POD/DEIM 4DVAR Data Assimilation of the Shallow Water Equation Model

R. Ştefănescu and Ionel M. Navon

Department of Scientific Computing
Florida State University
Tallahassee, Florida

May 23, 2013

Contents

Contents

Contents

POD/DEIM POD/EIM justification and methodology

- Model order reduction : Reduce the computational complexity/time of large scale dynamical systems by approximations of much lower dimension with nearly the same input/output response characteristics.

- Goal : Construct reduced-order model for different types of discretization method (finite difference (FD), finite element (FEM), finite volume (FV)) of unsteady and/or parametrized nonlinear PDEs. E.g., PDE:

$$\frac{\partial y}{\partial t}(x,t) = \mathrm{L}(y(x,t)) + \mathrm{F}(y(x,t)),\ \ t \in [0, T]$$

where L is a linear function and F a nonlinear one.

POD/DEIM methodology applied to FD SCHEMES

- The corresponding FD scheme is a n dimensional ordinary differential system

$$\frac{d}{dt}\mathbf{y}(t) = A\mathbf{y}(t) + \mathbf{F}(\mathbf{y}(t)),\ A \in \mathbb{R}^{n\times n},$$

where $\mathbf{y}(t) = [y_1(t), y_2(t), .., y_n(t)] \in \mathbb{R}^n$ and $y_i(t) \in \mathbb{R}$ are the spatial components $y(x_i, t),\ i = 1, .., n$. $\mathbf{F}$ is a nonlinear function evaluated at $\mathbf{y}(t)$ componentwise, i.e. $\mathbf{F} = [\mathrm{F}(y_1(t)), .., \mathrm{F}(y_n(t))]^T$, $\mathrm{F} : I \subset \mathbb{R} \rightarrow \mathbb{R}$.

POD/DEIM methodology applied to FD SCHEMES

- A common model order reduction method involves the Galerkin projection with basis $V_k \in \mathbb{R}^{n\times k}$ obtained from Proper Orthogonal Decomposition (POD), for $k \ll n$, i.e. $\mathbf{y} \approx V_k\tilde{\mathbf{y}}(\mathbf{t})$, $\tilde{\mathbf{y}}(\mathbf{t}) \in \mathbb{R}^k$. Applying an inner product to the ODE discrete system we get

$$\frac{d}{dt}\tilde{\mathbf{y}}(\mathbf{t}) = \underbrace{V_k^T A V_k}_{k\times k}\tilde{\mathbf{y}}(\mathbf{t}) + \underbrace{V_k^T \mathbf{F}(V_k\tilde{\mathbf{y}}(\mathbf{t}))}_{\tilde{N}(\tilde{\mathbf{y}})} \quad (1)$$

POD/DEIM methodology applied to FD SCHEMES

- The efficiency of POD - Galerkin technique is limited to the linear or bilinear terms. The projected nonlinear term still depends on the dimension of the original system

$$\tilde{N}(\tilde{\mathbf{y}}) = \underbrace{V_k^T}_{k\times n} \underbrace{\mathbf{F}(V_k\tilde{\mathbf{y}}(\mathbf{t}))}_{n\times 1}.$$

- To mitigate this inefficiency we introduce "Discrete Empirical Interpolation Method (DEIM) " for nonlinear approximation.
- DEIM is a discrete variation of the Empirical Interpolation method proposed by Barrault et al. (2004). The application was suggested by Chaturantabut and Sorensen (2008, 2010, 2012).

POD/DEIM methodology applied to FD SCHEMES

- M. Barrault, Y. Maday, N. Nguyen, A.T. Patera, An 'empirical interpolation' method: application to efficient reduced-basis discretization of partial differential equations, C. R. Acad. Sci. Paris, Ser. I 339 (2004) 667–672.
- It is an efficient reduced-basis discretization procedure for partial differential equations with non-affine parameter dependence. The method replaces non affine coefficient functions with a collateral reduced-basis expansion which then permits an (effectively affine) offline-online computational decomposition.

POD/DEIM methodology applied to FD SCHEMES

- The efficiency of POD - Galerkin technique is limited to the linear or bilinear terms. The projected nonlinear term still depends on the dimension of the original system

$$\tilde{N}(\tilde{\mathbf{y}}) = \underbrace{V_k^T}_{k \times n} \underbrace{\mathbf{F}(V_k \tilde{\mathbf{y}}(\mathbf{t}))}_{n \times 1}.$$

- For $m \ll n$

$$\tilde{N}(\tilde{\mathbf{y}}) \approx \underbrace{V_k^T U (P^T U)^{-1}}_{precomputed \;\; k \times m} \underbrace{\mathbf{F}(P^T V_k \tilde{\mathbf{y}}(\mathbf{t}))}_{m \times 1}$$

POD/DEIM nonlinear model reduction for SWE

- We applied DEIM to a POD alternating direction implicit (ADI) FD scheme of the SWE on a rectangular domain.
- We considered the alternating direction fully implicit finite-difference scheme (Gustafsson 1971, Fairweather and Navon 1980, Navon and De Villiers 1986, Kreiss and Widlund 1966) on a rectangular domain since the scheme remains stable at large Courant numbers (CFL).

SWE model

$$\frac{\partial w}{\partial t} = A(w)\frac{\partial w}{\partial x} + B(w)\frac{\partial w}{\partial y} + C(y)w, \quad (2)$$

$$0 \leq x \leq L,\ 0 \leq y \leq D,\ t \in [0, t_f],$$

where $w = (u, v, \phi)^T$, u, v are the velocity components in the x and y directions, respectively, h is the depth of the fluid, g is the acceleration due to gravity and $\phi = 2\sqrt{gh}$.
The matrices $A,\ B$ and C are expressed

$$A = -\begin{pmatrix} u & 0 & \phi/2 \\ 0 & u & 0 \\ \phi/2 & 0 & u \end{pmatrix},\ B = -\begin{pmatrix} v & 0 & 0 \\ 0 & v & \phi/2 \\ 0 & \phi/2 & v \end{pmatrix}$$

$$C = \begin{pmatrix} 0 & f & 0 \\ -f & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix},$$

$$f = \hat{f} + \beta(y - D/2) \text{ (Coriolis force)},\ \beta = \frac{\partial f}{\partial y}, \text{with } \hat{f} \text{ and } \beta \text{ constants.}$$

SWE model

- We assume periodic solutions in the x-direction

$$w(x, y, t) = w(x + L, y, t),$$

while in the $y-$direction we have

$$v(x, 0, t) = v(x, D, t) = 0.$$

- The initial conditions are derived from the initial height-field condition No. 1 of Grammelvedt (1969), i.e.

$$h(x, y) = H_0 + H_1 + \tanh\left(9\frac{D/2 - y}{2D}\right) + H_2 \operatorname{sech}^2\left(9\frac{D/2 - y}{2D}\right)\sin\left(\frac{2\pi x}{L}\right)$$

The nonlinear Gustafsson ADI finite difference implicit scheme

- First we introduce a network of $N_x \cdot N_y$ equidistant points on $[0, L] \times [0, D]$, with $dx = L/(N_x - 1),\ dy = D/(N_y - 1)$. We also discretize the time interval $[0, t_f]$ using NT equally distributed points and $dt = t_f/(NT - 1)$.
- Next we define vectors of unknown variables of dimension $n_{xy} = N_x \cdot N_y$ containing approximate solutions such as

$$\mathbf{u}(t) \approx u(x_i, y_j, t), \mathbf{v}(t) \approx v(x_i, y_j, t), \boldsymbol{\phi} \approx \phi(x_i, y_j, t) \in \mathbb{R}^{n_{xy}}$$

- The idea behind the ADI method is to split the finite difference equations into two, one with the x-derivative taken implicitly and the next with the y-derivative taken implicitly,

The POD version of SWE model

- The POD reduced-order system is constructed by applying the Galerkin projection method to ADI FD discrete model by first replacing $\mathbf{u}, \mathbf{v}, \phi$ with their POD based approximation $U\tilde{u}$, $V\tilde{v}$, $\Phi\tilde{\phi}$, respectively, and then premultiplying the corresponding equations by U^T, V^T and Φ^T, the POD bases.

- The resulting POD reduced system for the first step $(t_{n+\frac{1}{2}})$ of the ADI FD scheme is

$$
\begin{aligned}
&\tilde{u}(t_{n+\frac{1}{2}}) + \frac{\Delta t}{2} U^T \tilde{F}_{11}\Big(\tilde{u}(t_{n+\frac{1}{2}}), \tilde{\phi}(t_{n+\frac{1}{2}})\Big) = \tilde{u}(t_n) - \frac{\Delta t}{2} U^T \tilde{F}_{12}\Big(\tilde{u}(t_n), \tilde{v}(t_n)\Big) \\
&\qquad + \frac{\Delta t}{2} U^T \Big(\underbrace{[f, f, .., f]}_{N_x}{}^T * V\tilde{v}(t_n)\Big), \\
&\tilde{v}(t_{n+\frac{1}{2}}) + \frac{\Delta t}{2} V^T \tilde{F}_{21}\Big(\tilde{u}(t_{n+\frac{1}{2}}), \tilde{v}(t_{n+\frac{1}{2}})\Big) + \frac{\Delta t}{2} V^T \Big(\underbrace{[f, f, .., f]}_{N_x}{}^T * U\tilde{u}(t_{n+\frac{1}{2}})\Big) \\
&\qquad = \tilde{v}(t_n) - \frac{\Delta t}{2} V^T \tilde{F}_{22}\Big(\tilde{v}(t_n), \tilde{\phi}(t_n)\Big), \\
&\tilde{\phi}(t_{n+\frac{1}{2}}) + \frac{\Delta t}{2} \Phi^T \tilde{F}_{31}\Big(\tilde{u}(t_{n+\frac{1}{2}}), \tilde{\phi}(t_{n+\frac{1}{2}})\Big) = \tilde{\phi}(t_n) - \frac{\Delta t}{2} \Phi^T \tilde{F}_{32}\Big(\tilde{v}(t_n), \tilde{\phi}(t_n)\Big),
\end{aligned}
\tag{3}
$$

where $\tilde{F}_{11}, \tilde{F}_{12}, \tilde{F}_{21}, \tilde{F}_{22}, \tilde{F}_{31}, \tilde{F}_{32} : \mathbb{R}^k \times \mathbb{R}^k \to \mathbb{R}^k$ are defined by

The POD version of SWE model

$$\begin{aligned}
\tilde{F}_{11}(\tilde{u},\tilde{\phi}) &= (U\tilde{u}) * (\underbrace{A_x U}\,\tilde{u}) + \frac{1}{2}(\Phi\tilde{\phi}) * (\underbrace{A_x \Phi}\,\tilde{\phi}),\\
\tilde{F}_{12}(\tilde{u},\tilde{v}) &= (V\tilde{v}) * (\underbrace{A_y U}\,\tilde{u}), \tilde{F}_{21}(\tilde{u},\tilde{v}) = (U\tilde{u}) * (\underbrace{A_x V}\,\tilde{v}),\\
\tilde{F}_{22}(\tilde{v},\tilde{\phi}) &= (V\tilde{v}) * (\underbrace{A_y V}\,\tilde{v}) + \frac{1}{2}(\Phi\tilde{\phi}) * (\underbrace{A_y \Phi}\,\tilde{\phi}),\\
\tilde{F}_{31}(\tilde{u},\tilde{\phi}) &= \frac{1}{2}(\Phi\tilde{\phi}) * (\underbrace{A_x U}\,\tilde{u}) + (U\tilde{u}) * (\underbrace{A_x \Phi}\,\tilde{\phi}),\\
\tilde{F}_{32}(\tilde{v},\tilde{\phi}) &= \frac{1}{2}(\Phi\tilde{\phi}) * (\underbrace{A_y V}\,\tilde{v}) + (V\tilde{v}) * (\underbrace{A_y \Phi}\,\tilde{\phi}).
\end{aligned} \tag{4}$$

The POD version of SWE model

- The coefficient matrices defined in the linear terms of the POD reduced system as well as the coefficient matrices in the nonlinear functions (i.e. $A_xU, A_yU,$ $A_xV, A_yV, A_x\Phi, A_y\Phi \in \mathbb{R}^{n\times k}$ grouped by the curly braces) can be precomputed, saved and re-used in all time steps.
- However, performing the componentwise multiplications in (4) and computing the projected nonlinear terms in (3)

$$\underbrace{U^T}_{k\times n_{xy}} \underbrace{\tilde{F}_{11}(\tilde{u},\tilde{\phi})}_{n_{xy}\times 1}, U^T\tilde{F}_{12}(\tilde{u},\tilde{v}), V^T\tilde{F}_{21}(\tilde{u},\tilde{v}), \\ V^T\tilde{F}_{22}(\tilde{v},\tilde{\phi}), \Phi^T\tilde{F}_{31}(\tilde{u},\tilde{\phi}), \Phi^T\tilde{F}_{32}(\tilde{v},\tilde{\phi}), \tag{5}$$

still have computational complexities depending on the dimension n_{xy} of the original system from both evaluating the nonlinear functions and performing matrix multiplications to project on POD bases.

Discrete Empirical Interpolation Method (DEIM)

- Let $f : D \rightarrow \mathbb{R}^n,\ D \subset \mathbb{R}^n$ be a nonlinear function. If $U = \{u_l\}_{l=1}^m,\ u_i \in \mathbb{R}^n,\ i = 1, .., m$ is a linearly independent set, for $m \leq n$, then for $\tau \in D$, the DEIM approximation of order m for $f(\tau)$ in the space spanned by $\{u_l\}_{l=1}^m$ is given by

$$f(\tau) \approx Uc(\tau),\ U \in \mathbb{R}^{n\times m},\ c(\tau) \in \mathbb{R}^m. \quad (6)$$

- The basis U can be constructed effectively by applying the POD method on the nonlinear snapshots $f(\tau^{t_i}),\ i = 1, .., n_s$.

Discrete Empirical Interpolation Method (DEIM)

- Interpolation is used to determine the coefficient vector $c(\tau)$ by selecting m rows $\rho_1, .., \rho_m,\ \rho_i \in \mathbb{N}^*$, of the overdetermined linear system (6)

$$\underbrace{\begin{bmatrix} f_1(\tau) \\ \vdots \\ \vdots \\ f_n(\tau) \end{bmatrix}}_{f(\tau)\in\mathbb{R}^n} = \underbrace{\begin{bmatrix} u_{11} & \dots & u_{1m} \\ \vdots & \dots & \vdots \\ \vdots & \dots & \vdots \\ u_{n1} & \dots & u_{nm} \end{bmatrix}}_{U\in\mathbb{R}^{n\times m}} \underbrace{\begin{bmatrix} c_1(\tau) \\ \vdots \\ c_m(\tau) \end{bmatrix}}_{c(\tau)\in\mathbb{R}^m}.$$

to form a m-by-m linear system

$$\underbrace{\begin{bmatrix} f_{\rho_1}(\tau) \\ \vdots \\ f_{\rho_m}(\tau) \end{bmatrix}}_{f_{\vec{\rho}}(\tau)\in\mathbb{R}^m} = \underbrace{\begin{bmatrix} u_{\rho_1 1} & \dots & u_{\rho_1 m} \\ \vdots & \dots & \vdots \\ u_{\rho_m 1} & \dots & u_{\rho_m m} \end{bmatrix}}_{U_{\vec{\rho}}\in\mathbb{R}^{m\times m}} \underbrace{\begin{bmatrix} c_1(\tau) \\ \vdots \\ c_m(\tau) \end{bmatrix}}_{c(\tau)\in\mathbb{R}^m}.$$

Discrete Empirical Interpolation Method (DEIM)

- In the short notation form

$$U_{\vec{\rho}}c(\tau) = f_{\vec{\rho}}(\tau).$$

- Lemma 2.3.1 in Chaturantabut (2008) proves that $U_{\vec{\rho}}$ is invertible, thus we can uniquely determine $c(\tau)$

$$c(\tau) = U_{\vec{\rho}}^{-1} f_{\vec{\rho}}(\tau).$$

- The DEIM approximation of $F(\tau) \in \mathbb{R}^n$ is

$$f(\tau) \approx Uc(\tau) = UU_{\vec{\rho}}^{-1} f_{\vec{\rho}}(\tau).$$

Discrete Empirical Interpolation Method (DEIM)

- $U_{\vec{\rho}}$ and $f_{\vec{\rho}}(\tau)$ can be written in terms of U and $f(\tau)$

$$U_{\vec{\rho}} = P^T U, \qquad f_{\vec{\rho}}(\tau) = P^T f(\tau)$$

  where
  $P = [e_{\rho_1}, .., e_{\rho_m}] \in \mathbb{R}^{n\times m}$, $e_{\rho_i} = [0, ..0, \underbrace{1}_{\rho_i}, 0, .., 0]^T \in \mathbb{R}^n$.

- The DEIM approximation of $f \in \mathbb{R}^n$ becomes

$$f(\tau) \approx U(P^T U)^{-1} P^T f(\tau).$$

- Using the DEIM approximation, the complexity for computing the nonlinear term of the reduced system in each time step is now independent of the dimension n of the original full-order sytem.
- The only unknowns need to be specified are the indices $\rho_1, \rho_2, ..., \rho_m$ or matrix P. The algorithm can be found in Chaturantabut (2008), Chaturantabut and Sorensen (2010).

DEIM: Algorithm for Interpolation Indices

INPUT: $\{u_l\}_{l=1}^{m} \subset \mathbb{R}^n$ (linearly independent):

OUTPUT: $\vec{\rho} = [\rho_1, .., \rho_m] \in \mathbb{R}^m$

1. $[|\psi| \quad \rho_1] = max|u_1|, \psi \in \mathbb{R}$ and ρ_1 is the component position of the largest absolute value of u_1, with the smallest index taken in case of a tie.
2. $U = [u_1], \; P = [e_{\rho_1}], \; \vec{\rho} = [\rho_1]$.
3. For $l = 2, .., m$ do
   - **a** Solve $(P^T U)c = P^T u_l$ for c
   - **b** $r = u_l - Uc$
   - **c** $[|\psi| \quad \rho_l] = max\{|r|\}$
   - **d** $U \leftarrow [U \quad u_l], \; P \leftarrow [P \quad e_{\rho_l}], \; \vec{\rho} \leftarrow \begin{bmatrix} \vec{\rho} \\ \rho_l \end{bmatrix}$
4. end for.

The DEIM version of SWE model

- The projected nonlinear functions can be approximated by DEIM in a form that enables precomputation so that the computational cost is decreased and independent of the original system.
- Only a few entries of the nonlinear term corresponding to the specially selected interpolation indices from DEIM must be evaluated at each time step.
- DEIM approximation is applied to each of the nonlinear functions $\tilde{F}_{11}, \tilde{F}_{12}, \tilde{F}_{21}, \tilde{F}_{22}, \tilde{F}_{31}, \tilde{F}_{32}$ defined in (4).

The DEIM version of SWE model

- Let $U^{F_{11}} \in \mathbb{R}^{n_x y \times m},\ m \le n$, be the POD basis matrix of rank m for snapshots of the nonlinear function $\tilde{F}_{11}$ (obtained from ADI FD scheme).
- The DEIM approximation of F_{11} is

$$\tilde{F}_{11} \approx U^{F_{11}}(P_{F_{11}}^T U^{F_{11}})^{-1}\tilde{F}_{11}^m,$$

so the projected nonlinear term $U^T\tilde{F}_{11}(\tilde{u},\tilde{\phi})$ in the POD reduced system (3) can be approximated as

$$U^T\tilde{F}_{11}(\tilde{u},\tilde{\phi}) \approx \underbrace{U^T U^{F_{11}}(P_{F_{11}}^T U^{F_{11}})^{-1}}_{E_1 \in \mathbb{R}^{k\times m}} \underbrace{\tilde{F}_{11}^m(\tilde{u},\tilde{\phi})}_{m\times 1},$$

where $\tilde{F}_{11}^m(\tilde{u},\tilde{\phi}) = P_{F_{11}}^T\tilde{F}_{11}(\tilde{u},\tilde{\phi})$.
- Since $\tilde{F}_{11}$ is a pointwise function, $\tilde{F}_{11}^m : \mathbb{R}^k \times \mathbb{R}^k \to \mathbb{R}^m$ can be defined as

$$\tilde{F}_{11}^m(\tilde{u},\tilde{\phi}) = (P_{F_{11}}^T U\tilde{u}) * (\underbrace{P_{F_{11}}^T A_x U}\tilde{u}) + \frac{1}{2}(P_{F_{11}}^T \Phi\tilde{\phi}) * (\underbrace{P_{F_{11}}^T A_x \Phi}\tilde{\phi})$$

- Similarly we obtain the DEIM approximation for the rest of the projected nonlinear terms in (5)

Numerical Results

- The domain was discretized using a mesh of 301×221 points, with $\Delta x = \Delta y = 20km$. Thus the dimension of the full-order discretized model is 66521. The integration time window for ADI FD scheme was $24h$ and we used 91 time steps ($NT = 91$) with $\Delta t = 960s$.

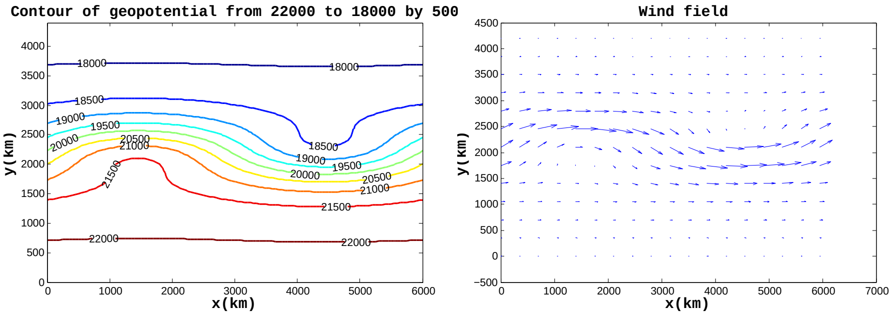
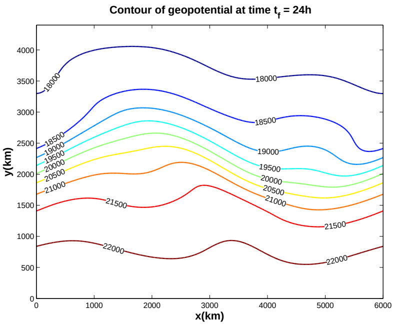


Fig.1 Initial condition: Geopotential height field for the Grammeltvedt initial condition (left). Wind field (arrows are scaled by a factor of 1km) calculated from the geopotential field by using the geostrophic approximation (right).

Numerical Results

- Courant-Friedrichs-Levy (CFL) condition: $\sqrt{gh}(\frac{\Delta t}{\Delta x}) < 7.188$.
- The nonlinear algebraic systems of ADI FD SWE scheme were solved with the Quasi-Newton method and the LU decomposition was performed only once every $6-th$ time step.



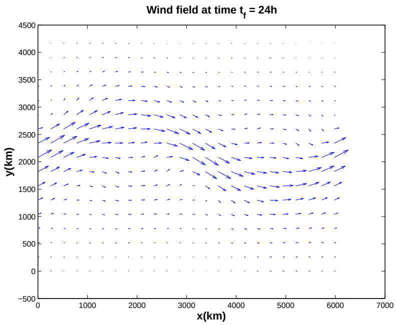
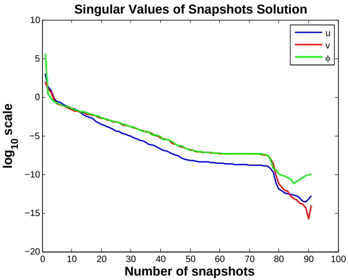


Fig.2 The geopotential field (left) and the wind field (the velocity unit is 1km/s) at $t = t_f = 24h$ obtained using the ADI FD SWE scheme for $\Delta t = 960s$.

Numerical Results



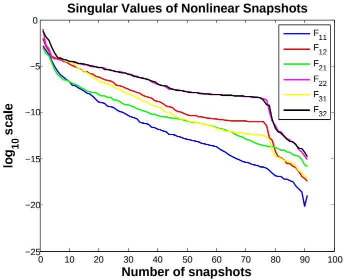


Fig.3 The decay around the singular values of the snapshots solutions for $u, v,\ \phi$ and nonlinear functions for $\Delta t = 960s$. The POD basis functions were constructed using 91 snapshots.

Numerical Results

- The dimension of the POD bases for each variable was taken to be 35.

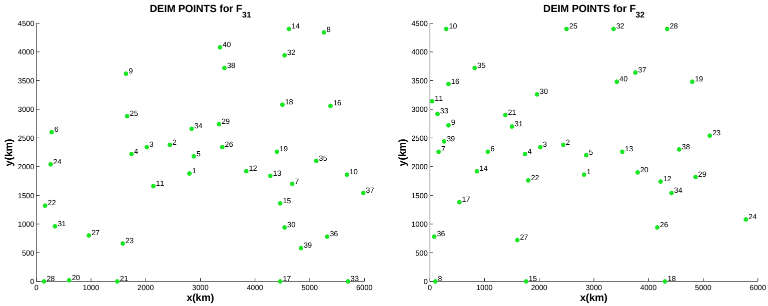


Fig.4 First 40 **points selected by DEIM for the nonlinear functions** F_{31} **(left) and** F_{32} **(right)**

- We also propose an Euler explicit FD SWE scheme as the starting point for a POD, POD/DEIM reduced model. The POD bases were constructed using the same 91 snapshots as in the POD ADI SWE case, only this time the Galerkin projection was applied to the Euler FD SWE model.
- The root mean square error was employed in order to compare the POD and POD/DEIM techniques at time $t = 24h$.

	ADI SWE	POD ADI SWE	POD/DEIM ADI SWE	POD EE SWE	POD/DEIM EE SWE
CPU time seconds	73.081	43.021	0.582	43.921	0.639
$RMSE_{\phi}$	-	5.416e-005	9.668e-005	1.545e-004	1.792e-004
$RMSE_{u}$	-	1.650e-004	2.579e-004	1.918e-004	3.126e-004
$RMSE_{v}$	-	8.795e-005	1.604e-004	1.667e-004	2.237e-004

Table 2 CPU time gains and the root mean square errors for each of the model variables at $t = t_f$. The POD bases dimensions were taken as 35 capturing more than 99.9% of the system energy. 90 DEIM points were chosen.

Numerical Results

- POD/DEIM Nonlinear model order reduction of an ADI implicit shallow water equations model, R. Stefanescu and I.M. Navon, Journal of Computational Physics, Volume 237, 15 March 2013, Pages 95–114.

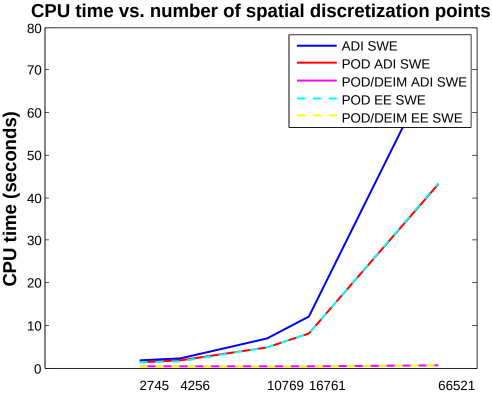


Fig.5 Cpu time vs. Spatial discretization points; POD DIM = 35, No. DEIM points = 90.

R. Ştefănescu I.M. Navon

POD/DEIM 4DVAR applied to a Leapfrog scheme of SWE

- We aim to incorporate data assimilation system into POD/DEIM model reduction
- We applied DEIM to a POD Leapfrog scheme of the SWE on a rectangular domain.
- We build the POD/DEIM 4D VAR minimization application to emphasize the DEIM impact in the context of reduced PDE constrained optimization.

POD/DEIM 4D VAR

- The aim of 4-D VAR data assimilation is to reconcile observations with model predictions subject to the model serving as a strong constraint. ,
- In the full high-fidelity nonlinear 4-D VAR, this process is implemented by minimizing the following cost functional with respect to the control variables:

$$J(w_0) = (w_0 - w^b)^T B^{-1}(w_0 - w^b) + \sum_{k=1}^{Nt} \gamma_w (H_k w_k - w_k^o)^T O^{-1}(H_k w_k - w_k^o), \tag{7}$$

where $w_0 = (u_0, v_0, \phi_0)$ and γ_w is a weight function.

POD/DEIM 4D VAR

- In the data assimilation process of the SWE model, we just consider the observation information and don't involve the background information.
- The POD/DEIM reduced order cost functional in POD/DEIM 4-D VAR assumes the form

$$J^{POD/DEIM}(w_0^{POD/DEIM}) = \sum_{k=1}^{Nt} \gamma_w (w_k^{POD/DEIM} - w_k^o)^T O^{-1} (w_k^{POD/DEIM} - w_k^o) \tag{8}$$

where $w_0^{POD/DEIM}$ is the reduced order control vector $w_0^{POD/DEIM} = (u_0^{POD/DEIM}, v_0^{POD/DEIM}, \phi_0^{POD/DEIM}) = (\tilde{u}_0, \tilde{v}_0, \tilde{\phi}_0)$ and $w_k^{POD/DEIM}$ is the model prediction obtained from the POD/DEIM reduced order forward model.

POD/DEIM 4D VAR

- In the process of minimizing the cost functional with respect to the control variables, the limited-memory BFGS (L-BFGS) quasi-Newton method is applied. The gradient of the reduced cost functional with respect to the control variables can be expressed as

$$\nabla_{\tilde{u}_0} J^{POD} = {U^{POD}}^T (\nabla_{u_0} J)|_{u_0 = U^{POD}\tilde{u}}$$

$$\nabla_{\tilde{v}_0} J^{POD} = {V^{POD}}^T (\nabla_{v_0} J)|_{v_0 = V^{POD}\tilde{v}}$$

$$\nabla_{\tilde{\phi}_0} J^{POD} = {\Phi^{POD}}^T (\nabla_{\phi_0} J)|_{\phi_0 = \Phi^{POD}\tilde{\phi}}$$

- Since our work is in progress, for this presentation, the gradient of the cost functional was calculated using the adjoint of the forward POD model.

POD/DEIM 4D VAR

- Consequently, computational savings are mainly achieved by a drastic reduction in the number of iterations due to the low dimension of the optimization problem.
- Since the validity of the POD/DEIM reduced order model is limited to the vicinity of the control space (POD bases and POD/DEIM bases), it might not be an appropriate model when the latest state is significantly different from the one on which the POD/DEIM reduced order model is based.

Ad-hoc POD/DEIM 4D VAR

An 'ad-hoc' adaptive POD/DEIM 4-D VAR algorithm is proposed as follows:

1. Generate a set of snapshots (including the nonlinear ones corresponding to the quadratic non-linear terms) from the solution of the full forward model and construct the POD/DEIM reduced order model
2. Perform iterations for the optimization problem using the reduced order model with the L-BFGS method and calculate the cost functional J_n where n is the n^{th} L-BFGS iteration.
3. Check the value of the cost functional.

Ad-hoc POD/DEIM 4D VAR

- If $|J_n| < \epsilon$ where ϵ is the tolerance for the optimization, then stop, the POD/DEIM 4-D VAR data assimilation is completed;
- If $|J_n| > \epsilon$ and $J_{n-1} - J_n > \eta$ $\eta > 0$, then set $n = n + 1$ and go back to (2);
- If $|J_n| > \epsilon$ and $J_{n-1} - J_n < \eta$, project back the reduced order control variables from the latest optimization iteration to the original space, then go to (1).

Numerical Results

- The domain was discretized using a mesh of 21×21 points, $\Delta x = 600km$, $\Delta y = 440km$. The integration window was $5h$ and we used 60 time steps ($Nt = 60$) with $\Delta t = 300s$.
- Leapfrog scheme was employed to obtain the snapshots for ROMs models.

Numerical Results

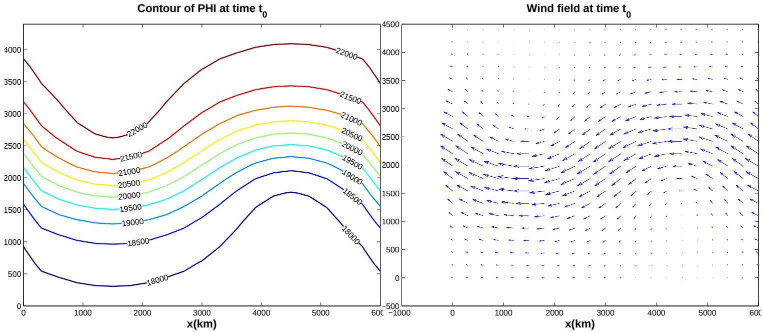


Fig.6 Initial condition: Geopotential height field for the Grammeltvedt initial condition (left). Wind field calculated from the geopotential field by using the geostrophic approximation (right).

Numerical Results

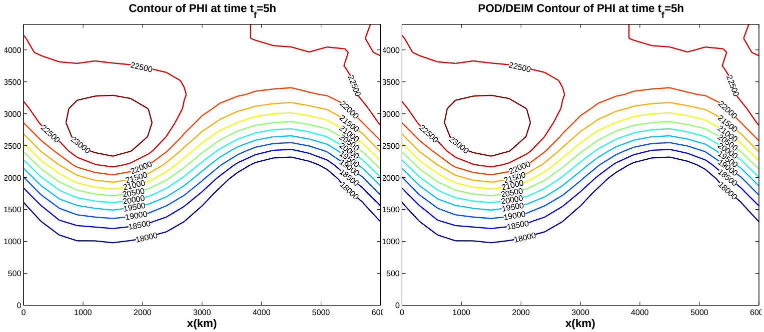


Fig.7 Geopotential height field at $t = t_f$ obtained using Leapfrog full scheme (left) and POD/DEIM Leapfrog scheme (right)

Numerical Results

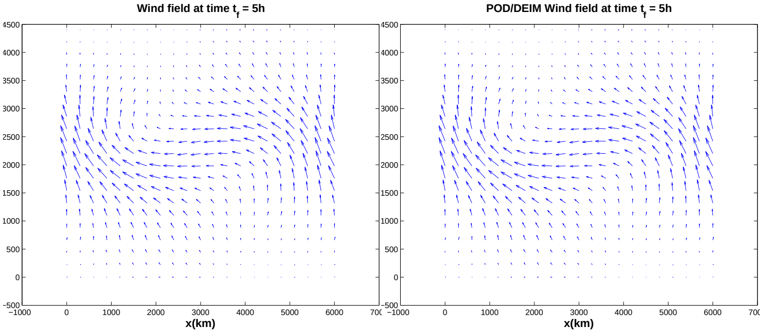


Fig.8 Wind field at $t = t_f$ obtained using Leapfrog full scheme (left) and POD/DEIM Leapfrog scheme (right)

Numerical Results

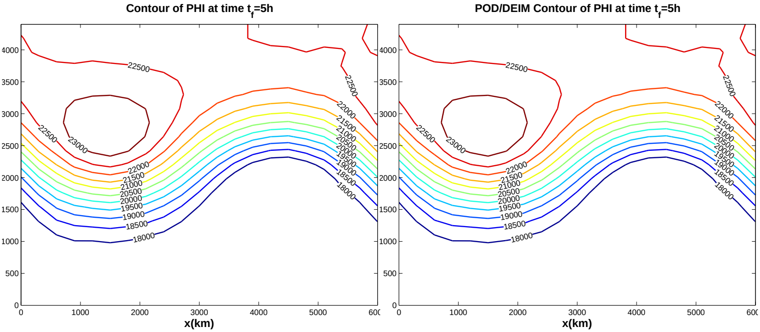


Fig.9 Geopotential height field at $t = t_f$ obtained using Leapfrog full scheme (left) and POD/DEIM Leapfrog scheme (right)

Numerical Results

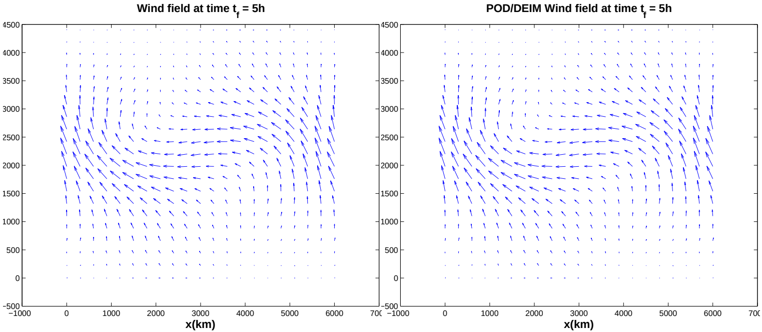


Fig.10 Wind field at $t = t_f$ obtained using Leapfrog full scheme (left) and POD/DEIM Leapfrog scheme (right)

- We employed the root mean square error calculation in order to compare the POD and POD/DEIM techniques at time $t = 5h$.

	Leapfrog SWE	Leapfrog POD SWE	Leapfrog POD/DEIM SWE
CPU time seconds	0.1486	9.9E-03	1.3E-03
$RMSE_{\phi}$	-	2.9655 E-04	3.4513 E-04
$RMSE_{u}$	-	6.13149 E-07	6.794 E-07
$RMSE_{v}$	-	1.03321 E-06	1.045 E-07

Table 2 CPU time gains and the root mean square errors for each of the model variables at $t = t_f$. The POD bases dimensions were taken as 15 capturing more than 99.9% of the system energy. 20 DEIM points were chosen.

Numerical Results

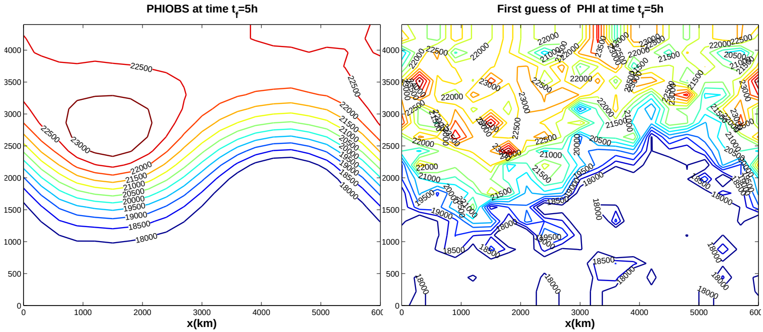


Fig.11 Phi observations (left) and first guess (right) for $t = t_f$

Numerical Results

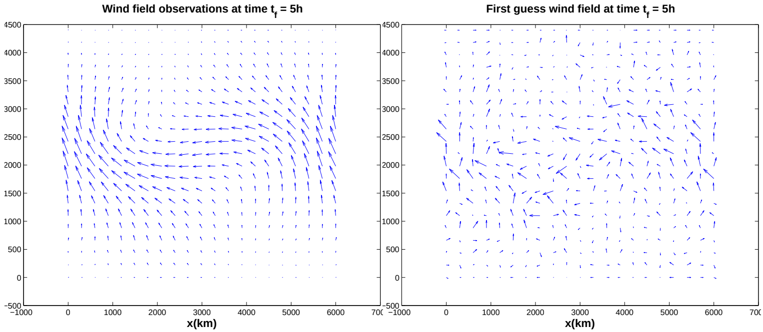


Fig.12 Wind field observations (left) and first guess (right) for $t = t_f$

Numerical Results

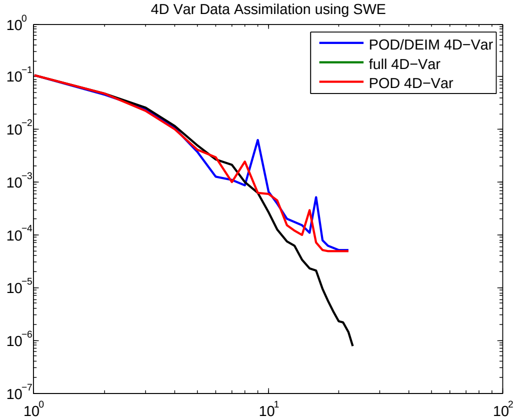


Fig.13 The performance of minimization of the cost functional for the Ad Hoc POD 4-D VAR, POD/DEIM 4-D VAR and full 4-D VAR.

Numerical Results

	Full 4D-VAR	POD 4D-VAR	POD/DEIM 4D-VAR
CPU time seconds	221.4154	52.0852	46.61469

CPU time for updating POD basis	0.1358
CPU time for updating POD/DEIM basis	0.4036

- Updating POD basis costs the same as one forward integration in time of the full model
- Updating POD/DEIM basis costs the same as three forward integration in time of the full model

Discussion and Future Work

- For our test, we chose the number of POD basis to be equal with 10 and the number of DEIM interpolation points used was also 10. By increasing the control space dimension and the number of DEIM points we expect that POD/DEIM and POD 4DVAR minimizations to perform better.
- The Leapfrog scheme should also include the Robert Asselin Williams (RAW) filter to dump the computational mode.
- POD/DEIM 4DVAR gradient test was affected by the inconsistency between the forward POD/DEIM model and the POD adjoint model.
- Using discretize and differentiate we'll obtain a faster POD/DEIM adjoint model comparing with POD adjoint model.

Discussion and Future Work

- The discrete POD/DEIM adjoint will reduce the computational complexity of POD adjoint model due to its depending on the nonlinear full dimension model and regain the full model reduction expected from the POD model.
- The trust region scheme will be combined with POD 4-D VAR data assimilation in order to solve the reduced order inverse problem more efficiently.

Conclusion and future research

- POD/DEIM Nonlinear model order reduction of an ADI implicit shallow water equations model, R. Stefanescu and I.M. Navon, Journal of Computational Physics, Volume 237, 15 March 2013, Pages 95–114.
- To obtain the approximate solution in case of both POD and POD/DEIM reduced systems, one must store POD or POD/DEIM solutions of order $O(kNT)$, k being the POD bases dimension and NT the number of time steps in the integration window.
- The coefficient matrices that must be retained while solving the POD reduced system are of order of $O(k^2)$ for projected linear terms and $O(n_{xy}k)$ for the nonlinear term, where n_{xy} is the space dimension.

Conclusion and future research

- In the case of solving POD/DEIM reduced system the coefficient matrices that need to be stored are of order of $O(k^2)$ for projected linear terms and $O(mk)$ for the nonlinear terms, where m is the number of DEIM points determined by the DEIM indexes algorithm, $m \ll n_{xy}$.
- Therefore DEIM improves the efficiency of the POD approximation and achieves a complexity reduction of the nonlinear term with a complexity proportional to the number of reduced variables.
- We proved the efficiency of DEIM using three different schemes, the ADI FD SWE fully implicit model, the Euler explicit FD SWE scheme, the Leapfrog scheme.
- Trust Region POD/DEIM 4DVAR algorithm shows great potential having the ability to decrease the CPU time of POD 4DVAR by the same rate as POD/DEIM reduces the computational complexity of the POD forward model once the POD/DEIM adjoint is constructed.