POD/DEIM Strategies for reduced data assimilation systems

Răzvan Ştefănescu ¹  Adrian Sandu ¹  Ionel M. Navon ²

¹Computational Science Laboratory, Department of Computer Science, Virginia Polytechnic Institute and State University

²Department of Scientific Computing, The Florida State University
Outline

1 Motivation

2 Reduced Order Modelling

3 ROM 4D-Var DA systems - Choice of bases

4 4D-Var SWE DA reduced order systems

5 Numerical results
   - Choice of POD bases
   - Reduced order POD based SWE 4D-Var DA systems

6 Conclusions and future research
Why do we need data assimilation?

By construction, **numerical weather forecasts are imperfect**

- **discrete** representation of the atmosphere in space and time (horizontal and vertical grids, spectral truncation, time step)

- **subgrid-scale processes** (e.g. turbulence, convective activity) need to be parametrized as functions of the resolved-scale variables.

- **errors in the initial conditions.**

- **Physical parametrizations** used in NWP models are constantly being improved: a. more and more **prognostic variables** (cloud variables, precipitation, aerosols), b. more and more **physical processes** accounted for (e.g. detailed microphysics).

- However, they remain approximate representations of the true atmospheric behaviour.
Why do we need data assimilation?

Conventional and satellite data assimilated at ECMWF 1996-2010

Unit is millions of data values assimilated per 24 hour period
Why do we need data assimilation?

- The goal of **data assimilation** is to **periodically constrain the initial conditions** of the forecast using a set of accurate observations that provide our best estimate of the local true atmospheric state.

- This is achieved by combining in an optimal statistical way all the information on the atmosphere, available over a selected time window (usually 6 or 12h)

- **Observations** with their accuracies (error statistics)

- **Short-range model forecast** (background) with associated error statistics

- **Atmospheric equilibria** (e.g. geostrophic balance)

- **Physical laws** (e.g. perfect gas law, condensation)
Why do we need data assimilation?

- **Data assimilation techniques:** OI, Nudging, 3D-Var, 4D-Var, Ensemble DA.

![Diagram showing 4D-Var process]

- All observations $y_0$ between $t_a - 9\text{h}$ and $t_a + 3\text{h}$ are valid at their actual time.
- Model trajectory from corrected initial state.
- Model trajectory from first guess $x_b$.

initial time $t_0$

analysis time $t_a$

4D-Var

$y_0$

Assimilation window
Why do we need data assimilation?

The objective function $J$ to be optimized is defined based on model-data misfit penalty terms as:

$$J(x_0) = \frac{1}{2}(x^b - x_0)^T B_0^{-1}(x^b - x_0)$$

$$+ \frac{1}{2} \sum_{i=1}^{N} (y^i - H(x_i))^T R_i^{-1}(y^i - H(x_i)),$$

subject to

$$x_{i+1} = M_i x_i, \quad i = 0, \ldots, N - 1,$$
Why do we need data assimilation?

- The optimality conditions:

  \[ \lambda_i = M_i^T \lambda_{i+1} + H^T R_i^{-1} (y^i - H(x_i)), \quad i = \overline{N - 1, 1}; \]

  \[ \lambda_N = H^T R_N^{-1} (y^N - H(x_N)) \quad \text{and} \quad \lambda_0 = M_0^T \lambda_0. \] (3)

  \[ \text{Cost function gradient:} \quad \nabla_{x_0} J = - B_0^{-1} (x^b - x_0) - \lambda_0 = 0; \] (4)
Why do we need reduced order data assimilation?
Why do we need reduced order data assimilation?

The operational configuration at ECMWF
- Deterministic forecast model: T1279L91 (∼ 16km)
- Outer loop of 4D-Var T1279L91
- Inner loops T159/T255/T255 (∼125km/80km/80km)

The Mesoscale and Microscale Meteorology (MMM) Division of NCAR currently maintains and supports WRFDA system
- Same resolutions in outer loops and inner loops
- Data Assimilation of lightning using 4D-Var WRFDA system and nonlinear observation operators
- Spatial resolution of 9km and only a 2h assimilation windows
Why do we need reduced order data assimilation?

Figure: 1 h precipitation (mm) ending at 2000 UTC 15 June from the control run, various assimilation procedures, and stage IV precipitation.
Why do we need reduced order data assimilation?

**Figure:** RMSE of precipitation (mm) for both study days compared with stage IV observations. Assimilation was not performed after 2000 UTC for the 1D+4D-VAR simulation, and not after 0000 UTC for the 3D-VAR and 1D+3D-VAR approaches.
Why do we need reduced order data assimilation?

- Replace the current linearized cost function to be minimized in the inner loop
- Low-rank surrogate models that accurately represent sub-grid-scale processes
- Highly non-linear and non-smooth observation operators
- Increased space and time resolutions
- Reduced computational complexity
Reduced Order Modelling

- dependent and independent of input
- For **linear models** we are able to produce input-independent highly accurate reduced models: balanced truncation, moment matching
- For general **nonlinear systems**, the transfer function approach is not yet applicable and input-specified semi-empirical methods are usually employed
- Application of generalized transfer functions and generalized moment matching (Benner and Breiten [2]) for nonlinear model order reduction
Reduced Order Modelling

- dependent and independent of input

- For linear models we are able to produce input-independent highly accurate reduced models: balanced truncation, moment matching

- For general nonlinear systems, the transfer function approach is not yet applicable and input-specified semi-empirical methods are usually employed

- Application of generalized transfer functions and generalized moment matching (Benner and Breiten [2]) for nonlinear model order reduction
Proper Orthogonal Decomposition

- The desired simulation is well approximated in the input collection.
- Data analysis is conducted to extract basis functions, from experimental data or detailed simulations of high-dimensional systems.
- Galerkin projections that yield low dimensional dynamical models.
- Standard POD models: Its nonlinear reduced terms still have to be evaluated on the original state space making the simulation of the reduced-order system too expensive.
- Tensorial POD - Stefanescu et al. [8] - "Comparison of POD reduced order strategies for the nonlinear 2D Shallow Water Equations"
Reduced Order Modelling

- Three POD based reduced order models will be considered for deriving reduced order SWE data assimilation systems: standard Proper Orthogonal Decomposition (SPOD), tensorial POD (TPOD) and standard POD/Discrete Empirical Interpolation Method (POD/DEIM).

- The reduced Jacobians are obtained analytically for all three ROMs and their computational complexity depends only on \( k \) the dimension of POD basis.

- We assume a Petrov-Galerkin projection for constructing the reduced order models. \( U \) denotes the POD basis and the test functions are stored in \( W \). \( W^T U = I_k \), \( I_k \) being the identity matrix of order \( k \). For simplicity we assume a POD expansion of \( x = U \tilde{x} \).

- The methods differ in the way the nonlinear terms are treated - polynomial quadratic nonlinearity \( x^2 \).
Reduced Order Modelling

**Standard POD**

\[
\tilde{N}(\tilde{x}) = \underbrace{W^T U \tilde{x}}_{k \times n} \otimes \underbrace{U \tilde{x}}_{n \times 1}, \quad \tilde{N}(\tilde{x}) \in \mathbb{R}^k
\]

(5)

where \( \otimes \) is the componentwise multiplication Matlab operator and \( n \) is usually the number of spatial mesh points.

**Tensorial POD**

\[
\tilde{N}(\tilde{x}) = \left[ \tilde{N}_i \right]_{i=1,...,k} \in \mathbb{R}^k; \quad \tilde{N}_i = \sum_{j=1}^{k} \sum_{l=1}^{k} T_{i,j,l} \tilde{x}_j \tilde{x}_l.
\]

(6)

\[
\tilde{N}(\tilde{x}) = \underbrace{T}_{k \times k^2} \underbrace{\tilde{X}}_{k^2 \times 1}
\]

\[
T = (T_{i,j,l})_{i,j,l=1,...,k} \in \mathbb{R}^{k \times k \times k}, \quad T_{i,j,l} = \sum_{r=1}^{n} W_{i,r} U_{j,r} U_{l,r}.
\]
Reduced Order Modelling

**Standard POD/DEIM**

\[
\tilde{N}(\tilde{x}) \approx W^T V (P^T V)^{-1} \left( (P^T U\tilde{x}) \odot (P^T U\tilde{x}) \right)
\]

(7)

where \( m \) is the number of interpolation points, \( V \in \mathbb{R}^{n \times m} \) gathers the first \( m \) POD basis modes of the nonlinear term while \( P \in \mathbb{R}^{n \times m} \) is the DEIM interpolation selection matrix (Chaturantabut [4], Chaturantabut and Sorensen [6, 5]).
Figure: 100 DEIM interpolation points. The background consists in isolines of the maximum values of the nonlinear terms over time.
Reduced Order Modelling

\[ F_{11}(u, \phi) = u \odot A_x u + \frac{1}{2} \phi \odot A_x \phi, \quad F_{12}(u, v) = v \odot A_y u \]

\[ F_{21}(u, v) = u \odot A_x v, \quad F_{22}(v, \phi) = v \odot A_y v + \frac{1}{2} \phi \odot A_y \phi, \]

\[ F_{31}(u, \phi) = \frac{1}{2} \phi \odot A_x u + u \odot A_x \phi, \quad F_{32}(v, \phi) = \frac{1}{2} \phi \odot A_y v + v \odot A_y \phi. \]

<table>
<thead>
<tr>
<th></th>
<th>Full ADI SWE</th>
<th>Standard POD</th>
<th>Tensorial POD</th>
<th>POD/DEIM m=180</th>
<th>POD/DEIM m=70</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU time</td>
<td>950.0314s</td>
<td>161.907</td>
<td>2.125</td>
<td>0.642</td>
<td>0.359</td>
</tr>
<tr>
<td>(u)</td>
<td>-</td>
<td>5.358e-5</td>
<td>5.358e-5</td>
<td>5.646e-5</td>
<td>7.453e-5</td>
</tr>
<tr>
<td>(v)</td>
<td>-</td>
<td>2.728e-5</td>
<td>2.728e-5</td>
<td>3.418e-5</td>
<td>4.233e-5</td>
</tr>
<tr>
<td>(\phi)</td>
<td>-</td>
<td>8.505e-5</td>
<td>8.505e-5</td>
<td>8.762e-5</td>
<td>9.212e-5</td>
</tr>
</tbody>
</table>

**Table:** CPU time gains and the root mean square errors for each of the model variables at \( t_f = 3h \) for a 3h time integration window. Number of POD modes was \( k = 50 \) and two tests with different number of DEIM points \( m = 180, 70 \) were simulated.103, 776 spatial points.
Figure: Cpu time vs. the number of spatial discretization points for $t_f = 3h$; number of POD modes = 50; two different numbers of DEIM points 70 and 180 have been employed.
ROM 4D-Var DA systems - Choice of bases

- The “reduced adjoint” (RA) approach projects the first order optimality equations of the full system onto the POD reduced spaces.

- Accurate low-order surrogate models; It's not clear what information should be included in the reduced basis used for full space gradient equation projection.

- The “adjoint of reduced” (AR) model approach formulates the first order optimality conditions from the forward reduced order model.

- Consistent KKT discrete optimality conditions; Reduced adjoint model approximates poorly its full counterpart and POD bases rely only on forward dynamics information.
ROM 4D-Var DA systems - Choice of bases

- To guide the POD bases snapshots selection process for Petrov-Galerkin based reduced data assimilation systems governed by non-linear models

- Consistent reduced Karush Kuhn Tucker (KKT) optimality conditions and accurate reduced POD adjoint model solutions with respect to the full adjoint model outputs

- Every type of reduced optimization involving adjoint models and projection based reduced order methods including reduced basis approach will benefit.
We derive the optimality conditions as in the AR approach.

Forward POD manifold $U_f$ is computed using snapshots of the full forward model solution only $x \approx U_f \tilde{x}$

Petrov-Galerkin (PG) projection; the test functions POD basis $W_f$ is different than the trial functions POD manifold $U_f$

\[
J^{POD}(\tilde{x}_0) = \frac{1}{2} (x^b - U_f \tilde{x}_0)^T B_0^{-1} (x^b - U_f \tilde{x}_0)^T
\]

\[
+ \frac{1}{2} \sum_{i=1}^{N} (y^i - H(U_f \tilde{x}_i))^T R_i^{-1} (y^i - H(U_f \tilde{x}_i))^T,
\]

subject to $\tilde{x}_{i+1} = \tilde{M}_i \tilde{x}_i$, $\tilde{M}_i = W_f^T M_i U_f$, $i = 0, \ldots, N - 1$. (8)
ROM 4D-Var DA systems - Choice of bases

- **Reduced adjoint model**

\[
\tilde{\lambda}_i = U_f^T M_i^T W_f \tilde{\lambda}_{i+1} + U_f^T H^T R_i^{-1} (y^i - H(U_f \tilde{x}_i)), \quad i = N - 1, 1;
\]

\[
\tilde{\lambda}_N = U_f^T H^T R_N^{-1} (y^N - H(U_f \tilde{x}_N)) \quad \text{and} \quad \tilde{\lambda}_0 = U_f^T M_0^T W_f \tilde{\lambda}_1
\]  

(10)

- **Reduced Cost Function gradient**

\[
\nabla \tilde{x}_0 J^{POD} = -U_f^T B_0^{-1} (x^b - U_f \tilde{x}_0) - \tilde{\lambda}_0 = 0;
\]  

(11)
ROM 4D-Var DA systems - Choice of bases

- RA approach: the full forward and adjoint models are projected onto separate reduced manifolds

- $U_f$ and $U_a$ are the trial POD reduced subspaces and $W_f$ and $W_a$ are the test functions POD manifolds, $x_i \approx U_f \tilde{x}_i$, $\lambda_i \approx U_a \tilde{\lambda}_i$, $i = 0, \ldots, N$.

- Reduced forward model:

  $$\tilde{x}_{i+1} = \tilde{M}_i \tilde{x}_i, \quad \tilde{M}_i = W_f^T M_i U_f, \quad i = 0, \ldots, N - 1.$$  
  \hspace{1cm} (12)

- Reduced adjoint model:

  $$\tilde{\lambda}_i = W_a^T M_i^T U_a \tilde{\lambda}_{i+1} + W_a^T H^T R_i^{-1} (y^i - H(U_f \tilde{x}_i)), \quad i = N - 1, 1$$
  $$\tilde{\lambda}_N = W_a^T H^T R_N^{-1} (y^N - H(U_f \tilde{x}_N)) \text{ and } \tilde{\lambda}_0 = W_a^T M_0^T U_a \tilde{\lambda}_1,$$

  \hspace{1cm} (13)
ROM 4D-Var DA systems - Choice of bases

- **AR adjoint model:**

\[
\tilde{\lambda}_i = U_f^T M_i^T W_f \tilde{\lambda}_{i+1} + U_f^T H^T R_i^{-1} (y^i - H(U_f \tilde{x}_i)), \quad i = N - 1, 1; \\
\tilde{\lambda}_N = U_f^T H^T R_N^{-1} (y^N - H(U_f \tilde{x}_N)) \quad \text{and} \quad \tilde{\lambda}_0 = U_f^T M_0^T W_f \tilde{\lambda}_1
\]

- **RA adjoint model:**

\[
\tilde{\lambda}_i = W_a^T M_i^T U_a \tilde{\lambda}_{i+1} + W_a^T H^T R_i^{-1} (y^i - H(U_f \tilde{x}_i)), \quad i = N - 1, 1 \\
\tilde{\lambda}_N = W_a^T H^T R_N^{-1} (y^N - H(U_f \tilde{x}_N)) \quad \text{and} \quad \tilde{\lambda}_0 = W_a^T M_0^T U_a \tilde{\lambda}_1
\]

- For Petrov Galerkin and Galerking projections:

\[
W_f = U_a \quad \text{and} \quad W_a = U_f, \quad \text{and} \quad U_f = U_a.
\]
4D-Var SWE DA reduced order systems

Algorithm 1 Standard and Tensorial POD SWE DA systems

Off-line stage

1: Generate background state $u$, $v$ and $\phi$.
2: Solve full forward ADI SWE model to generate state variables snapshots.
3: Solve full adjoint ADI SWE model to generate adjoint variables snapshots.
4: For each state variable compute a POD basis using snapshots describing dynamics of the forward and its corresponding adjoint trajectories.
5: Compute tensors as $T$ required for reduced Jacobian calculations. Calculate other POD coefficients corresponding to linear terms.
4D-Var SWE DA reduced order systems

Algorithm 1 Standard and Tensorial POD SWE DA systems

On-line stage - Minimize reduced cost functional $J^{POD}$ (8)

1. Solve forward reduced order model (9)
2. Solve adjoint reduced order model (10)
3. Compute reduced gradient (11)

Decisional stage

4. Project the suboptimal reduced initial condition generated by the on-line stage and perform steps 1 and 2 of off-line stage. Using full forward information evaluate $J$ in (1). If $\|J\| > \varepsilon_3$ then continue the off-line stage from step 3, otherwise STOP.
Algorithm 2 POD/DEIM SWE DA system

Off-line stage

1: Generate background state $\mathbf{u}$, $\mathbf{v}$ and $\phi$.
2: Solve full forward ADI SWE model to generate nonlinear terms and state variables snapshots.
3: Solve full adjoint ADI SWE model to generate adjoint variables snapshots.
4: For each state variable compute a POD basis using snapshots describing dynamics of the forward and its corresponding adjoint trajectories. For each nonlinear term compute a POD basis using snapshots from the forward model.
5: Compute discrete empirical interpolation points for each nonlinear term.
6: Calculate other linear POD coefficients and POD/DEIM coefficients as $E_{11}$.
7: Compute tensors such as $T$ using algorithm described in Stefanescu et al. [8, p.7] required for reduced Jacobian calculations.
4D-Var SWE DA reduced order systems

- The on-line stage - minimization of the cost function $J^{POD}$ performed on a reduced POD manifold

- The stoping criteria are

$$\|\nabla J^{POD}\| \leq \varepsilon_1, \quad \|J^{POD}_{(i+1)} - J^{POD}_{(i)}\| \leq \varepsilon_2, \quad \text{MXFUN} \leq \text{iter}_{\text{Max}}$$  \hspace{1cm} (17)

- The off-line stage - outer iteration - general stopping criterion

$$\|J\| \leq \varepsilon_3$$
Numerical Results

- ADI SWE model
- 10% uniform perturbations on the initial conditions of Grammeltvedt [7] and generate twin-experiment observations at every grid space point location and every time step
- Background state is computed using a 5% perturbations of the initial conditions
- The length of the assimilation window: $3h$.
- BFGS optimization method (CONMIN)
- We use $\epsilon_1 = 10^{-14}$ and $\epsilon_2 = 10^{-5}$.
Numerical Results - Choice of POD basis

- The "adjoint of reduced" (AR) approach is compared with "adjoint of reduced + reduced of adjoint" (AR+RA) method using tensorial POD 4D-Var DA system.

- AR - no need for implementing the full adjoint SWE model since the POD basis relies only on forward trajectories snapshots.

- We select $31 \times 23$ mesh points 91 time steps and use 50 POD basis functions. MXFUN is set to 25 and $\varepsilon_3 = 10^{-16}$.

<table>
<thead>
<tr>
<th></th>
<th>$|u_{TPOD}^{AR} - u|$</th>
<th>$|u_{TPOD}^{AR+RA} - u|$</th>
<th>$|v_{TPOD}^{AR} - v|$</th>
<th>$|v_{TPOD}^{AR+RA} - v|$</th>
<th>$|\phi_{TPOD}^{AR} - \phi|$</th>
<th>$|\phi_{TPOD}^{AR+RA} - \phi|$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$|u_{TPOD}^{AR} - u|$</td>
<td>2.39e-15</td>
<td>1.39e-6</td>
<td>8.16e-16</td>
<td>7.54e-7</td>
<td>2.75e-13</td>
<td>1.59e-6</td>
</tr>
<tr>
<td>$|v_{TPOD}^{AR} - v|$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$|\phi_{TPOD}^{AR} - \phi|$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table: Average absolute errors of forward (up) and adjoint (down) tensorial
Numerical Results - Choice of POD basis

(a) Forward model snapshots only

(b) Forward and adjoint models snapshots

Figure: The decay around the singular values of the snapshots solutions for $u$, $v$, $\phi$ for $\Delta t = 960s$ and integration time window of $3h$. 
Figure: Tensorial POD/4DVAR ADI 2D Shallow water equations – Evolution of cost function and gradient norm as a function of the number of minimization iterations. The information from the adjoint equations has to be incorporated into POD basis.
Numerical Results - Choice of POD basis

Figure: Tensorial POD/4DVAR ADI 2D Shallow water equations. The information from the adjoint equations has to be incorporated into POD basis.
POD/DEIM SWE 4D-Var DA system

- Mesh of $31 \times 23$ and $17 \times 13$ points, a POD basis dimension of $k = 50$, and various number of DEIM interpolation points are used.
- MXFUN is set to 100 and $\varepsilon_3 = 10^{-16}$

(a) Number of mesh points $31 \times 23$  
(b) Number of mesh points $17 \times 13$

Figure: Standard POD/DEIM ADI SWE 4D-Var system – Evolution of cost function and gradient norm as a function of the number of minimization iterations for different number of mesh points and various number of DEIM points.
Reduced order POD based SWE 4D-Var DA systems

(a) Evolution of cost function

(b) Evolution of gradient

Figure: Standard POD/DEIM ADI SWE 4D-Var system – Evolution of cost function and gradient norm as a function of the number of minimization iterations for different number of mesh points and various number of DEIM points. $n = 31 \times 23$
Reduced order POD based SWE 4D-Var DA systems

(a) Evolution of cost function

(b) Evolution of gradient

Figure: Standard POD/DEIM ADI SWE 4D-Var system – Evolution of cost function and gradient norm as a function of the number of minimization iterations for different number of mesh points and various number of DEIM points. $n = 17 \times 13$
Hybrid POD/DEIM SWE 4D-Var DA system

Figure: Tangent linear and adjoint test for standard POD/DEIM SWE 4D-Var system. Optimization performances of Standard POD/DEIM and Hybrid POD/DEIM 4DVAR ADI 2D shallow water equations for 50 DEIM points and \( n = 17 \times 13 \) space points.
Hybrid POD/DEIM SWE 4D-Var DA system

(a) Evolution of cost function

(b) Evolution of gradient

Figure: Optimization performances of Standard POD/DEIM and Hybrid POD/DEIM 4DVAR ADI 2D shallow water equations for 50 DEIM points and \( n = 17 \times 13 \)
Hybrid POD/DEIM SWE 4D-Var DA system

Figure: Performances of hybrid POD/DEIM SWE DA system with various values of POD basis dimensions (a) and different number of interpolation points (b). The spatial configuration uses $n = 61 \times 45$ and maximum number of function evaluation per inner iteration is $\text{MXFUN} = 30$
Hybrid POD/DEIM SWE 4D-Var DA system

<table>
<thead>
<tr>
<th></th>
<th>$k = 20$</th>
<th>$k = 30$</th>
<th>$k = 50$</th>
<th>$k = 70$</th>
<th>$k = 90$</th>
<th>Full</th>
</tr>
</thead>
<tbody>
<tr>
<td>$E_u^o$</td>
<td>1.03e-4</td>
<td>1.57e-6</td>
<td>4.1e-11</td>
<td>3.75e-11</td>
<td>2.1e-11</td>
<td>2.77e-12</td>
</tr>
<tr>
<td>$E_v^o$</td>
<td>4.04e-4</td>
<td>6.29e-6</td>
<td>1.72e-10</td>
<td>1.16e-10</td>
<td>8.55e-11</td>
<td>1.27e-11</td>
</tr>
<tr>
<td>$E_{\phi}^o$</td>
<td>2.36e-6</td>
<td>3.94e-8</td>
<td>1.09e-12</td>
<td>6.78e-13</td>
<td>2.3e-13</td>
<td>5.79e-14</td>
</tr>
</tbody>
</table>

Table: Relative errors of suboptimal hybrid POD/DEIM SWE 4D-Var solutions, optimal solution computed using high-fidelity ADI SWE 4D-Var system and observations. The number of DEIM interpolation points is hold constant $m = 50$ while $k$ is varied.

<table>
<thead>
<tr>
<th></th>
<th>$m = 10$</th>
<th>$m = 20$</th>
<th>$m = 30$</th>
<th>$m = 50$</th>
<th>$m = 100$</th>
<th>Full</th>
</tr>
</thead>
<tbody>
<tr>
<td>$E_u^o$</td>
<td>4.28e-6</td>
<td>4.91e-8</td>
<td>4.15e-10</td>
<td>4.1e-11</td>
<td>3.08e-11</td>
<td>2.77e-12</td>
</tr>
<tr>
<td>$E_v^o$</td>
<td>6.88e-6</td>
<td>1.889e-7</td>
<td>1.54e-9</td>
<td>1.72e-10</td>
<td>1.25e-10</td>
<td>1.27e-11</td>
</tr>
<tr>
<td>$E_{\phi}^o$</td>
<td>7.15e-8</td>
<td>1.04e-9</td>
<td>7.79e-12</td>
<td>1.09e-13</td>
<td>7.39e-13</td>
<td>5.79e-14</td>
</tr>
</tbody>
</table>

Table: Relative errors of suboptimal hybrid POD/DEIM SWE 4D-Var solutions, optimal solution computed using high-fidelity ADI SWE 4D-Var system and observations. Different DEIM interpolation points are tested and $k$ is hold constant 50.
POD based SWE 4D-Var DA systems

- $n = 61 \times 45$ space points, number of POD basis modes $k = 50$, 
  $\text{MXFUN} = 10$ and $\varepsilon_3 = 10^{-7}$.

**Figure:** Number of minimization iterations and CPU time comparisons for the reduced Order SWE DA systems vs. full SWE DA system. The spatial configuration uses $n = 61 \times 45$ and maximum number of function evaluation per inner iteration is $\text{MXFUN} = 10$.
POD based SWE 4D-Var DA systems

- \( n = 151 \times 111 \) space points, number of POD basis modes \( k = 50 \), \( \text{MXFUN} = 15 \) and \( \varepsilon_3 = 10^{-1} \).

(a) Iteration performance

(b) Time performance

Figure: Number of iterations and CPU time comparisons for the reduced Order SWE DA systems vs. full SWE DA system.
**POD based SWE 4D-Var DA systems**

<table>
<thead>
<tr>
<th>Space points</th>
<th>31 × 23</th>
<th>61 × 45</th>
<th>101 × 71</th>
<th>121 × 89</th>
<th>151 × 111</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\varepsilon_3$</td>
<td>$</td>
<td></td>
<td>J</td>
<td></td>
<td>&lt; 1. e - 09$</td>
</tr>
<tr>
<td>hybrid DEIM 50</td>
<td>48.77</td>
<td>63.34</td>
<td>199.46</td>
<td>358.17</td>
<td>246.39</td>
</tr>
<tr>
<td>hybrid DEIM 120</td>
<td>44.37</td>
<td>64.77</td>
<td>210.66</td>
<td>431.46</td>
<td>286.00</td>
</tr>
<tr>
<td>standard POD</td>
<td>63.14</td>
<td>131.43</td>
<td>533.05</td>
<td>760.46</td>
<td>560.61</td>
</tr>
<tr>
<td>tensorial POD</td>
<td>54.54</td>
<td>67.13</td>
<td>216.29</td>
<td>391.07</td>
<td>303.95</td>
</tr>
<tr>
<td>FULL</td>
<td>10.64</td>
<td>117.02</td>
<td>792.93</td>
<td>1562.34</td>
<td>3038.24</td>
</tr>
</tbody>
</table>

**Table**: CPU time for reduced optimization and full 4D-Var. Number of POD modes is selected 30 and MXFUN = 15.

<table>
<thead>
<tr>
<th>Space points</th>
<th>31 × 23</th>
<th>61 × 45</th>
<th>101 × 71</th>
<th>121 × 89</th>
<th>151 × 111</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\varepsilon_3$</td>
<td>$</td>
<td></td>
<td>J</td>
<td></td>
<td>&lt; 1. e - 14$</td>
</tr>
<tr>
<td>hybrid DEIM 50</td>
<td>214.78</td>
<td>288.63</td>
<td>593.36</td>
<td>499.67</td>
<td>594.04</td>
</tr>
<tr>
<td>hybrid DEIM 50</td>
<td>211.51</td>
<td>246.65</td>
<td>529.93</td>
<td>512.72</td>
<td>603.21</td>
</tr>
<tr>
<td>standard POD</td>
<td>190.57</td>
<td>402.21</td>
<td>1243.23</td>
<td>1315.57</td>
<td>1375.4</td>
</tr>
<tr>
<td>tensorial POD</td>
<td>269.08</td>
<td>311.11</td>
<td>585.04</td>
<td>662.95</td>
<td>685.57</td>
</tr>
<tr>
<td>FULL</td>
<td>14.10</td>
<td>155.67</td>
<td>1057.71</td>
<td>2261.67</td>
<td>5268.7</td>
</tr>
</tbody>
</table>

**Table**: CPU time for reduced optimization and full 4D-Var. Number of POD modes is selected 50 and MXFUN = 15.
### Table: CPU time for full 4D-Var for a number of mesh points of $151 \times 111$.  

| Process | $||J|| < 1.e - 01$ | Time | # | Total |
|---------|-------------------|------|---|-------|
| Solve full forward model | | $\approx 80s$ | 26x | $\approx 2080s$ |
| Solve full adjoint model | | $\approx 76.45s$ | 26x | $\approx 1987.7s$ |
| Other (Line Search, Hessian approx) | | $\approx 46.165$ | 26x | $\approx 1200.3s$ |
| **Total full 4D-Var** | | | | $\approx 5268s$ |

### Table: Cpu time Hybrid POD/DEIM 4D-Var for a number of mesh points of $151 \times 111$, POD basis dimension $k = 50$ and 30 DEIM interpolation points.  

<table>
<thead>
<tr>
<th>Process</th>
<th>$k= 50$</th>
<th>Time</th>
<th>#</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Off-line stage</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Solve full forward model + nonlinear snap.</td>
<td></td>
<td>$\approx 80.88s$</td>
<td>2x</td>
<td>$\approx 161.76s$</td>
</tr>
<tr>
<td>Solve full adjoint model + nonlinear snap.</td>
<td></td>
<td>$\approx 76.45s$</td>
<td>2x</td>
<td>$\approx 152.9s$</td>
</tr>
<tr>
<td>SVD for state variables</td>
<td></td>
<td>$\approx 53.8$</td>
<td>2x</td>
<td>$\approx 107.6s$</td>
</tr>
<tr>
<td>SVD for nonlinear terms</td>
<td></td>
<td>$\approx 11.57$</td>
<td>2x</td>
<td>$\approx 23.14s$</td>
</tr>
<tr>
<td>DEIM interpolation points</td>
<td></td>
<td>$\approx 0.115$</td>
<td>2x</td>
<td>$\approx 0.23s$</td>
</tr>
<tr>
<td>POD/DEIM model coefficients</td>
<td></td>
<td>$\approx 1.06$</td>
<td>2x</td>
<td>$\approx 2.12s$</td>
</tr>
<tr>
<td>tensorial POD model coefficients</td>
<td></td>
<td>$\approx 8.8$</td>
<td>2x</td>
<td>$\approx 17.6s$</td>
</tr>
<tr>
<td><strong>On-line stage</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Solve ROM forward</td>
<td></td>
<td>$\approx 2s$</td>
<td>33x</td>
<td>$\approx 66s$</td>
</tr>
<tr>
<td>Solve ROM adjoint</td>
<td></td>
<td>$\approx 1.9s$</td>
<td>33x</td>
<td>$\approx 62.69s$</td>
</tr>
<tr>
<td><strong>Total Hybrid POD/DEIM 4D-Var</strong></td>
<td></td>
<td></td>
<td></td>
<td>$\approx 594.04s$</td>
</tr>
<tr>
<td>Process</td>
<td>k=50</td>
<td>Time</td>
<td>#</td>
<td>Total</td>
</tr>
<tr>
<td>----------------------------------------------</td>
<td>------</td>
<td>--------</td>
<td>----</td>
<td>---------</td>
</tr>
<tr>
<td><strong>Off-line stage</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Solve full forward model + nonlinear snap.</td>
<td></td>
<td>$\approx 80\text{s}$</td>
<td>2x</td>
<td>$\approx 160\text{s}$</td>
</tr>
<tr>
<td>Solve full adjoint model + nonlinear snap.</td>
<td></td>
<td>$\approx 76.45\text{s}$</td>
<td>2x</td>
<td>$\approx 152.9\text{s}$</td>
</tr>
<tr>
<td>SVD for state variables</td>
<td></td>
<td>$\approx 53.8\text{s}$</td>
<td>2x</td>
<td>$\approx 107.6\text{s}$</td>
</tr>
<tr>
<td>tensorial POD model coefficients</td>
<td></td>
<td>$\approx 23.735\text{s}$</td>
<td>2x</td>
<td>$\approx 47.47\text{s}$</td>
</tr>
<tr>
<td><strong>On-line stage</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Solve ROM forward</td>
<td></td>
<td>$\approx 4.9\text{s}$</td>
<td>32x</td>
<td>$\approx 156.8\text{s}$</td>
</tr>
<tr>
<td>Solve ROM adjoint</td>
<td></td>
<td>$\approx 1.9\text{s}$</td>
<td>32x</td>
<td>$\approx 60.8\text{s}$</td>
</tr>
<tr>
<td>Total Tensorial POD 4D-Var</td>
<td></td>
<td></td>
<td></td>
<td>$\approx 685.57\text{s}$</td>
</tr>
</tbody>
</table>

**Table:** The calculation times for solving the optimization problem using Tensorial POD 4D-Var for a number of mesh points of $151 \times 111$, and POD basis dimension $k = 50$.

<table>
<thead>
<tr>
<th>Process</th>
<th>k=50</th>
<th>Time</th>
<th>#</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>On-line stage</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Solve ROM forward</td>
<td></td>
<td>$\approx 26.523\text{s}$</td>
<td>32x</td>
<td>$\approx 846.72\text{s}$</td>
</tr>
<tr>
<td>Solve ROM adjoint</td>
<td></td>
<td>$\approx 1.9\text{s}$</td>
<td>32x</td>
<td>$\approx 60.8\text{s}$</td>
</tr>
<tr>
<td>Total Standard 4D-Var</td>
<td></td>
<td></td>
<td></td>
<td>$\approx 1375.4\text{s}$</td>
</tr>
</tbody>
</table>

**Table:** The calculation times for on-line stage of Standard POD 4D-Var for a number of mesh points of $151 \times 111$ and POD basis dimension $k = 50$. The off-line stage is identical with the one in Tensorial POD 4D-Var system.
Conclusions and future research

- New efficient POD bases selection strategies for POD based reduced 4DVar data assimilation systems governed by nonlinear state models using both Petrov-Galerkin and Galerkin projections.

- Consistent reduced Karush Kuhn Tucker (KKT) optimality conditions + accurate reduced POD adjoint model solutions with respect to the full adjoint model outputs.

- Petrov-Galerkin projection - test functions POD bases of the forward and adjoint models have to match the trial functions POD bases of the adjoint and forward models

- Galerkin projection - one single POD basis is required and the correlation matrix must contain snapshots from both forward and adjoint full models
Conclusions and future research

- Every type of reduced optimization involving adjoint models and projected based reduced order methods including reduced basis approach benefit from the new strategies.

- Standard POD, Tensorial POD and Standard POD/DEIM SWE 4D-Var systems

- The POD/DEIM approximations of four nonlinear terms involving height field out of ten partially lost their accuracy during the optimization where input data are different than the ones used to generate the interpolation points - Hybrid tensorial POD/DEIM 4D-Var SWE.

- For meshes of $151 \times 111$ points or higher the hybrid POD/DEIM reduced data assimilation system is approximately 10 times faster then the full space data assimilation system
Conclusions and future research

- This rate increases directly proportional with the mesh size.

- Hybrid POD/DEIM SWE 4D-Var is at least two times faster than standard POD SWE 4D-Var for numbers of space points larger or equal to $101 \times 71$.

- Stabilization strategies proposed by Amsallem and Farhat [1], Bui-Thanh et al. [3] must be pursued in order to obtain feasible Petrov-Galerkin reduced order data assimilation systems.

- A generalization of DEIM to approximate operators has not been yet developed.

- A-priori error estimates extensions for nonlinear-quadratic reduced order optimal problems.

- A-posteriori error estimation apparatus - hybrid POD/DEIM SWE 4D-Var system - POD basis construction and to efficiently select the number of DEIM interpolation points.
Manuscripts related to the present research effort


Figure: Deep down, the founder of modern weather forecasting, Vilhelm Bjerknes, would have preferred to work on theoretical physics. Until he got funding for weather research, that is.
POD/DEIM Strategies for reduced data assimilation systems

Thank You


