Least squares approximations in system identification and model reduction
II. Discrete Empirical Interpolation Method

Zlatko Drmač
(Based on the ongoing work with Serkan Gugercin and Arvind Krishna Saibaba)

Faculty of Science, Department of Mathematics
University of Zagreb
Supported by HRZZ 9345

drmac@math.hr

November 2, 2017
Groningen Autumn School on Model Order Reduction

Overview

The stage: projection based POD+DEIM model order reduction

Suppose we want to run numerical simulations of the set of the ODE's:

$$\dot{x}(t) = Ax(t) + \mathbf{f}(x(t)), \quad x(0) = x_0 \in \mathbb{R}^n,$$

where $A \in \mathbb{R}^{n\times n}$, $\mathbf{f} : \mathbb{R}^n \longrightarrow \mathbb{R}^n$.

n is big. A, $\mathbf{f}$ may be parameter dependent.

Galerkin projection

- low dimensional subspace $\mathcal{V}_k = \mathrm{range}(V_k)$, $V_k^T V_k = \mathbb{I}_k$
- seek an approximation $x(t) \approx \overline{x} + V_k \widehat{x}(t)$, $\widehat{x}(0) = V_k^T(x(0) - \overline{x})$
- enforce orthogonality of the residual and $\mathcal{V}_k$

$$\dot{\widehat{x}}(t) = \underbrace{V_k^T A V_k}_{k\times k} \widehat{x}(t) + \underbrace{V_k^T A \overline{x}}_{k\times 1} + \underbrace{V_k^T \mathbf{f}(\overline{x} + V_k \widehat{x}(t))}_{(k\times n)\cdot\mathbf{f}(n\times 1 + (n\times k)\cdot(k\times 1))}$$

DEIM trick, Chaturantabut and Sorensen, 2010

Bottleneck $V_k^T\mathbf{f}(\overline{x}+V_k\widehat{x}(t))$ :: $(k\times n)\cdot\mathbf{f}(n\times 1+(n\times k)\cdot(k\times 1))$

- empirically determine an m-dimensional subspace $\mathcal{U}_m=\mathrm{range}(U_m)$, $U_m^T U_m=\mathbb{I}_m$. such that $U_m U_m^T\mathbf{f}(\overline{x}+V_k\widehat{x}(t))\approx\mathbf{f}(\overline{x}+V_k\widehat{x}(t))$.
- inserting the orthogonal projection $U_m U_m^T$ into the projected system:

$$\begin{aligned}\dot{\widehat{x}}(t) &= V_k^T A V_k\widehat{x}(t)+V_k^T A\overline{x}+V_k^T\mathbf{U_m}\mathbf{U_m^T}\,\mathbf{f}\left(\overline{x}+V_k\widehat{x}(t)\right)\\ &+ V_k^T(\mathbb{I}_m-U_m U_m^T)\,\mathbf{f}\left(\overline{x}+V_k\widehat{x}(t)\right),\end{aligned}$$

DEIM trick: trade an orthogonal for an oblique projector

- replace the orthogonal projector $\mathbf{U_m}\mathbf{U_m^T}$ with an oblique projector $\mathcal{D}=U_m(\mathscr{S}^T U_m)^{-1}\mathscr{S}^T$, where $\mathscr{S}=\left(\mathbb{I}_n(:,i_1),\ldots,\mathbb{I}_n(:,i_m)\right)$
- $V_k^T\mathbf{f}(\overline{x}+V_k\widehat{x}(t))$ is approximated with $V_k^T\mathcal{D}\mathbf{f}(\overline{x}+V_k\widehat{x}(t))$

$$\begin{aligned}\dot{\widehat{x}}(t) &= V_k^T A V_k\widehat{x}(t)+V_k^T A\overline{x}+V_k^T U_m(\mathscr{S}^T U_m)^{-1}\mathscr{S}^T\mathbf{f}(\overline{x}+V_k\widehat{x}(t))\\ &+ V_k^T(\mathbb{I}_n-\mathcal{D})\mathbf{f}(\overline{x}+V_k\widehat{x}(t))\end{aligned}$$

DEIM trick, Chaturantabut and Sorensen, 2010

If $\mathbf{f}$ is defined at a vector $x = (x_i)_{i=1}^n$ component-wise as[1]
$\mathbf{f}(x) = (\phi_1(x_1), \phi_2(x_2), \ldots, \phi_n(x_n))^T$ then

$$\mathscr{S}^T \mathbf{f}(\overline{x}+V_k\widehat{x}(t)) = \begin{pmatrix} \phi_{i_1}(\overline{x}_{i_1} + V_k(i_1,:)\widehat{x}(t)) \\ \phi_{i_2}(\overline{x}_{i_2} + V_k(i_2,:)\widehat{x}(t)) \\ \vdots \\ \phi_{i_m}(\overline{x}_{i_m} + V_k(i_m,:)\widehat{x}(t)) \end{pmatrix} \equiv \mathbf{f}_{\mathscr{S}}(\mathscr{S}^T\overline{x}+(\mathscr{S}^T V_k)\widehat{x}(t)),$$

the computational complexity of

$$V_k^T \mathcal{D}\mathbf{f}(\overline{x} + V_k\widehat{x}(t)) = (V_k^T U_m)(\mathscr{S}^T U_m)^{-1}\mathbf{f}_{\mathscr{S}}(\mathscr{S}^T\overline{x} + (\mathscr{S}^T V_k)\widehat{x}(t))$$

becomes independent of the dimension n, once the time independent matrices are precomputed in the off-line phase.

[1] For a general nonlinear $\mathbf{f}(x) = (\varphi_1(x_{\mathcal{I}_1}), \varphi_2(x_{\mathcal{I}_2}), \ldots, \varphi_n(x_{\mathcal{I}_n}))^T$, where $x_{\mathcal{I}_j}$ ($\mathcal{I}_j \subseteq \{1, \ldots, n\}$) denotes a sub-array of x needed to evaluate $\varphi_j(x)$, the situation is more complicated, see Chaturantabut and Sorensen, 2010.

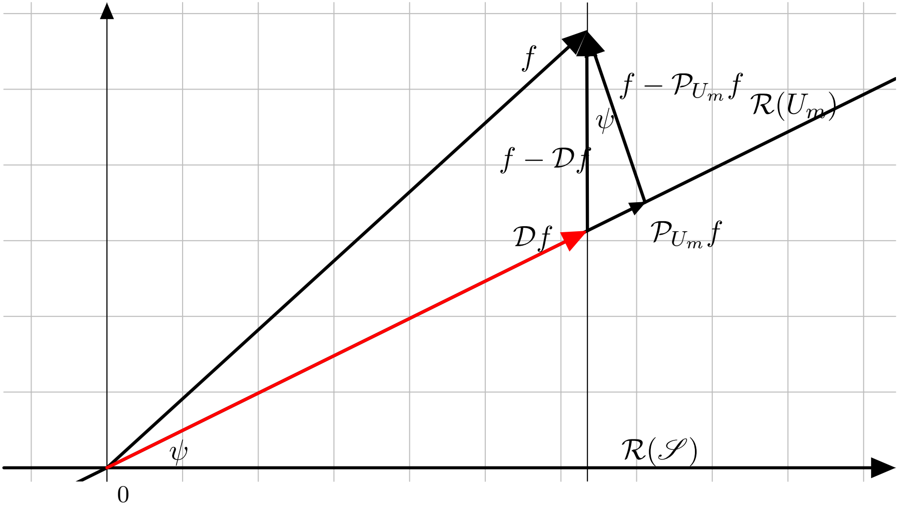


Figure: DEIM interpolatory projection

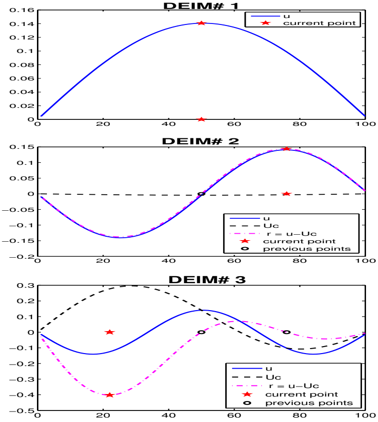
Discrete Empirical Interpolation Method (DEIM)

DEIM

INPUT: $u_1, \ldots, u_m \in \mathbb{C}^n$ (linearly independent)

OUTPUT: $\wp_1, \ldots, \wp_m$

- $[\rho \;\; \wp_1] = max|u_1|$
  $\mathsf{U} = [u_1]$, $\vec{\wp} = [\wp_1]$, $\mathscr{S} = [e_{\wp_1}]$
- for $j = 2$ to m
  1. $u \leftarrow u_j$
  2. Solve $(\mathscr{S}^T \mathsf{U})z = \mathscr{S}^T u$ for z
  3. $r = u - \mathsf{U}z$
  4. $[\rho \;\; \wp_j] = max\{|r|\}$
  5. $\mathsf{U} \leftarrow [\mathsf{U} \;\; u]$, $\vec{\wp} \leftarrow \begin{bmatrix} \vec{\wp} \\ \wp_j \end{bmatrix}$, $\mathscr{S} \leftarrow [\mathscr{S} \;\; e_{\wp_j}]$



- [Chaturantabut/Sorensen, 2010]
- Discrete version of EIM [Barrault/Maday/Nguyen/Patera, 2004]

EIM, PBDW, GEIM, MDEIM, QDEIM, WDEIM, UDEIM, NNDEIM

DEIM originates in the Empirical Interpolation Method (EIM) of Barrault, Nguye, Maday and Patera (2004). Most general framework of PBDW (Maday, Patera, Penn, Yano 2015). For a related discrete version of EIM see Haasdonk, Ohlberger and Rozza (2008)

DEIM has been successfully deployed in many applications, and tuned for better performance, giving rise to the localized DEIM (Peherstorfer, Butnaru, Willcox, Bungartz 2014), unassembled DEIM (UDEIM) (Tiso ad Rixen 2013), matrix DEIM (Wirtz, Sorensen, Kaasdonk 2014; Negri, Manzoni, Amsallem 2015), nonnegative DEIM (NNDEIM) (Amsallem, Nordström 2016), Q-DEIM (Drmač, Gugercin 2016), and WDEIM (Drmač, Saibaba 2017). The latter two are orthogonal variant of DEIM, that can be efficiently implemented with high-performance libraries such as LAPACK and ScaLAPACK. Furthermore, Q-DEIM and WDEIM admit a better condition number bounds; they allow randomized sampling.

Error, intuition, ...

Following Chaturantabut and Sorensen (2010), let

$$\widehat{\mathbf{f}} = \underbrace{\mathsf{U}(\mathscr{S}^T\mathsf{U})^{-1}\mathscr{S}^T}_{\mathcal{D}}\,\mathbf{f} \equiv \mathcal{D}\mathbf{f}, \quad f_* = \mathsf{U}\mathsf{U}^T\mathbf{f}$$

Note $\mathcal{D}\mathbf{f}_* = \mathbf{f}_*$. Then $\widehat{\mathbf{f}} = \mathcal{D}\mathbf{f} = \mathcal{D}(\mathbf{f} - \mathbf{f}_*) + \mathbf{f}_*$ and

$$\mathbf{f} - \widehat{\mathbf{f}} = (\mathbf{f} - \mathbf{f}_*) + \mathbf{f}_* - \mathcal{D}(\mathbf{f} - \mathbf{f}_*) - \mathbf{f}_* = (\mathbb{I}_n - \mathcal{D})(\mathbf{f} - \mathbf{f}_*).$$

Hence

$$\|\mathbf{f} - \mathbf{f}_*\|_2 \leq \|\mathbb{I}_n - \mathcal{D}\|_2 \|(\mathbb{I}_n - \mathsf{U}\mathsf{U}^T)\mathbf{f}\|_2 = \|\mathcal{D}\|_2 \|(\mathbb{I}_n - \mathsf{U}\mathsf{U}^T)f\|_2$$

Since $\|\mathcal{D}\|_2 = \|(\mathscr{S}^T\mathsf{U})^{-1}\|_2$, the index selection $\mathscr{S}$ should pick submatrix of the orthonormal basis U with small inverse.

Lemma (Chaturantabut/Sorensen, 2010)

Let $\mathsf{U} \in \mathbb{R}^{n\times m}$ *be orthonormal* ($\mathsf{U}^*\mathsf{U} = \mathbb{I}_m$, $m < n$) *and let*

$$\widehat{f} = \mathsf{U}(\mathscr{S}^T\mathsf{U})^{-1}\mathscr{S}^T f \tag{1.1}$$

be the DEIM projection $f \in \mathbb{R}^n$*, with* $\mathscr{S}$ *computed by DEIM. Then*

$$\|f - \widehat{f}\|_2 \le \mathbf{c}\|(\mathbb{I} - \mathsf{U}\mathsf{U}^*)f\|_2, \quad \mathbf{c} = \|(\mathscr{S}^T\mathsf{U})^{-1}\|_2, \tag{1.2}$$

where

$$\mathbf{c} \le \frac{(1+\sqrt{2n})^{m-1}}{\|u_1\|_\infty} \le \sqrt{n}(1+\sqrt{2n})^{m-1}.$$

- If $\mathcal{R}(\mathsf{U})$ captures the behavior of $\mathbf{f}$ well, and if $\mathscr{S}$ results in a moderate $\mathbf{c}$, the DEIM approximation will succeed.

Towards a different selection operator $\mathscr{S}$

- The error bound is rather pessimistic and DEIM performs drastically better than the bound predicts.
- $\mathscr{S}$ computed by DEIM depends on a particular basis for $\mathcal{U}$.
- The complexity of DEIM is $O(m^2 n) + O(m^3)$, and it has unfavorable flop per memory reference ratio (level 2 BLAS).
- Questions of interests:
  - Can the upper bound be improved and what selection operator $\mathscr{S}$ will have sharper a priori error bound?
  - Can we devise a selection operator $\mathscr{S}$ independent of the choice of an orthonormal basis U of $\mathcal{U}$?
  - Can we have an algorithm based on BLAS 3 building blocks?
  - Can we reduce the contribution of the factor n without substantial loss in the quality of the computed selection operator?

A new DEIM framework

Theorem (Z.D./Gugercin,2015)

Let $\mathsf{U} \in \mathbb{C}^{n\times m}$, $\mathsf{U}^*\mathsf{U} = \mathbb{I}_m$, $m < n$. *Then :*

- *There exists an algorithm to compute* $\mathscr{S}$ *with complexity* $O(nm^2)$ *s.t.*

$$\|(\mathscr{S}^T\mathsf{U})^{-1}\|_2 \leq \sqrt{n-m+1}\,\frac{\sqrt{4^m+6m-1}}{3}, \qquad (2.1)$$

  and for any $f \in \mathbb{C}^n$

$$\|f - \mathsf{U}(\mathscr{S}^T\mathsf{U})^{-1}\mathbb{P}^T f\|_2 \leq \sqrt{n}\,O(2^m)\,\|f - \mathsf{U}\mathsf{U}^* f\|_2. \qquad (2.2)$$

- *There exists a selection operator* $\mathscr{S}_\star$ *such that*

$$\|f - \mathsf{U}(\mathscr{S}_\star^T\mathsf{U})^{-1}\mathscr{S}_\star^T f\|_2 \leq \sqrt{1+m(n-m)}\,\|f - \mathsf{U}\mathsf{U}^* f\|_2. \qquad (2.3)$$

- *The selection operators* $\mathscr{S}$, $\mathscr{S}_\star$ *do not change if* U *is changed to* $\mathsf{U}\Omega$ *where* Ω *is arbitrary* $m\times m$ *unitary matrix.*

New approach – rank revealing QR factorization

- Proof is constructive and uses the ideas from [Drmač,2009], arising in the analysis of block Jacobi algorithm for diagonalization of Hermitian matrices.
- Selection strategy $\mathscr{S}$ simply amounts to the pivot selection in QR factorization with column pivoting of U* !!! Let

$$\mathsf{U}^*\Pi = QR = \left(\begin{array}{ccc|cccc} \divideontimes & \ast & \ast & \ast & \ast & \ast & \ast \\ 0 & \ast & \ast & \ast & \ast & \ast & \ast \\ 0 & 0 & \star & \ast & \ast & \ast & \ast \end{array}\right) = Q \begin{pmatrix} R_{[1]} & R_{[2]} \end{pmatrix}$$

be pivoted QR. Consider the Businger–Golub pivoting:

$$\begin{array}{c} \\ i \\ \\ m \end{array}\begin{array}{c} \begin{array}{ccccccc} & i & & p_i & & & n \end{array} \\ \begin{pmatrix} \star & {\color{blue}\star} & \star & {\color{red}\star} & \star & \star & \star \\ 0 & {\color{blue}\ast} & \ast & {\color{red}\circledast} & \ast & \ast & \ast \\ 0 & {\color{blue}\ast} & \ast & {\color{red}\circledast} & \ast & \ast & \ast \\ 0 & {\color{blue}\ast} & \ast & {\color{red}\circledast} & \ast & \ast & \ast \end{pmatrix} \end{array} \xrightarrow{swap(i,p_i)} \begin{array}{c} \\ i \\ \\ m \end{array}\begin{array}{c} \begin{array}{ccccccc} & i & & p_i & & & n \end{array} \\ \begin{pmatrix} \star & {\color{red}\star} & \star & {\color{blue}\star} & \star & \star & \star \\ 0 & {\color{red}\circledast} & \ast & {\color{blue}\ast} & \ast & \ast & \ast \\ 0 & {\color{red}\circledast} & \ast & {\color{blue}\ast} & \ast & \ast & \ast \\ 0 & {\color{red}\circledast} & \ast & {\color{blue}\ast} & \ast & \ast & \ast \end{pmatrix} \end{array}.$$

- $\mathscr{S}$: selection operator that collects the columns of W to build $R_{[1]}$;

Discrete Empirical Interpolation Method (DEIM)

DEIM

INPUT: $u_1, \ldots, u_m \in \mathbb{C}^n$ (linearly independent)

OUTPUT: $\wp_1, \ldots, \wp_m$

- $[\rho \;\; \wp_1] = max|u_1|$
  $\mathsf{U} = [u_1]$, $\vec{\wp} = [\wp_1]$, $\mathscr{S} = [e_{\wp_1}]$
- for $j = 2$ to m
  1. $u \leftarrow u_j$
  2. Solve $(\mathscr{S}^T \mathsf{U})z = \mathscr{S}^T u$ for z
  3. $r = u - \mathsf{U}z$
  4. $[\rho \;\; \wp_j] = max\{|r|\}$
  5. $\mathsf{U} \leftarrow [\mathsf{U} \;\; u]$, $\vec{\wp} \leftarrow \begin{bmatrix} \vec{\wp} \\ \wp_j \end{bmatrix}$, $\mathscr{S} \leftarrow [\mathscr{S} \;\; e_{\wp_j}]$

Q_DEIM, Z.D., S. Gugercin 2015.

```
function S = q_deim(U) ;
[~,~,P] = qr( U', 'vector' ) ;
S = P(1:size(U,2)) ;
end
```

Q_DEIM properties:

- simple, efficient, blocked, parallelizable, numerically robust code already available
- basis independent
- better error bounds
- close to optimal volume property
- randomized sampling QRCP enhanced version possible

Comments on rank revealing, volume maximization

- The existence of $\mathscr{S}_\star$ is due to Goreinov et al., 1997 and uses the concept of matrix volume (the absolute value of the determinant).
- $\mathscr{S}_\star$ is defined to be the one that maximizes the volume of $\mathscr{S}_\star^T$U over all $\binom{n}{m} = \frac{n!}{m!(n-m)!}$ $m \times m$ submatrices of U.
- Either $\mathscr{S}$ or $\mathscr{S}_\star$ does not change by a unitary transformation
- The volume of the submatrix selected by $\mathscr{S}$ equals the volume $\prod_{i=1}^{m} |R_{ii}|$ of the upper triangular R, And, pivoting, by design, at each step tries to produce maximal possible $|R_{ii}|$; thus it can be interpreted as a greedy volume maximizing scheme.
- Pioneering work on strong rank revealing QR by Chandrasekaran and Ipsen 1994, Gu and Eisenstat 1996.
- The idea of maximizing the determinant over al square submatrices goes back to Knuth (to find best spanning subset) 1985.

Numerical Implementation

- The new selection is called Q-DEIM
- It is still an interpolatory DEIM process, but with a different $\mathbb{P}$

```
function [ S, M ] = dime( U ) ;
% Input  : U n-by-m with orthonormal columns
% Output : S selection of m row indices with guaranteed upper bound
%          norm(inv(U(S,:))) <= sqrt(n-m+1) * O(2^m).
%        : M the matrix U*inv(U(S,:)); the DEIM projection of
%          n-by-1 f is M*f(S).
% Coded by Zlatko Drmac, April 2015.
[n,m] = size(U) ;
if nargout == 1
[~,~,P] = qr(U','vector') ; S = P(1:m)  ;
else
[Q,R,P] = qr(U','vector') ; S = P(1:m)  ;
M = [eye(m) ; (R(:,1:m)\R(:,m+1:n))']   ;
Pinverse(P) = 1 : n ; M = M(Pinverse,:) ;
end
end
```

Using restricted/randomized basis information

- If n is gargantuan, it will be necessary to reduce the $O(m^2 n)$ factor
- We only need to ensure that $\mathsf{T} = \mathbb{R}(1:m, 1:m)$ has small inverse where T is the pivoted QR triangular factor of columns of W.
- Use only a small selection of the columns of W (the rows of U):
- Randomly pick $k \geq m$ columns and store them in L:

$$
\begin{array}{c}
\begin{array}{ccccccccccccccccccc}
\Uparrow & \uparrow & & & \uparrow & & \uparrow & & \uparrow & \Uparrow & & & \Uparrow & & \uparrow & & \uparrow & & \Uparrow
\end{array}\\
\begin{pmatrix}
\star & * & . & . & * & . & * & . & * & \star & . & . & \star & . & * & . & * & . & \star \\
\star & * & . & . & * & . & * & . & * & \star & . & . & \star & . & * & . & * & . & \star \\
\star & * & . & . & * & . & * & . & * & \star & . & . & \star & . & * & . & * & . & \star \\
\star & * & . & . & * & . & * & . & * & \star & . & . & \star & . & * & . & * & . & \star
\end{pmatrix}
\end{array}
\mapsto
\overbrace{\begin{pmatrix}
* & * & * & * & * & * \\
* & * & * & * & * & * \\
* & * & * & * & * & * \\
* & * & * & * & * & *
\end{pmatrix}}^{\mathsf{L}}
$$

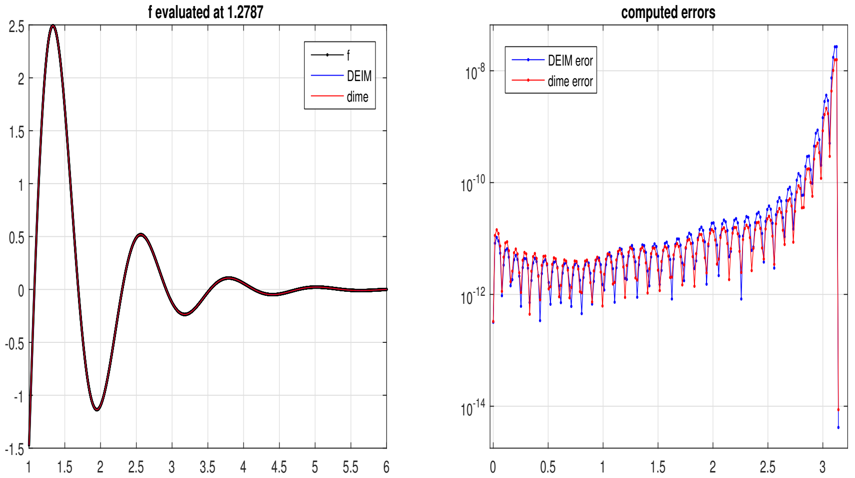
- Apply QR with column pivoting on L with a built-in Incremental Condition Estimator (ICE) that estimates $\|\mathsf{L}(1:j, 1:j)^{-1}\|$
- Define a threshold for the inverse.

$$\begin{pmatrix} * & * & \times & \times & \times & \times \\ 0 & * & \times & \times & \times & \times \\ 0 & 0 & \circledast & \odot & \odot & \odot \\ 0 & 0 & 0 & \odot & \odot & \odot \end{pmatrix} \rightsquigarrow \begin{pmatrix} * & * & \star & \star & \star & \star \\ 0 & * & \star & \star & \star & \star \\ 0 & 0 & \star & \star & \star & \star \\ 0 & 0 & \star & \star & \star & \star \end{pmatrix} \rightsquigarrow \begin{pmatrix} * & * & * & \star & \star & \star \\ 0 & * & * & \star & \star & \star \\ 0 & 0 & * & \star & \star & \star \\ 0 & 0 & 0 & \star & \star & \star \end{pmatrix}$$

- If $\|\mathsf{L}(1:j,1:j)^{-1}\|$ is below threshold, continue.
- If not, the (j,j)th position $\circledast$ is too small, and, due to pivoting, that all entries in the active submatrix of L ($\odot$ are also small. ($\circledast$)
- The columns j to k in L are useless; discard them
- Draw new $k-j+1$ columns from the active columns of W ($\overset{\Uparrow}{\star}$).
- At any point, only k columns are processed.
- Algorithm is called `dimer`.

Example 3

- $\mathbf{f}(t;\mu) = 10e^{-\mu t}(\cos(4\mu t) + \sin(4\mu t))$, $1 \leq t \leq 6$, $0 \leq \mu \leq \pi$.
- Take 40 uniformly μ sample and compute the snapshots over the discretized t–domain at $n = 10000$ uniformly spaced nodes.
- The best low rank approximation returned U with $m = 34$ columns.
- Let $k = m$ columns in the work array L, and set the upper bound for $\mathbf{c}$ at $\sqrt{m}\sqrt{n-m+1}$.
- Column index drawing is “random”.
- After processing 113 rows of U (out of 10000), `dimer` selected a submatrix with $\mathbf{c} \approx 181.45$;
- DEIM processed the whole matrix U and returned $\mathbf{c} \approx 79.13$.


f evaluated at 1.2787
f
DEIM
dime
computed errors
DEIM eror
dime error

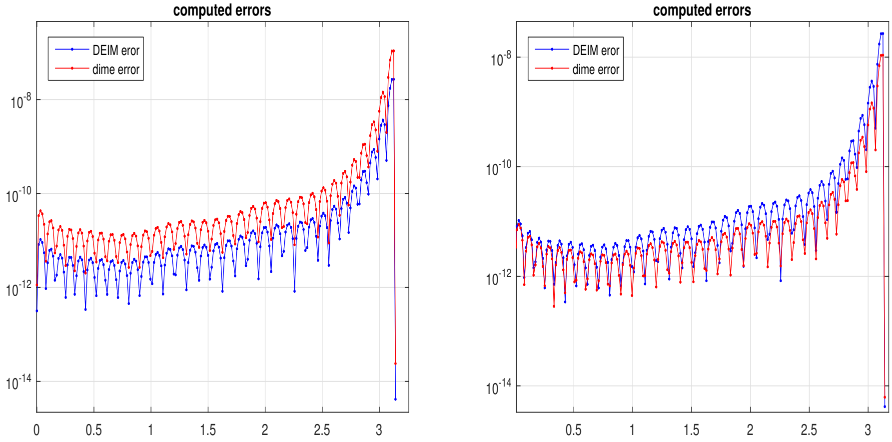


Figure: Left figure: Upper bound in `dimer` set to $m\sqrt{n-m+1}$; it used 53 rows with $\mathbf{c} \approx 2532.9$. Right figure: Upper bound in `dimer` set to $\sqrt{m}\sqrt{n-m+1}/5$; it used 220 rows with $\mathbf{c} \approx 103.1$.

Q-DEIM with strong rank revealing pivoting

Connection

Strong rank revealing pivoting in the QR factorization is a volume maximization scheme.

Theorem (Z.D. and A. K. Saibaba)

Let $V \in \mathbb{R}^{n \times n}$ with $V^T V = \mathbb{I}_m$. Applying SRRQR of Gu and Eisenstat with target rank m and tuning parameter $\eta \geq 1$ to V^T gives a submatrix $\mathscr{S} \in \mathbb{R}^{n \times m}$ of $\mathbb{I}_n$ with

$$\frac{1}{\sqrt{1 + \eta^2 m(n-m)}} \leq \sigma_j(\mathscr{S}^T V) \leq 1, \qquad 1 \leq j \leq m,$$

and $1 \leq \|(\mathscr{S}^T V)^{-1}\|_2 \leq \sqrt{1 + \eta^2 m(n-m)}$.

η can be taken as e.g. $\eta = \sqrt{n}$ or $\eta = \sqrt{m}$; smaller η requires more work.

Canonical structure of the DEIM projector

$$\mathcal{D} = U_m(\mathscr{S}^T U_m)^{-1}\mathscr{S}^T = (\mathcal{P}_{\mathscr{S}}\mathcal{P}_{U_m})^{\dagger},$$

Let now $\mathscr{S}^T U_m \in \mathbb{R}^{s\times m}$ be a rectangular matrix where $s \neq m$.

$$(\mathcal{P}_{\mathscr{S}}\mathcal{P}_{U_m})^{\dagger} = U_m(\mathscr{S}^T U_m)^{\dagger}\mathscr{S}^T. \qquad (3.1)$$

Case $\operatorname{rank}(\mathcal{D}) = m < s$

1. The interpolation property does not hold, i.e., $\mathscr{S}^T(\mathcal{D}f) \neq \mathscr{S}^T f$. This is because $(\mathscr{S}^T U_m)^{\dagger}$ is no longer a right multiplicative inverse. However, $\mathscr{S}^T(\mathcal{D}f)$ is the least square projection of $\mathscr{S}^T f$ onto the range of $\mathscr{S}^T U_m$. To see this

$$\mathscr{S}^T(\mathcal{D}f) = \mathscr{S}^T U_m(\mathscr{S}^T U_m)^{\dagger}\mathscr{S}^T f = \mathcal{P}_{\mathcal{X}}(\mathscr{S}^T f), \;\; \mathcal{X} = \mathcal{R}(\mathscr{S}^T U_m).$$

2. In this case $\mathcal{D}\mathcal{P}_{U_m} = \mathcal{P}_{U_m}$ since $(\mathscr{S}^T U_m)^{\dagger}$ is a left multiplicative inverse of $\mathscr{S}^T U_m$.

Canonical structure of the DEIM projector

Let $U_m \in \mathbb{R}^{n\times m}$, $\mathscr{S} \in \mathbb{R}^{n\times s}$ be orthonormal, $\mathcal{D} = U_m(\mathscr{S}^T U_m)^\dagger \mathscr{S}^T$ and assume that $1 \leq m, s \leq n$. Let $\ell \equiv \dim(\mathcal{R}(\mathscr{S}) \bigcap \mathcal{R}(U_m))$, $p \equiv \mathrm{rank}(\mathcal{D}) - \ell$, and let the singular values $\sigma_i = \cos\psi_i$ of $\mathscr{S}^T U_m$ be

$$1 = \sigma_1 = \ldots = \sigma_\ell > \sigma_{\ell+1} \geq \ldots \geq \sigma_{\ell+p} > \sigma_{\ell+p+1} = \ldots = \sigma_{\min(m,s)} = 0.$$

(Here $0 < \psi_{\ell+1} \leq \ldots \leq \psi_{\ell+p} < \pi/2$ are the acute principal angles between the ranges of $\mathscr{S}$ and U_m.)
(i) There exists an orthogonal $n \times n$ matrix Z such that the matrix $\mathcal{D}$ can be represented as

$$\mathcal{D} = (\mathcal{P}_{\mathscr{S}}\mathcal{P}_{U_m})^\dagger = Z \begin{pmatrix} \mathbb{I}_\ell & & \\ & \bigoplus_{i=1}^{p} T_i & \\ & & \mathbf{0}_{n-\ell-2p} \end{pmatrix} Z^T, \quad T_i = \begin{pmatrix} 1 & 0 \\ \tan\psi_{\ell+i} & 0 \end{pmatrix}.$$

(ii) The DEIM projector $\mathcal{D}$ satisfies $\|\mathcal{D}\|_2 = 1/\cos\psi_{\ell+p}$. If, in addition, $\mathcal{D} \neq 0$ and $\mathcal{D} \neq \mathbb{I}_n$, then $\|\mathcal{D}\|_2 = \|\mathbb{I}_n - \mathcal{D}\|_2 = 1/\cos\psi_{\ell+p}$.

Canonial structure of a pair of projections (Wedin 1982)

$$\mathcal{P}_{\mathscr{S}} = Z \begin{pmatrix} \mathbb{I}_\ell & & \\ & \bigoplus_{i=1}^{p} J_i & \\ & & D_s \end{pmatrix} Z^T, \quad \text{where } J_i = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \begin{pmatrix} 1 & 0 \end{pmatrix}, \quad \text{and}$$

$$\mathcal{P}_{U_m} = Z \begin{pmatrix} \mathbb{I}_\ell & & \\ & \bigoplus_{i=1}^{p} \Psi_i & \\ & & D_u \end{pmatrix} Z^T, \quad \Psi_i = \begin{pmatrix} \cos\psi_{\ell+i} \\ \sin\psi_{\ell+i} \end{pmatrix} \begin{pmatrix} \cos\psi_{\ell+i} & \sin\psi_{\ell+i} \end{pmatrix},$$

with $\psi_{\ell+i}$'s as stated in the theorem, and D_s, D_u are diagonal matrices with diagonal entries 0 or 1 and such that $D_s D_u = \mathbf{0}$.

$$(\mathcal{P}_{\mathscr{S}}\mathcal{P}_{U_m})^\dagger = Z \begin{pmatrix} \mathbb{I}_\ell & & \\ & \bigoplus_{i=1}^{p} (J_i\Psi_i)^\dagger & \\ & & \mathbf{0} \end{pmatrix} Z^T.$$

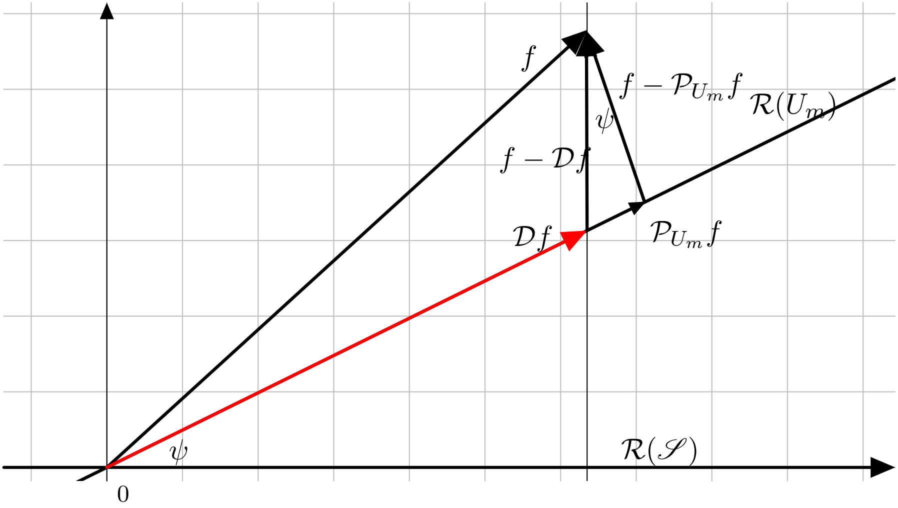


Figure: The nontrivial action of DEIM projection consists of $\dim(\mathcal{R}(\mathscr{S}) \bigcap \mathcal{R}(U_m))$–dimensional identity and $\mathrm{rank}(\mathcal{D}) - \dim(\mathcal{R}(\mathscr{S}) \bigcap \mathcal{R}(U_m))$ 2–dimensional oblique (interpolatory) projections as shown in the figure.

Connection to the Cosine Sine Decomposition (CSD)

The structure of $\mathcal{D}$ can be also analyzed using the CSD. Assume for simplicity that the rows of U_m are ordered so that $\mathscr{S} = \mathbb{I}_n(:,1:m)$. If this not the case, we work with $\Pi^T \mathcal{D} \Pi$, where Π is a permutation matrix. Assume that $\mathscr{S}^T U_m$ is invertible and therefore, the DEIM operator is $\mathcal{D} = U_m(\mathscr{S}^T U_m)^{-1}\mathscr{S}^T$. Further, let $\mathscr{S}_\perp = \mathbb{I}_n(:,m+1:n)$.
With these assumptions, U_m has the CS decomposition

$$U_m = \begin{pmatrix} \mathscr{S}^T U_m \\ \mathscr{S}_\perp^T U_m \end{pmatrix} = \begin{pmatrix} W_1 & \\ & W_2 \end{pmatrix} \begin{pmatrix} \mathrm{Cos}\Psi \\ \mathrm{Sin}\Psi \end{pmatrix} V^T .$$

Here $W_1, V \in \mathbb{R}^{m\times m}$, $W_2 \in \mathbb{R}^{(n-m)\times(n-m)}$ are orthogonal matrices and

$$\mathrm{Cos}\Psi = \mathrm{diag}(\cos\psi_i)_{i=1}^m \in \mathbb{R}^{m\times m}, \quad \mathrm{Sin}\Psi = \mathrm{diag}(\sin\psi_i)_{i=1}^m \in \mathbb{R}^{(n-m)\times m}.$$

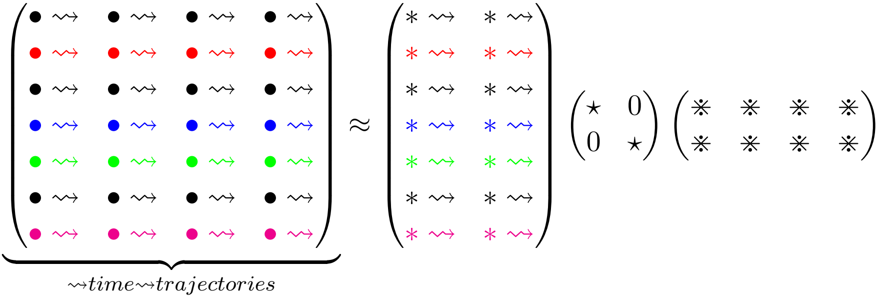
We can therefore represent $\mathcal{D}$ as

$$\mathcal{D} = \begin{pmatrix} W_1 \,\mathrm{Cos}\Psi\, V^T \\ W_2 \,\mathrm{Sin}\Psi\, V^T \end{pmatrix} V\,(\mathrm{Cos}\Psi)^{-1}\, W_1^T \begin{pmatrix} \mathbb{I}_m & \mathbf{0} \end{pmatrix} = \begin{pmatrix} \mathbb{I}_m & \mathbf{0} \\ \mathrm{Tan}\Psi & \mathbf{0} \end{pmatrix},$$

where $\mathrm{Tan}\Psi = W_2 \mathrm{Sin}\Psi (\mathrm{Cos}\Psi)^{-1} W_1^T = W_2 \mathrm{diag}(\tan\psi_i)_{i=1}^m W_1^T$.

Another point of view – a surprising connection (or not?!)

Consider the snapshots matrix $X \approx U\Sigma V^T$, $X(:,i) = x(t_i)$, $t_{i+1} = t_i + \delta t$



- representative trajectories $\bullet \rightsquigarrow$, $\bullet \rightsquigarrow$, $\bullet \rightsquigarrow$, $\bullet \rightsquigarrow$
- representative projected trajectories $* \rightsquigarrow$, $* \rightsquigarrow$, $* \rightsquigarrow$, $* \rightsquigarrow$ represented in the basis of ΣV^T (or V^T and Σ carries weights)
- find representatives by clustering; simplest: k-means

K-means based K-DEIM

```
function S = K_DEIM( U ) ;
% Input  : U n-by-m real matrix [with orthonormal columns]
% Output : S m-by-1 integer vector -- selection of m row indices
%          for an interpolatory DEIM oblique projection
% Coded by Zlatko Drmac, April 2017.
[~,m] = size(U) ; S = zeros(m,1) ;
[IDX, C] = kmeans( U, m ) ; % use the default Euclidean metric
for i = 1 : m
   J = find(IDX == i) ; % select the i-th cluster
   XX = U(J,:) - ones(length(J),1)*C(i,:) ;
   [~, is] = min( sum( XX.*XX, 2 )./sum( (U(J,:).^2), 2 ) ) ;
   S(i) = J(is(1)) ;  % cluster representative closest to centroid
end
end
```

k-means based interpolation indicies for DEIM. Clustering is done in the Euclidean metric for simplicity; other choices are possible. Cluster representative is the one closest to the cluster centroid.

An Example: The FitzHugh-Naguma (F–N) System

The F–N system used in modeling the dynamics of a spiking neuron. Following the notation of Chaturantabut/Sorensen,2010, we let v and w denote, respectively, the voltage and recovery of voltage. The coupled nonlinear PDEs

$$\varepsilon v_t(x,t) = \varepsilon^2 v_{xx}(x,t) + f(v(x,t)) - w(x,t) + c \quad (4.1)$$

$$w_t(x,t) = bv(x,t) - \gamma w(x,t) + c \quad (4.2)$$

where $x \in [0, L]$ and $t \geq 0$, with the initial and boundary conditions

$$\begin{array}{lll} v(x,0) = 0, & w(x,0) = 0, & x \in [0,L], \\ v_x(0,t) = -i_0(t), & v_x(L,t) = 0, & t \geq 0, \end{array}$$

describe the underlying dynamics. The nonlinearity is due to the term $f(v) = v(v - 0.1)(1 - v)$. Let $L = 1$, $\varepsilon = 0.015$, $b = 0.5$, $\gamma = 2$, $c = 0.05$ and the stimulus $i_0(t) = 50000t^3e^{-15t}$. A finite difference discretization yields a semidiscretized system of dimension $n = 2048$. We simulate the system for $t = [0, 8]$ to obtain the state and nonlinear snapshots.

F-N System Example - quality of reconstruction

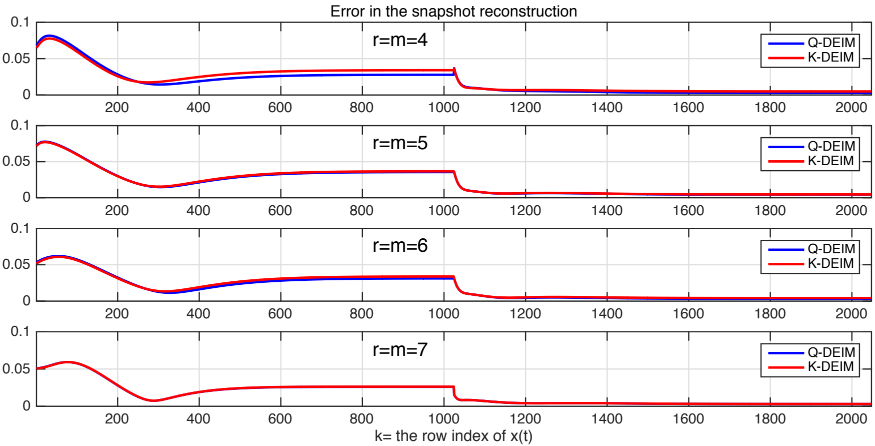


Figure: F–N System Example: Comparison of the relative error in the snapshot reconstruction for the F-N model for different r and m values. The horizontal axis variable k corresponds to the k^{rh} row of $x(t)$ for which the relative error is computed.

Why it works? What k-means does? Connection to DEIM?

Partition $\pi = (\pi_1, \ldots, \pi_m) :: \bigcup_{i=1}^{m} \pi_i = \{1, \ldots, n\}$, $\pi_i \bigcap \pi_j = \emptyset$ for $i \neq j$. Let $\sharp\pi_i$ denote the cardinality of π_i. k-means tries to minimize

$$\mathfrak{F}(\pi) = \sum_{i=1}^{m} \sum_{j \in \pi_i} \|a_j - c_i\|_2^2, \quad \text{where} \quad c_i = \frac{1}{\sharp\pi_i} \sum_{j \in \pi_i} a_j .$$

$$\mathcal{C}_{ij} = \begin{cases} 1/\sqrt{\sharp\pi_j} & \text{if the } i\text{th point belongs to the } j\text{th cluster} \\ 0 & \text{otherwise} \end{cases}$$

$$\mathfrak{F}(\pi) \equiv \mathfrak{F}(\mathcal{C}) = \mathrm{trace}(A^T A) - \mathrm{trace}(\mathcal{C}^T A^T A \mathcal{C}) \longrightarrow \min$$

$$\mathrm{trace}(\mathcal{C}^T A^T A \mathcal{C}) \longrightarrow \max_{\mathcal{C}}$$ (combinatorial, NP hard)

Relaxation: use $\mathcal{C}^T\mathcal{C} = \mathbb{I}_m$, relax discrete structure $\looparrowright$ POD basis

$$\mathrm{trace}(U^T A^T A U) \longrightarrow \max_{U}, \quad U^T U = \mathbb{I}_m$$

Relaxed k-means and POD + Q-DEIM

After computing POD basis U that solves $\mathrm{trace}(U^T A^T A U) \rightarrow \max_U$ we need to discretize U and turn it into $\mathcal{C}$. Need orthogonal Q such that $U \leftarrow UQ$ is as discrete/sparse as possible.
H. Zha, X. He, C. Ding, M. Gu, H. Simon use pivoted QR factorization $U^T P = Q(R_{[11]},\ R_{[12]})$ and then discretize (heavy $\sin\Theta$ argument)

$$\widetilde{R} \equiv R_{[11]}^{-1} R = \begin{pmatrix} \mathbb{I}_m & R_{[11]}^{-1} R_{[12]} \end{pmatrix} = \begin{pmatrix} \bullet & & & & \cdot & \cdot & \cdot & \cdot & \bullet \\ & \bullet & & & \bullet & \cdot & \cdot & \cdot & \cdot \\ & & \bullet & & \cdot & \cdot & \cdot & \cdot & \cdot \\ & & & \bullet & \cdot & \bullet & \bullet & \cdot & \cdot \end{pmatrix}.$$

Simple intuition: most independent columns of U^T selected and used as a basis. Motivates how to pick extra points. No need fo $\sin\Theta$ theorem. This connection opens more interesting questions and gives ideas to enhance the DEIM framework; e.g. by introducing kernels.

You've gotta ask yourself one question: do I feel I use the right inner product?

• Hilbert space with the inner product $(f,g)_{L^2(\Omega)} = \int_\Omega f(x)g(x)\rho(x)dx$, Discrete (finite n-dimensional) framework yield a weighted inner product in $\mathbb{R}^n$, $(u,v)_W = v^T W u$, where W is the corresponding symmetric positive definite matrix. Then the natural framework for devising e.g. a POD approximation is given by the Hilbert space structure of $(\cdot,\cdot)_W$.

• Further, for the equations of e.g. compressible fluid flow, Galerkin projection in an $(\cdot,\cdot)_{L^2(\Omega)}$ inner product may not preserve the underlying physics, such as energy conservation or stability, see e.g. Rowley 2005.

• Different inner products may yield substantially different results, see e.g. Freund and Colonious 2002, Kalashnikova and Arunajatesan 2014.

• In model order reduction, for instance, a Galerkin projection may be naturally defined in a Lyapunov inner product, generated by the positive definite solution W of a Lyapunov matrix equation. For further examples and in-depth discussion see the work of Rowley, Kalashnikova et al, Barone et al, Farhat et al, Calo eta al, Zimmerman, Willcox, Noack et al ...

Weighted inner product

$W = LL^T$ SPD defines $(u,v)_W \equiv v^T W u$.

- weighted norm $\|u\|_W \equiv \sqrt{(u,u)_W} = \sqrt{u^T W u} = \|L^T u\|_2$,
- the induced operator norm of an $M \in \mathbb{R}^{n\times n}$ equals

$$\|M\|_W = \max_{x\neq 0} \frac{\|Mx\|_W}{\|x\|_W} = \max_{y\neq 0} \frac{\|L^T M L^{-T} y\|_2}{\|y\|_2} = \|L^T M L^{-T}\|_2.$$

- Further, in the W-inner product space, the adjoint of M is $M^{[T]} \equiv W^{-1} M^T W$, where M^T is the transpose of M.
- Weighted POD of $Y \in \mathbb{R}^{n\times n_s}$ (see e.g. Volkwein POD tutorial)
  1. Compute the thin SVD $L^T Y = U \Sigma V^T$.
  2. Determine an appropriate index $1 \leq m \leq \mathrm{rank}(L^T Y)$ and select $U_m \equiv U(:,1:m)$. The weighted POD basis is $\widehat{U} \equiv L^{-T} U_m$.
- Young-Mirsky theorem on best low rank approximation applies

Preliminaries

- Weighted POD computes a matrix $\widehat{U}$ whose columns are W-orthonormal, i.e., $\widehat{U}^T W \widehat{U} = \mathbb{I}_m$, and the POD projection in the weighted inner product space is represented by

$$\widehat{\mathcal{P}}_{\widehat{U}} \equiv \widehat{U}\widehat{U}^T W = L^{-T} U_m U_m^T L^T. \tag{5.1}$$

- Note that $\widehat{\mathcal{P}}_{\widehat{U}}^2 = \widehat{\mathcal{P}}_{\widehat{U}}$ and that $\widehat{\mathcal{P}}_{\widehat{U}}^{[T]} = \widehat{\mathcal{P}}_{\widehat{U}}$. In fact, $Y = \widehat{U}\Sigma V^T$ is a GSVD (Van Loan 1976) of Y.
- For $\mathbb{R}^{n\times m} \ni \widehat{U} : (\mathbb{R}^m, (\cdot,\cdot)_2) \longrightarrow (\mathbb{R}^n, (\cdot,\cdot)_W)$, the adjoint matrix in the two inner products is, by definition, given as $\widehat{U}^{<T>} = \widehat{U}^T W$. Hence, $\widehat{U}^{<T>}\widehat{U} = \mathbb{I}_m$ and we can write the W-orthogonal projector (5.1) conveniently in the usual form as $\widehat{\mathcal{P}}_{\widehat{U}} = \widehat{U}\widehat{U}^{<T>}$.

W-(Q)DEIM

With $(\cdot,\cdot)_W$–POD, one needs to define an appropriate DEIM projection operator in this weighted setting.

Definition (W-DEIM)

Let $\widehat{U} \in \mathbb{R}^{n\times m}$ be W-orthogonal. With a full column rank **generalized selection operator** $\mathbb{P} \in \mathbb{R}^{n\times s}$ $(s \geq m)$, define W-DEIM projector

$$\mathcal{D} \equiv \widehat{U}(\mathbb{P}^{<T>}\widehat{U})^{\dagger}\mathbb{P}^{<T>} = \widehat{U}(\mathbb{P}^T W\widehat{U})^{\dagger}\mathbb{P}^T W. \quad (5.2)$$

Proposition

If $\mathbb{P}^T W\widehat{U}$ is of full column rank, then $\|f - \mathcal{D}f\|_W \leq \|\mathcal{D}\|_W \|f - \widehat{\mathcal{P}}_{\widehat{U}} f\|_W$.

caveat: W may be ill-conditioned; naive approach may fail

Note that $\|\mathcal{D}\|_W \leq \sqrt{\kappa_2(W)}\|\mathcal{D}\|_2$. **Need proper selection operator $\mathbb{P}$.**

A choice of $\mathbb{P}$

Definition

Let $\widehat{U} = L^{-T} U_m$ as in W-POD. Define

$$\mathbb{P}^T = \mathscr{S}^T L^{-1}, \;\; \mathcal{D} \equiv \widehat{U}(\mathscr{S}^T U_m)^{\dagger} \mathscr{S}^T L^T = L^{-T} U_m (\mathscr{S}^T U_m)^{\dagger} \mathscr{S}^T L^T,$$

where $\mathscr{S}$ is an $n \times s$ index selection operator (s selected columns of the identity $\mathbb{I}_n$, $s \geq m$).

Proposition

$\mathbb{P}^T W \mathbb{P} = \mathbb{I}_s$ and $\|D\|_W = \|(\mathscr{S}^T U_m)^{\dagger}\|_2$.

Theorem (For $\mathscr{S}$ from an SRRQR QDEIM(η) applied to U_m)

$$\|f - \mathcal{D} f\|_W \leq \sqrt{1 + \eta^2 n(n-m)} \|f - \widehat{\mathcal{P}}_{\widehat{U}} f\|_W .$$

Great!

Wait, wait! And what about the interpolation?

... gone with the W

Let $\mathbb{P}^T W \widehat{U}$ be invertible. Then $\mathbb{P}^T W \mathcal{D} f = \mathbb{P}^T W f$. If $\mathbb{P} = L^{-T} \mathscr{S}$,

$$\mathscr{S}^T (L^T \mathcal{D} f) = \mathscr{S}^T (L^T f).$$

$$\mathscr{S}^T \begin{pmatrix} \ell_1^T \mathcal{D} f \\ \vdots \\ \ell_n^T \mathcal{D} f \end{pmatrix} = \mathscr{S}^T \begin{pmatrix} \ell_1^T f \\ \vdots \\ \ell_n^T f \end{pmatrix}, \quad \text{i.e.,} \quad \ell_{i_j}^T \mathcal{D} f = \ell_{i_j}^T f, \quad j = 1, \ldots, m.$$

GEIM: Y. Maday, O. Mula, A. T. Patera. M, Yano 2013., 2014., 2015

Functions may not be available through point evaluation either because there is no analytical expression or they may be sensor data corrupted by noise. In those cases, pointwise interpolation may not be possible, nor desirable. GEIM uses linear functionals selected from a dictionary.

So, we have a DGEIM. And if we insist on interpolation ..

The point interpolation corresponds to the point evaluation functional, $(\ell_i)_j = W_{ji} = W_{ij} = (\ell_j)_i = \delta_{ij}$, where δ_{ij} is the Kronecker delta.

Definition

Let the weighted selection operator $\mathbb{P}$ and the corresponding W-DEIM projector $\mathcal{D}$, respectively, be defined as

$$\mathbb{P}^T \equiv \mathscr{S}^T W^{-1} \qquad \mathcal{D} \equiv \widehat{U}(\mathscr{S}^T \widehat{U})^{-1} \mathscr{S}^T.$$

Here $\widehat{U}$ is W-orthogonal and $\mathscr{S}$ has columns from the identity matrix $\mathbb{I}_n$

It is easily checked that $\mathscr{S}^T \mathcal{D} f = \mathscr{S}^T f$.

Theorem

If $\mathcal{D}$ is as above, with $\mathscr{S}$ from SRRQR QDEIM applied to orth$(\widehat{U})$

$$\|f - \mathcal{D} f\|_W \leq \sqrt{1 + \eta^2 m(n-m)} \sqrt{\kappa_2(W)} \|f - \widehat{\mathcal{P}}_{\widehat{U}} f\|_W .$$

Weighted W-scaling independent POD+QDEIM

Want to reduce the impact of $\kappa_2(W)$ to the DEIM projection error.

Algorithm $([\widehat{U}, \mathscr{S}, Q_{\widehat{U}}, \Delta, \widehat{\Delta}] = W\text{-}\Delta\text{-POD-QDEIM}(Y, W \equiv LL^T, \eta)\ )$

Input: *Snapshots* $Y \in \mathbb{R}^{n\times n_s}$, $n_s < n$. *SPD* $W \in \mathbb{R}^{n\times n}$. $\eta \geq 1$.

1: *Compute the thin SVD of* $L^T Y$ *as* $L^T Y = U\Sigma V^T$.
   $\{Y = (L^{-T}U)\Sigma V^T$ *is a GSVD of* Y, *with* W*-orthogonal* $L^{-T}U$.$\}$

2: *Determine an appropriate index* m *and define* $U_m = U(:, 1:m)$.

3: $\Delta = \mathrm{diag}(\sqrt{W_{ii}})_{i=1}^n$; $L_s = \Delta^{-1}L$.

4: *Compute the thin QR factorization of* $L_s^{-T}U_m$ *as* $L_s^{-T}U_m = Q_{\widehat{U}} R_s$.

5: *Apply strong RRQR*(η) *to* $Q_{\widehat{U}}^T$ *to give*

$$Q_{\widehat{U}}^T \begin{pmatrix} \Pi_1 & \Pi_2 \end{pmatrix} = Q \begin{pmatrix} R_{11} & R_{22} \end{pmatrix}, \quad \Pi = \begin{pmatrix} \Pi_1 & \Pi_2 \end{pmatrix}.$$

6: $\mathscr{S} = \Pi_1$; $\widehat{\Delta} = \mathrm{diag}(\mathscr{S}^T \mathrm{diag}(W))$.

Output: $\mathcal{D} = \widehat{U}(\mathscr{S}^T\widehat{U})^{-1}\mathscr{S}^T \equiv \Delta^{-1}Q_{\widehat{U}}(\mathscr{S}^T Q_{\widehat{U}})^{-1}\mathscr{S}^T\Delta$.

Theorem (Z.D., A.K. Saibaba 2017.)

Assume that the DEIM projection operator $\mathcal{D}$ is defined as in Algorithm W-Δ-POD-QDEIM. Let $W_s = L_s L_s^T$ $((W_s)_{ii} = 1)$. Then

$$\|f - \mathcal{D}f\|_W \leq \sqrt{1+\eta^2 m(n-m)}\sqrt{\kappa_2(W_s)}\|f - \widehat{\mathcal{P}}_U f\|_W .$$

Comments:

- QR with Businger-Golub pivoting (`xGEQP3` in LAPACK, `PxGEQPF` in ScaLAPACK, `qr` in Matlab) can be used with confidence; worst case $\sqrt{n}2^m$ unlikely. Randomized approach feasible.
- thin QRF can be computed efficiently (Demmel, Grigori, Hoemmen Langou 2012)
- Van der Sluis: $\kappa_2(W_s) \leq n \min_{D=\mathrm{diag}} \kappa_2(DWD)$.
  Hence, if W is diagonal, or diagonally dominant (possibly graded) its condition number dos not affect the error bound.

Numerical experiments on synthetic examples show even better behavior.

Conclusions and Future Work

- New applications of QDEIM, W-QDEIM; already proved its effectiveness in applications such as optimal sensor placement; see e.g. results related to fluid flow in Manohar, Bingni, Brunton, Kutz, Brunton (Data driven sparse sensor placement, 2017)
- Ongoing project - applications of DEIM in fluid flow analysis (DMD and Koopman spectral analysis).
- Work in progress:

Z. Drmač and A. K. Saibaba, *The Discrete Empirical Interpolation Method: Canonical Structure and Formulation in Weighted Inner Product Spaces*, SIAM Journal on Matrix Analysis and Applications (in review)

Z. Drmač and S. Gugercin, *Discrete Empirical Interpolation Method: From Gappy POD to DEIM to Clustering*, preprint 2017.

References

- M. Barrault, N. C. Nguyen, Y. Maday, and A. T. Patera, *An empirical interpolation method: Application to efficient reduced-basis discretization of partial differential equations*, C. R. Acad. Sci. Paris, Série I., 339 (2004), pp. 667–672.
- S. Chaturantabut and D. Sorensen, *Nonlinear model reduction via discrete empirical interpolation*, SIAM J. Sci. Comp. (2010).
- Z. Drmač and S. Gugercin, *A new selection operator for the Discrete Empirical Interpolation Method – improved a priori error bound and extensions*, SIAM J. Sci. Comp. (2016).
- C. Farhat, T. Chapman, and P. Avery, *Structure-preserving, stability, and accuracy properties of the energy-conserving sampling and weighting method for the hyper reduction of nonlinear finite element dynamic models*, International Journal for Numerical Methods in Engineering, 102 (2015), pp. 1077–1110.
- S. A. Goreinov, E. E. Tyrtyshnikov, and N. L. Zamarashkin, *A theory of pseudoskeleton approximations*, Linear Algebra Appl., 261 (1997), pp. 1–21.
- M. Gu and S. C. Eisenstat, *Efficient algorithms for computing a strong rank-revealing QR factorization*, SIAM J. Sci. Comput., 17 (1996), pp. 848–869.

References

I. Kalashnikova and S. Arunajatesan, *A stable Galerkin reduced order modeling (ROM) for compressible flow*, in 10th Wold Congress on Computational Mechanics, Blucher Mechanical Engineering Proceedings, May 2014.

Y. Maday and O. Mula, *A generalized empirical interpolation method : Application of reduced basis techniques to data assimilation*, in Analysis and Numerics of Partial Differential Equations, Springer INdAM Series, Springer, January 2013, pp. 221–235.

Y. Maday, O. Mula, A. T. Patera, and M. Yano, *The generalized empirical interpolation method: Stability theory on Hilbert spaces with an application to the Stokes equation*, Comput. Methods Appl. Mech. and Engrg., 287 (2015), pp. 310–334.

C. W. Rowley, *Model reduction for fluids, using balanced proper orthogonal decomposition*, Internat. J. Bifur. Chaos Appl. Sci. Engrg., 15 (2005), pp. 997–1013.

C. W. Rowley, T. Colonius, and R. M. Murray, *Model reduction for compressible flows using POD and Galerkin projection*, Phys. D, 189 (2004), pp. 115–129.