to get $\Delta y$ and then compute the update to $\phi^{K-1}$ as

$$\phi \approx \phi^{K-1} + U \Delta y.$$  \hfill (3)
The algorithm is as follows

1. Perform $R$ source iterations: $\phi^\ell = A\phi^{\ell-1} + b$.
2. Compute $K$ source iterations to form $Y_+$ and $Y_-$. The last column of $Y_-$ we call $\phi^{K-1}$.
3. Compute $\phi = \phi^{K-1} + U\Delta y$ as above.

- Each pass of the algorithm requires $R + K$ source iterations.
- The $R$ source iterations are used to correct any errors caused by the approximation of $A$ using the SVD.
- It is easiest to assess convergence between the source iterations.
- This works regardless of the spatial discretization used.
- Other algorithms are possible:
  - Rather than extrapolate to an infinite number of iterations, we can use $\tilde{A}$ to approximate a finite number of source iterations.
  - We could use a coarsened vector $\tilde{\phi}$ in the DMD procedure to reduce the memory/computational cost.

DMD acceleration requires only source iteration and SVD: no diffusion solver.
DMD works perfectly on a homogenous slab, the ur-demonstration problem for acceleration schemes.

- We consider a slab with vacuum boundaries and a scattering ratio of $c = 0.99$ and $1.0$ and 400 spatial zones, $S_8$ angular discretization, and the diamond difference spatial discretization.
- Solid lines are $c = 0.99$ results and dashed lines are $c = 1.0$
A comparison of the number of iterations as a function of $K$ and $c$ indicates that the convergence is nearly independent of $c$.

- On the same problem set up, the number of iterations to converge is shown below.

<table>
<thead>
<tr>
<th>$K/c$</th>
<th>0.1</th>
<th>0.5</th>
<th>0.9</th>
<th>0.99</th>
<th>0.999</th>
<th>0.9999</th>
<th>0.99999</th>
<th>0.999999</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>8</td>
<td>15</td>
<td>39</td>
<td>70</td>
<td>70</td>
<td>70</td>
<td>70</td>
<td>70</td>
</tr>
<tr>
<td>5</td>
<td>10</td>
<td>11</td>
<td>28</td>
<td>90</td>
<td>90</td>
<td>90</td>
<td>90</td>
<td>90</td>
</tr>
<tr>
<td>10</td>
<td>15</td>
<td>15</td>
<td>29</td>
<td>60</td>
<td>140</td>
<td>140</td>
<td>140</td>
<td>140</td>
</tr>
<tr>
<td>20</td>
<td>25</td>
<td>25</td>
<td>25</td>
<td>49</td>
<td>74</td>
<td>76</td>
<td>76</td>
<td>76</td>
</tr>
<tr>
<td>50</td>
<td>55</td>
<td>55</td>
<td>55</td>
<td>56</td>
<td>57</td>
<td>57</td>
<td>57</td>
<td>57</td>
</tr>
<tr>
<td>SI</td>
<td>6</td>
<td>17</td>
<td>89</td>
<td>637</td>
<td>2439</td>
<td>3681</td>
<td>3889</td>
<td>3911</td>
</tr>
</tbody>
</table>
We consider a problem with vacuum boundaries, 1000 cells, unit domain length, with \( c = 0.9999 \) and

\[
\sigma_t = \begin{cases} 
2^p & \text{cell number odd} \\
2^{-p} & \text{cell number even}
\end{cases}
\]

Below we see convergence for \( p = 5 \) (dashed) and \( p = 8 \) (solid), a factor of about 1000 and \( 6.5 \times 10^4 \) between thick and thin cells, respectively.

Performance does degrade on an insanely heterogeneous problem.
A version of the crooked pipe problem is a more realistic test.

- We solve a linear, xy-geometry version of the crooked pipe problem where all materials have a scattering ratio of 0.988 (to simulate a realistic sized time step).
- The density ratio between the tick and thin material is 1000.
- Problem solved using fully lumped, bilinear discontinuous Galerkin in space and $S_8$ product quadrature.
The number of iterations required increases slowly with mesh refinement.

- The number of iterations for source iteration and DMD-accelerated calculations with $K = 10$ and $R = 3$.

<table>
<thead>
<tr>
<th>$(N_x \times N_y)$</th>
<th>DMD</th>
<th>SI</th>
</tr>
</thead>
<tbody>
<tr>
<td>25 $\times$ 15</td>
<td>53</td>
<td>811</td>
</tr>
<tr>
<td>50 $\times$ 25</td>
<td>52</td>
<td>873</td>
</tr>
<tr>
<td>100 $\times$ 60</td>
<td>78</td>
<td>974</td>
</tr>
<tr>
<td>150 $\times$ 90</td>
<td>91 $\infty_{\text{RML}}$</td>
<td></td>
</tr>
<tr>
<td>200 $\times$ 120</td>
<td>104 $\infty_{\text{RML}}$</td>
<td></td>
</tr>
</tbody>
</table>

$\infty_{\text{RML}}$ = functionally infinite on my laptop.

- The increase seems to be the resolution to the 1/2 power (square root of the number of cells per dimension).
A recent paper, Roberts, Jeremy A., et al. "Acceleration of the Power Method with Dynamic Mode Decomposition." arXiv preprint arXiv:1904.09493 (2019), uses these ideas for power iterations. Here is Fig. 3 from that paper:

Fig. 3: The error in the predicted eigenmode for DMD-PM(n), where n is the number of power iterations performed. Errors are also included for the power method (PM) and Arnoldi’s method.
We could use DMD acceleration to compute a low-order transport acceleration (the so-called TSA method). In this case the we would use low-order in angle transport sweeps to estimate the slowly converging modes.

Additionally, it is possible to estimate $\tilde{A}$ using independently generated vectors. This would enable the $Y_\pm$ matrices to be generated using sweeps computed in parallel.

The big win could be from applying this to other iterative components:
- Energy group iterations
- Temperature iterations in radiative transfer.

The performance of DMD on meshes with cycles is also a possible impact area.
Using a DMD approach to compute approximate operators gives one the ability to
- Estimate eigenvalues for the system, and
- Accelerate calculations.

There is much further research to be done, but progress is exciting.
The DMD acceleration work was sponsored by Lawrence Livermore National Laboratory project titled "Dynamic Mode Decomposition (DMD) Acceleration Methods for Thermal Radiative Transfer" contract B627130.