Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Computer Science 00 (2013) 000–000

Procedia Computer Science

www.elsevier.com/locate/procedia

International Conference on Computational Science, ICCS 2013

Dynamic sensor network configuration in InfoSymbiotic systems using model singular vectors

Adrian Sandu[a], Alexandru Cioaca[a], Vishwas Rao[a]

[a]*Computational Science Laboratory, Department of Computer Science, Virginia Tech, 2202 Kraft Drive, Blacksburg,VA 24060, USA*

Abstract

Data assimilation is an important data-driven application (DDDAS) where measurements of the real system are used to constrain simulation results. This paper describes a methodology for dynamically configuring sensor networks in data assimilation systems based on numerical models of time-evolving differential equations. The proposed methodology uses the dominant model singular vectors, which reveal the directions of maximal error growth. New sensors are dynamically placed such as to minimize an estimation error energy norm. A shallow water test problem is used to illustrate our approach.

Keywords: Dynamic data driven applications, model singular vectors, sensor network configuration.

1. Introduction

Data assimilation is an important application of data-driven application systems (DDDAS, or InfoSymbiotic systems) where measurements of the real system are used to constrain simulation results. Specifically, data assimilation is the process that combines prior information, numerical model predictions, observational data, and the corresponding error statistics, to produce a better estimate of the state of a physical system [1, 2]. The four-dimensional variational (4D-Var) approach is formulated as a nonlinear optimization problem constrained by the numerical model. The initial conditions (as well as boundary conditions, forcings, or model parameters) are adjusted such as to minimize the discrepancy between the model trajectory and a set of time-distributed observations. In real-time operations, the analysis is performed in cycles: observations within an assimilation time window are used to obtain an optimal trajectory, which provides the initial condition for the next time window, and the process is repeated.

Since data assimilation is a cyclic process, we are interested to improve the efficiency of next assimilation windows by analyzing the previous ones. This paper describes a methodology for dynamically configuring sensor networks in 4D-Var data assimilation applications. At the end of each assimilation window the areas of maximum energy impact are estimated using Hessian singular vectors. New sensors dynamically placed in those areas will improve the quality of the results for the next assimilation window. Specifically, they provide a maximal reduction of the L^2 norm of the assimilation error.

*Corresponding author. Tel.: +1-540-231-2193 ; fax: +1-540-231-6075 .
E-mail address: {sandu, alexgc, visrao}@cs.vt.edu.

The singular vector method searches for the error structures in the analysis field at current time which, propagated by the forecast model, achieve maximum growth at the verification time $t_{\rm v}$. Singular vectors (SVs) are the directions of fastest error growth over a finite time interval [3, 4, 5, 6]. Buizza and Montani [7] showed that SVs can identify the most sensitive regions of the atmosphere for targeted observations. They are useful as long as the linearity assumption of error propagation holds [8]. Majumdar et al.[9] compare the SV approach for observation targeting to the ensemble transform Kalman filter. Palmer et al. [10] argue that for predictability studies an appropriate metric is the perturbation energy. Daescu and Navon [11, 12] discuss the adaptive observation problem in the context of 4D-Var data assimilation. Estimation of the optimal placement of adaptive observations is also discussed in [13, 14]. Leutbecher [15] derives optimal locations of observations by minimizing the variance of the assimilated field; a computationally tractable problem is obtained by projecting the covariance on the subspace of the dominant singular vectors.

Observations can be dynamically placed in well-chosen locations such as to reduce the initial condition uncertainties and decrease forecast errors. A number of methods were proposed to "target observations", i.e. to select areas where additional observations are expected to improve considerably the skill of a given forecast. Singular vectors identify sensitive regions of the atmospheric flow and can be used to optimally configure the observational network.

2. Data assimilation

Data assimilation produces improved estimates of the true state $\mathbf{x}^{\rm true}$ of a physical system by combining information from three different sources: the physical and chemical laws of evolution (encapsulated in the model), the reality (as captured by the observations), and the current best estimate of the distribution of pollutants in the atmosphere (encapsulated in the prior) – all with associated errors [16].

The background (prior) probability density encapsulates our current knowledge of the tracer distribution. A typical assumption is that this is a normal probability density, $\mathbf{x}_0^{\rm true} \in \mathcal{N}\left(\mathbf{x}_0^{\rm b}, \mathbf{B}_0\right)$, where $\mathbf{x}_0^{\rm b} \in \mathbb{R}^n$ is the background initial state and $\mathbf{B}_0 \in \mathbb{R}^{n\times n}$ is the background error covariance matrix.

The model encapsulates our knowledge about physical and chemical laws that govern the evolution of the physical system. It evolves an initial state $\mathbf{x}_0 \in \mathbb{R}^n$ at time t_0 to future state values $\mathbf{x}_i \in \mathbb{R}^n$ at future times t_i,

$$\mathbf{x}_i = \mathcal{M}_{t_0\to t_i}\left(\mathbf{x}_0\right) . \tag{1}$$

Perturbations (small errors) evolve according to the tangent linear model (TLM)

$$\delta\mathbf{x}(t_{\rm F}) = \mathbf{M}_{t_0\to t_{\rm F}}\,\delta\mathbf{x}(t_0) , \tag{2}$$

where $\mathbf{M} = \mathcal{M}'$ is the linearized model solution operator. Adjoint variables evolve according to the adjoint model

$$\lambda(t_0) = \mathbf{M}^*_{t_{\rm F}\to t_0}\lambda(t_{\rm F}) . \tag{3}$$

Here $\mathbf{M}^*_{t_{\rm F}\to t_0} = \mathbf{M}^T_{t_0\to t_{\rm F}}$ denotes the adjoint of the linearized solution operator $\mathbf{M}$. The state error covariance matrix is propagated from t_0 to $t_{\rm F}$ according to

$$\mathbf{P}(t_{\rm F}) = \mathbf{M}_{t_0\to t_{\rm F}}\,\mathbf{P}(t_0)\,\mathbf{M}^*_{t_{\rm F}\to t_0} + \mathbf{Q} . \tag{4}$$

The additional term $\mathbf{Q}$ represents the covariance of the model errors.

Observations are sparse and noisy snapshots of reality available at several discrete time moments. Specifically, measurements $\mathbf{y}_i \in \mathbb{R}^m$ of the physical state are taken at times t_i, $i = 1, \cdots, N$. In order to relate the model state to observations we consider the relation

$$\mathbf{y}_i = \mathcal{H}\left(\mathbf{x}_i^{\rm true}\right) - \varepsilon_i^{\rm obs} , \quad i = 1, \cdots, N , \tag{5}$$

where $\mathcal{H}$ is the observation operator, and $\varepsilon_i^{\rm obs}$ the observation errors, assumed to be normally distributed, $\varepsilon_i^{\rm obs} \in \mathcal{N}\left(\mathbf{0}, \mathbf{R}_i\right)$. Observation errors at different times ($\varepsilon_i^{\rm obs}$ and $\varepsilon_j^{\rm obs}$ for $i \neq j$) are assumed to be independent.

Based on these three sources of information data assimilation computes the analysis (posterior) probability density. The best estimate $\mathbf{x}^{\mathrm{a}}$ of the true state obtained from analysis distribution is called the aposteriori, or the *analysis state*. The strongly-constrained 4D-Var data assimilation looks for the maximum a posteriori (MAP) estimate $\mathbf{x}_0^{\mathrm{a}}$ of the true initial conditions by solving the optimization problem:

$$\mathbf{x}_0^{\mathrm{a}} = \arg\min \mathcal{J}\left(\mathbf{x}_0\right) \quad \text{subject to: } \mathbf{x}_i = \mathcal{M}_{t_0 \to t_i}\left(\mathbf{x}_0\right)\ ,\ i = 1, \cdots, N\,, \tag{6}$$

where

$$\mathcal{J}\left(\mathbf{x}_0\right) = \frac{1}{2}\left(\mathbf{x}_0 - \mathbf{x}_0^{\mathrm{b}}\right)^T\ \mathbf{B}_0^{-1}\ \left(\mathbf{x}_0 - \mathbf{x}_0^{\mathrm{b}}\right) + \frac{1}{2}\sum_{i=1}^{N}\left(\mathcal{H}(\mathbf{x}_i) - \mathbf{y}_i\right)^T\ \mathbf{R}_i^{-1}\ \left(\mathcal{H}(\mathbf{x}_i) - \mathbf{y}_i\right)\ . \tag{7}$$

The gradient of (7) reads

$$\nabla_{\mathbf{x}_0}\mathcal{J}\left(\mathbf{x}_0\right) = \mathbf{B}_0^{-1}\ \left(\mathbf{x}_0 - \mathbf{x}_0^{\mathrm{b}}\right) + \sum_{i=1}^{N}\mathbf{M}_{t_i \to t_0}^{*}\,\mathbf{H}_i^T\,\mathbf{R}_i^{-1}\ \left(\mathcal{H}(\mathbf{x}_i) - \mathbf{y}_i\right)\ . \tag{8}$$

The Hessian of (7) is obtained by differentiating (8)

$$\nabla^2_{\mathbf{x}_0,\mathbf{x}_0}\mathcal{J}\left(\mathbf{x}_0\right) \quad = \quad \mathbf{B}_0^{-1} + \sum_{i=1}^{N}\mathbf{M}_{t_i \to t_0}^{*}\,\mathbf{H}_i^T\,\mathbf{R}_i^{-1}\,\mathbf{H}_i\,\mathbf{M}_{t_0 \to t_i} + \sum_{i=1}^{N}\left(\frac{d}{d\,\mathbf{x}_0}\,\mathbf{M}_{t_i \to t_0}^{*}\right)\mathbf{H}_i^T\,\mathbf{R}_i^{-1}\ \left(\mathcal{H}(\mathbf{x}_i) - \mathbf{y}_i\right)\ .$$

The Gauss-Newton Hessian approximation discards the last term containing second derivatives of the model solution operator times the residual. The second order adjoint (SOA) of the chemical transport model [17, 18] computes matrix vector products between the Hessian and user-supplied vectors. The SOA model provides information about the aposteriori error via the observation that the Hessian inverse approximates the posterior error covariance [19]

$$\mathbf{A}_0 \approx \left(\nabla^2_{\mathbf{x}_0,\mathbf{x}_0}\mathcal{J}(\mathbf{x}_0^{\mathrm{a}})\right)^{-1}\ .$$

3. Model singular vectors

Singular vectors (SVs) determine the most rapidly growing perturbations in the atmosphere. The magnitude of the perturbation at the initial time t_0 is measured in the L^2 ("energy") norm defined by a symmetric positive definite matrix $\mathbf{E}$

$$\|\,\delta\mathbf{x}(t_0)\,\|^2_{\mathbf{E}} = \langle\ \delta\mathbf{x}(t_0)\,,\ \mathbf{E}\ \delta\mathbf{x}(t_0)\ \rangle\,. \tag{9}$$

Similarly, the perturbation magnitude at the verification time t_{v} is measured in a norm defined by a positive definite matrix $\mathbf{F}$

$$\|\,\delta\mathbf{x}(t_{\mathrm{v}})\,\|^2_{\mathbf{F}} = \langle\ \delta\mathbf{x}(t_{\mathrm{v}})\,,\ \mathbf{F}\ \delta\mathbf{x}(t_{\mathrm{v}})\ \rangle\,. \tag{10}$$

We call the norms (9) and (10) squared the "perturbation energies".

Our main interest is to minimize the forecast uncertainty over a well defined area (the "verification domain" $\Omega^{\mathrm{v}} \subset \Omega$) at a well defined time (the "verification time" t_{v}). We define a spatial projection operator $\mathbf{\Pi}$ from the whole model domain to the verification domain:

$$\mathbf{\Pi} : \Omega \subset \mathbb{R}^n \longrightarrow \Omega^{\mathrm{v}} \subset \mathbb{R}^{n_{\mathrm{v}}}\,. \tag{11}$$

The ratio between perturbation energies at t_{v} (over the verification domain) and at t_0 (over the entire domain) offers a measure of error growth:

$$\sigma^2 = \frac{\|\mathbf{\Pi}\,\delta\mathbf{x}(t_{\mathrm{v}})\|^2_{\mathbf{F}}}{\|\delta\mathbf{x}(t_0)\|^2_{\mathbf{E}}} = \frac{\left\langle \delta\mathbf{x}(t_0), \mathbf{M}^{*}_{t_{\mathrm{v}} \to t_0}\mathbf{\Pi}^{*}\,\mathbf{F}\,\mathbf{\Pi}\,\mathbf{M}_{t_0 \to t_{\mathrm{v}}}\delta\mathbf{x}(t_0)\right\rangle}{\langle \delta\mathbf{x}(t_0), \mathbf{E}\,\delta\mathbf{x}(t_0)\rangle}\,. \tag{12}$$

Model singular vectors (SVs) are defined as the directions of maximal error growth, i.e. the vectors $s_k(t_0)$ that maximize the ratio σ^2 in equation (12). These directions are the solutions of the generalized eigenvalue problem

$$\mathbf{M}^{*}_{t_{\mathrm{v}} \to t_0}\,\mathbf{\Pi}^{*}\,\mathbf{F}\,\mathbf{\Pi}\,\mathbf{M}_{t_0 \to t_{\mathrm{v}}}\ s_k(t_0) = \sigma_k^2\ \mathbf{E}\ s_k(t_0)\,. \tag{13}$$

Using the square root of the the symmetric positive definite matrix $\mathbf{E}$ the generalized eigenvalue problem (13) can be reduced to a simple eigenvalue problem

$$\mathbf{E}^{-\frac{1}{2}}\,\mathbf{M}^{*}_{t_{\mathrm{v}}\rightarrow t_0}\,\mathbf{\Pi}^{*}\,\mathbf{F}\mathbf{\Pi}\,\mathbf{M}_{t_0\rightarrow t_{\mathrm{v}}}\,\mathbf{E}^{-\frac{1}{2}}\,v_k=\sigma_k^2\,v_k(t_0)\,,\quad v_k=\mathbf{E}^{\frac{1}{2}}\,s_k(t_0)\,. \tag{14}$$

The eigenvalue problem (14) can be solved efficiently using the software package ARPACK [20]. Furthermore, $v_k(t_0)$ are the left singular vectors in the singular value decomposition

$$\mathbf{F}^{\frac{1}{2}}\,\mathbf{\Pi}\,\mathbf{M}_{t_0\rightarrow t_{\mathrm{v}}}\,\mathbf{E}^{-\frac{1}{2}}=U\cdot\Sigma\cdot V^{T}\quad\text{where}\quad\Sigma=\mathrm{diag}_k\{\sigma_k\}\,,\quad\sigma_k\,u_k=\mathbf{F}^{\frac{1}{2}}\,\mathbf{\Pi}\,s_k(t_{\mathrm{F}})\,. \tag{15}$$

The SVs s_k are $\mathbf{E}$-orthogonal at t_0 and $\mathbf{F}$-orthogonal at t_{F}

$$\left\langle\, s_k(t_0),\,\mathbf{E}s_j(t_0)\,\right\rangle=0\qquad\text{and}\qquad\left\langle\,\mathbf{\Pi}s_k(t_{\mathrm{F}}),\,\mathbf{F}\mathbf{\Pi}s_j(t_{\mathrm{F}})\,\right\rangle=0\qquad\text{for}\quad j\neq k\,. \tag{16}$$

The equations (15) and (16) justify the name of "total energy singular vectors". The singular value decomposition of the linear operator $\mathbf{M}_{t_0\rightarrow t_{\mathrm{v}}}$, with the $\mathbf{E}$ scalar product at t_0 and the $\mathbf{F}$ scalar product at t_{F}, has the left singular vectors $s_k(t_0)$ and the right singular vectors $s_k(t_{\mathrm{F}})$. The singular values σ_k are the error amplification factors along each direction s_k.

A special set of energy norms is provided by the choice $\mathbf{F}=\mathbf{\Pi}=\mathbf{I}$ and $\mathbf{E}=\mathbf{A}_0^{-1}$. In this case the resulting singular vectors $s_k(t_0)$ evolve into the leading eigenvectors $s_k(t_{\mathrm{v}})$ of the forecast error covariance matrix $\mathbf{P}^{f}(t_{\mathrm{v}})$. The eigenvectors $s_k(t_{\mathrm{F}})$ are called "Hessian singular vectors" (HSVs) associated with the cost functional $\mathcal{J}$ [21]. Since the leading eigenvectors of $\mathbf{P}^{f}(t_{\mathrm{v}})$ are the directions of maximum variance of forecast error, HSVs define the directions along which we must do a good job of analysis in order to minimize the forecast error at t_{v}. We assume that the model error is negligible over the period $[t_0,t_{\mathrm{v}}]$. From equation (13) it follows that Hessian singular vectors are the solutions of the following generalized eigenvalue problem

$$\mathbf{M}^{*}_{t_{\mathrm{v}}\rightarrow t_0}\,\mathbf{M}_{t_0\rightarrow t_{\mathrm{v}}}\,s_k(t_0)=\sigma_k^2\,\mathbf{A}_0^{-1}s_k(t_0)=\sigma_k^2\,\left(\nabla^2_{\mathbf{x}_0,\mathbf{x}_0}\mathcal{J}\right)\,s_k(t_0)\,. \tag{17}$$

The second relation comes from the fact that the inverse of the analysis covariance matrix is equal to the Hessian matrix of the analysis cost function $\mathcal{J}$ in the variational analysis system. This motivates the name *Hessian singular vectors* for the solutions $s_k(t_0)$ of the generalized eigenproblem (17).

The eigenvalue problem (17) is solved in the time interval $[t_0,t_{\mathrm{v}}]$ for each target using an iterative method that require only matrix–vector products [20, 22]. In [23] we have computed the total energy singular vectors for CTMs. To the best of our knowledge this is the first work that attempts the computation of Hessian singular vectors for CTMs. In [24] we have used Hessian singular vectors for the purpose of observation targeting.

The singular vectors associated with the largest m singular values,

$$\mathcal{V}_m(t_0)=\left[s_1(t_0),\cdots,s_m(t_0)\right]\,, \tag{18}$$

span a subspace of the state space which will have the maximum influence on the verification area at the verification time.

4. Dynamic Configuration of the Sensor network

We now determine those locations where perturbations have the largest energy impact over the verification area. For this, consider an initial perturbation vector $\delta_{\mathfrak{k}}$ equal to zero everywhere, except for one component at a given location where its value is 1. The index $\mathfrak{k}$ spans all variables and all grid points in the model, and represents one independent observation of one variable. The vector can be written in terms of the singular vectors

$$\delta_{\mathfrak{k}}=\sum_m\alpha_m\,s_m\,(t_0)\;,$$

where the expansion coefficients can be obtained by the orthogonality relations of the singular vectors

$$\alpha_m=\left\langle\,\delta_{\mathfrak{k}}\,,\mathbf{E}\,s_m\,(t_0)\right\rangle=\left(\mathbf{E}\,s_m(t_0)\right)_{\mathfrak{k}}\;.$$

The vector of perturbations at the final time is

$$\delta\mathbf{x}(t_{\mathrm{F}}) = \sum_m \alpha_m \, \mathbf{M}_{t_0 \to t_{\mathrm{v}}} \, s_m \, (t_0) = \sum_m \alpha_m \, \sigma_m \, s_m(t_{\mathrm{F}}) \, .$$

Using the orthogonality of the singular vectors at the final time in the $\mathbf{F}$-norm we have that the total perturbation energy is

$$E_{\mathfrak{k}} = \left\langle \, \mathbf{\Pi}\delta\mathbf{x}(t_{\mathrm{F}}) \, , \, \mathbf{F}\mathbf{\Pi}\delta\mathbf{x}(t_{\mathrm{F}}) \, \right\rangle = \sum_m \sigma_m^2 \alpha_m^2 = \sum_m \sigma_m^2 \left((\mathbf{E} \, s_m(t_0))_{\mathfrak{k}} \right)^2 \, .$$

A vector which has each component equal to the energy impact of the corresponding delta initial perturbation is therefore:

$$E = \sum_m \sigma_m^2 \, (\mathbf{E} \, s_m(t_0))^2 = \sum_m \zeta_m^2 \, . \tag{19}$$

The squares of the vectors are considered in an element by element sense. This sum can be approximated by the first several terms which correspond to the dominant singular values.

The best observations should be located at those points where the energetic impact over the verification area is the largest. These points are easily identified as they are the largest entries of the E vector.

5. Numerical Experiments

5.1. The shallow water equations

We apply this computational methodology to the shallow-water equations (SWE), which approximate fluid flow inside a shallow basin. Consider the following two-dimensional PDE system:

$$\begin{aligned}
\frac{\partial}{\partial t}h + \frac{\partial}{\partial x}(uh) + \frac{\partial}{\partial y}(vh) &= 0 \\
\frac{\partial}{\partial t}(uh) + \frac{\partial}{\partial x}(u^2h + \frac{1}{2}gh^2) + \frac{\partial}{\partial y}(uvh) &= 0 \\
\frac{\partial}{\partial t}(vh) + \frac{\partial}{\partial x}(uvh) + \frac{\partial}{\partial y}(v^2h + \frac{1}{2}gh^2) &= 0 \, .
\end{aligned}$$

A numerical model was built to compute the solution of these equations. The space discretization was performed using a finite volume-type scheme and the time discretization using fourth-order Runge-Kutta. This method was introduced by Liska and Wendroff in [25].

The spatial domain is square shaped ($\Omega = [-3, 3]^2$), and the integration window is $t^0 = 0 \leq t \leq t^F = 0.1$. Here $h(t, x, y)$ denotes the fluid layer thickness, and $u(t, x, y)$, $v(t, x, y)$ are the components of the velocity field. For a square grid of size n^2, the model has $3 \cdot n^2$ variables. g is the standard value of the gravitational acceleration. The boundary conditions are periodic in both directions. For more advanced applications, such as nonlinear optimization and sensitivity analysis, we need to compute derivatives of the model states and parameters. This can be accomplished using the methodology of adjoint models [26, 27]. A distinction is made between continuous adjoints, obtained by linearizing the differential equations, and discrete adjoints, obtained by linearizing the numerical method. The resulting adjoint equations are then solved numerically through time integration.

We build the adjoint models associated with our SWE model through automatic differentiation [28] using the TAMC tool [29]. The tangent-linear model (TLM) propagates perturbations forward in time. The first-order adjoint model (FOA) propagates perturbations backwards in time and can be used to compute the gradients of a scalar cost function defined on the model trajectory. Second-order adjoint models (SOA) compute the product between the Hessian of f and a vector. Second-order adjoint models are considered to be the best approach to compute Hessian-vector products, but have yet to become popular in practice because of their computational demands. When one does not have access to the second-order adjoint, Hessian-vector products can be computed through various approximations, such as finite difference with gradients or the Gauss-Newton approximation.

The overhead of running adjoint models is a crucial aspect of the computational strategy. For our particular implementation, one full SOA integration is about 3.5 times more expensive than a single first-order adjoint run,

while the FOA takes 3.7 times longer than the forward run. However, this ratio depends on the numerical methods used to solve the differential equations or the automatic differentiation tool employed. For certain numerical methods used to solve the forward model, it is possible to develop smart strategies of reusing computation which lead to adjoint models that take less time to run than the forward model. An example can be found in [30] where the adjoint equations of this very SWE system are derived by hand and then implemented as numerical models.

5.2. Data assimilation scenario

We apply our computational framework to two data assimilation scenarios with SWE. These two scenarios have very similar setting, the only difference being the amount of observations we assimilate. The 4D-Var parameters are as follows:

- The resolution of the 2D computational grid on which the models operate is 40 grid points on both directions (1600 in total). Thus, the model has 4800 variables.
- The size of the timestep is set at $1e-3$ and the models are configured to run for $1-e1$ seconds ($N = 100$ discrete steps of time).
- The reference solution is synthesized for h as a Gaussian pulse of amplitude $A = 30$ centered on the grid. The u/v fields are made consistent with h with the help of the forecast model.
- The background solution x^b is generated by applying a correlated perturbation on the reference solution for h, u and v.
- The background error covariance B was generated for a standard deviation of 5% with a nondiagonal structure and correlation distance of 5 grid points. This will help the 4D-Var method to spread the information for each grid point to its neighbors.
- The model is ran starting from the reference solution in order to generate the synthetic observations. The observation frequency is set to once every 20 time steps. In order to simulate the effect of noise over observations, we apply a normal random perturbation to the perfect synthetic observations.
- The observation error covariance R is a diagonal matrix, based on the assumption that the observational errors are uncorrelated. The standard deviation of these errors was set to 1% of the largest absolute value of the observations for each variable.
- The observation operator H is configured to select observations for each variable at each point of the grid (observations are dense in space). In a realistic setting, the operator H is enforced by the structure of the observational network.

Difference between the two experiments

1. REG100. Observations are available for h, u and v at every other fourth grid point on horizontal and vertical. This leads to 100 observations for each variable and 300 in total, as shown in figure 2(a).
2. SEL10. Observations are available for h, u and v at 10 selected locations, as shown in figure 2(b).

In Figure 1 we present 6 snapshots of the model trajectory, as started from the background solution and taken at every 20 time steps. These correspond to the assimilation times.

To minimize the 4D-Var cost function, we use the L-BFGS-B solver [31]. Since the SWE model does not represent a significant computational burden, we allowed the solver to run for 500 iterations, or until the gradient of the 4D-Var cost function was reduced from a magnitude of $1e+2$ to $1e-4$.

5.3. Numerical results

The minimizer of the 4D-Var cost function as provided by L-BFGS-B represents an improved estimate of the initial state of the model assimilating the available observations. The next step is to compute the singular vectors as described in Section 3. This requires the solution of a generalized eigenvalue problem. The main computational constraint consists in circumventing the fact that we do not have access to the full Jacobian (and its transpose) of the model states or to the 4D-Var Hessian. Instead, we can compute their action on a vector by using adjoint models. Jacobian-vector products can be computed by running the TLM initialized with said vector, and the transpose-Jacobian-vector products by running the FOA backwards in time, initialized with the seed vector. 4D-Var Hessian vector-products can be obtained using the second-order adjoint model.

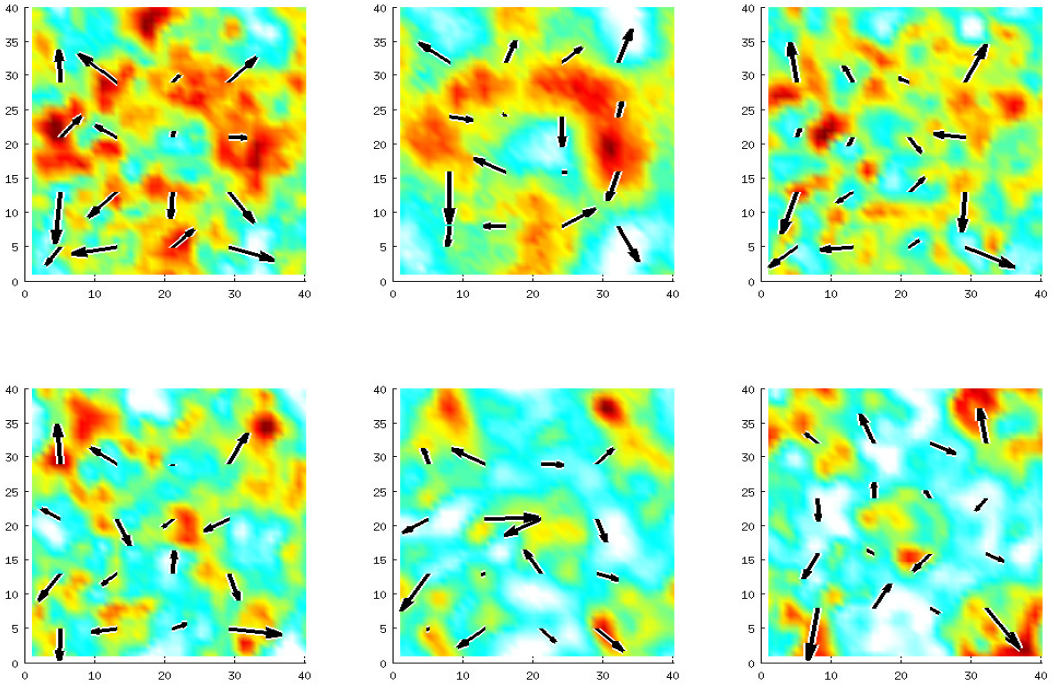
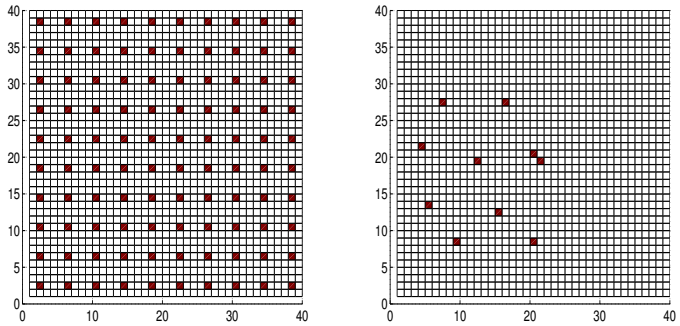
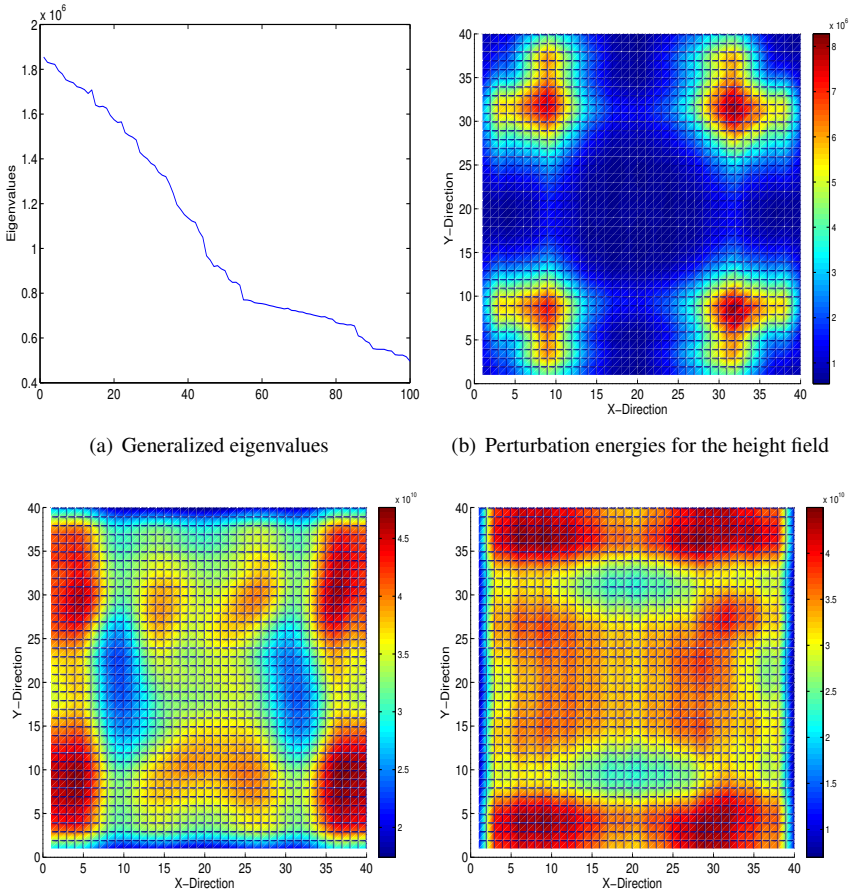


Fig. 1. The evolution of the height of the flow with time. Superimpoposed are the flow velocity vectors. Six different time instances (corresponding to 0, 20, 40, 60, 80, and 100 model time steps) are shown.



(a) Sensor Locations for the REG100 case (b) Sensor Locations for the SEL10 case

Fig. 2. Sensor locations for two different scenarios.



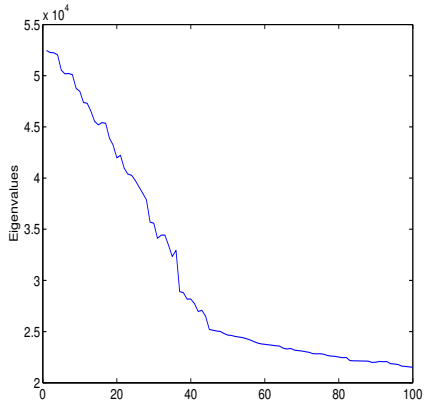
(a) Generalized eigenvalues

(b) Perturbation energies for the height field

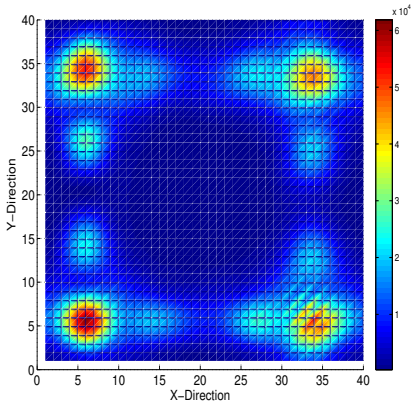
(c) Perturbation energies for the U velocity field component

(d) Perturbation energies for the V velocity field component

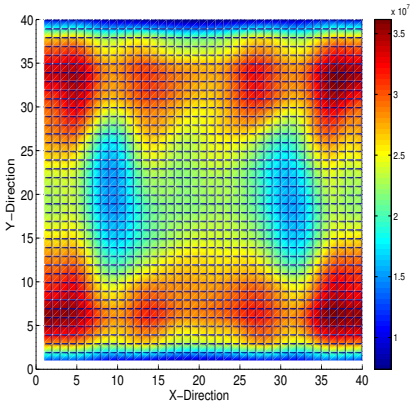
Fig. 3. Perturbation energies for the variables in the REG100 case. Higher energies correspond to better virtual locations of the sensors.



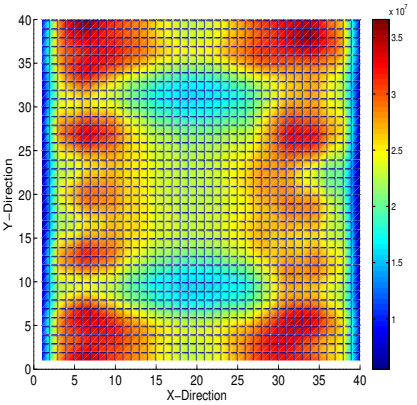
(a) Generalized Eigenvalues



(b) Perturbation energies for the height field



(c) Perturbation energies for the U velocity field component



(d) Perturbation energies for the V velocity field component

Fig. 4. Perturbation energies for the variables in the SEL10 case. Higher energies correspond to better virtual locations of the sensors.

Generalized eigenvalue problems can be solved iteratively when we only have access to matrix-vector products. Classic algorithms such as Arnoldi [32] or Lanczos [33] compute the eigenvectors associated with the m largest or smallest eigenvalues, where m corresponds to the number of matrix-vector products to be evaluated and is specified by the user. For our scenario, we compute 100 leading eigenvectors and then compute the perturbation energy associated with them following equation (19).

We use an iterative algorithm to compute the eigenvectors associated with the largest 100 eigenvalues for our two test scenarios, whose values are plotted in figures 3(a) and 4(a). We notice there is a significant cutoff after the first 50 eigenvalues so we keep only these eigenvectors for computing the energy norm E.

Figures 3 and 4 show the plot of the perturbation energy of h (3(b) and 4(b)), u (3(c) and 4(c)) and v (3(d) and 4(d)) on the computational grid, for REG100 and SEL10, respectively. We notice similar features across both cases. The profile of the perturbation energy for h, the height of the flow, reveals areas of high uncertainty in the four corners of the computational grid. Meanwhile, the perturbation energy of the wind vector components u and v is considerably larger on the opposite edges of the grid which correspond to the direction represented by the vector component: East-West for u and North-South for v. For the u and v variables there can also be noticed visible features in the middle of the grid that are similar between the two cases. This is to be expected, since both cases operate on the same forecast scenario.

The magnitude of the perturbation energies reflects the fact that one of the scenario (REG100) assimilates more information than the other (SEL10). For the former case, the perturbation energies have values of order 10^4 which can be associated with a reanalysis of smaller uncertainty than for other case, where magnitudes are of order 10^6. Also, the areas of high uncertainty in h for the REG100 case are smaller than their equivalents for the SEL10 case. This is a clear indication that the forecast started from the reanalysis obtained from assimilating more information (REG100) is less likely to be affected by perturbations situated along the principal directions of error growth. These directions of error growth constitute the left-hand side of the generalized eigenvalue problem and are the same for any data assimilation scenario. The difference between various scenarios is found on the right-hand side in the matrix norm, the 4D-Var Hessian, which gives us a meaningful measure for the perturbation energy that reflects the particular data assimilation setting used. The 4D-Var Hessian approximates the inverse of the covariance matrix of the errors in the reanalysis, so it can be interpreted as a measure of trust in the reanalysis.

In order to improve the quality of the reanalysis forecast, we need to target for observation (and subsequently, assimilation) those areas where the perturbation energy is large. This means we have to install additional sensors in the corners of the grid and on the borders.

6. Conclusions

This paper describes a methodology for dynamically configuring sensor network in dynamic data driven applications. The methodology uses Hessian singular vectors computed in the context of 4D-Var data assimilation. At the end of each assimilation window the areas of maximum energy impact are estimated; new sensors placed in those areas will provide a maximal reduction of the L^2 norm of the assimilation error.

The singular vectors of the tangent linear model are a powerful tool for analyzing the directions of maximum error growth in the forecast. When the forecast error is measured in the norm given by the 4D-Var Hessian at the reanalysis, these singular vectors are named "Hessian singular vectors". They are tailored to the particular data assimilation process through the Hessian matrix and reveal the areas of high uncertainty for the reanalyzed forecast under the constraint given by the 4D-Var setting, including available observations.

Computing Hessian singular vectors has the advantage of being feasible in practice. They do not need the full expression of the 4D-Var Hessian or the model tangent linear and adjoint operators. Instead, they can be computed through algorithms that work with matrix-vector products, which can be obtained by running tangent linear, and first and second order adjoint models. Also, since one only needs to compute the leading singular vectors, they can be computed within a budget of model runs suitable for large real-time operations. When a higher accuracy is required, more model runs can be allocated.

We perform experiments with a data assimilation system for the shallow water equations, in two settings differentiated by the number and location of available observations. The dominant 50 dominant singular vectors are used to approximate the perturbation energy. When assimilating more observations, the perturbation energy is

smaller, indicating a lower uncertainty. The profile of the perturbation energy reveals that additional observations in areas situated on the borders of the computational grid would improve the quality of the analysis and the forecast accuracy.

Acknowledgements

This work was supported by AFOSR DDDAS program through the award FA9550–12–1–0293–DEF, managed by Dr. Frederica Darema.

References

[1] R. Daley., Atmospheric data analysis, Cambridge University Press, Cambridge, 1991.
[2] E. Kalnay., Atmospheric modeling, data assimilation and predictability, Cambridge University Press, Cambridge, 2002.
[3] E. Lorenz, A study of the predictability of a 28 variable atmospheric model, Tellus 17 (1965) 321–333.
[4] F. Molteni, T. Palmer, Predictability and finite-time instability of the northern winter circulation, Quarterly Journal of the Royal Meteorological Society 119 (1993) 269–298.
[5] R. Mureau, F. Molteni, T. Palmer, Ensemble prediction using dynamically-conditioned perturbations, Quarterly Journal of the Royal Meteorological Society 119 (1993) 299–323.
[6] Z. Li, I. Navon, M. Hussaini, Analysis of the singular vectors of the full-physics fsu global spectral model, Tellus accepted.
[7] R. Buizza, A. Montani, Targeting observations using singular vectors, Journal of the Atmospheric Sciences 56 (1999) 2965–2985.
[8] J. Hansen, A. Smith, The role of operational constraints in selecting supplementary observations, Journal of the Atmospheric Sciences 57 (2000) 2859–2871.
[9] S. Majumdar, C. Bishop, R. Buizza, R. Gelaro, A comparison of ensemble transform Kalman filter targeting guidance with ecmwf and nrl total-energy singular vector guidance, Quarterly Journal of the Royal Meteorological Society 128 (585) (2002) 2527–2549.
[10] T. Palmer, R. Gelaro, J. Barkmeijer, R. Buizza, Singular vectors, metrics, and adaptive observations, Journal of the Atmospheric Sciences 55 (4) (1998) 633–653.
[11] D. Daescu, G. Carmichael, An adjoint sensitivity method for the adaptive location of the observations in air quality modeling, Journal of the Atmospheric Sciences 60 (2) (2003) 434–450.
[12] D. Daescu, I. Navon, Adaptive observations in the context of 4D-Var data assimilation, Meteorology and Atmospheric Physics 84 (4) (2004) 205–226.
[13] R. Gelaro, R. Buizza, T. N. Palmer, E. Klinker, Sensitivity analysis of forecast errors and the construction of optimal perturbations using singular vectors, Journal of the Atmospheric Sciences 55 (6) (1998) 1012–1037.
[14] E. Lorenz, K. Emanuel, Optimal sites for supplementary observations: simulation with a small model, Journal of the Atmospheric Sciences 55 (1998) 399–414.
[15] T. P. M. Leutbecher, J. Barkmeijer, A. Thorpe, Potential improvement of forecasts of two severe storms using targeted observations, Quarterly Journal of the Royal Meteorological Society 128 (583) (2002) 1641–1670.
[16] G. Carmichael, T. Chai, A. Sandu, E. Constantinescu, D. Daescu, Predicting air quality: improvements through advanced methods to integrate models and measurements, Journal of Computational Physics 227 (7) (2008) 3540–3571.
[17] A. Sandu, L. Zhang, Discrete second order adjoints in atmospheric chemical transport modeling, Jurnal of Computational Physics 227 (12) (2008) 5949–5983. doi:10.1016/j.jcp.2008.02.011.
[18] A. Cioaca, M. Alexe, A. Sandu, Second order adjoints for solving PDE-constrained optimization problems, Optimization Methods and Software 27 (4–5) (2012) 625–653.
[19] F. LeDimet, V. Shutyaev, J. Gejadze, Analysis error via Hessian in variational data assimilation, in: ARIMA Journal, CARI06, Cotonou, Benin, 2006.
[20] R. Lehoucq, K. Maschhoff, D. Sorensen, C. Yang, ARPACK software (parallel and serial), `http://www.caam.rice.edu/software/ARPACK` [cited 2012].
URL `http://www.caam.rice.edu/software/ARPACK`
[21] J. Barkmeijer, M. van Gijzen, F. Bouttier, Singular vectors and estimates of the analysis error covariance metric, Quarterly Journal of the Royal Meteorological Society 124 (549) (1998) 1695–1713.
[22] K. Maschhoff, D. Sorensen, Parallel ARPACK home page, `http://www.caam.rice.edu/~kristyn/parpack_home.html`.
URL `http://www.caam.rice.edu/~kristyn/parpack_home.html`
[23] W. Liao, A. Sandu, T. Chai, G. Carmichael, Total energy singular vector analysis for atmospheric chemical transport models, Monthly Weather Review 134 (9) (2006) 2443–2465.
[24] A. Sandu, Targeted observations for atmospheric chemistry and transport models, in: V. A. et al. (Ed.), International Conference on Computational Science (ICCS 2006), Vol. 3991 of Lecture Notes in Computer Science, 2006, pp. 712–719.
[25] R. Liska, B. Wendroff., Composite schemes for conservation laws, SIAM J. Numer. Anal. 35 (6) (1998) 2250–2271.
[26] F. L. Z. Wang, I.M. Navon, X. Zou., The second order adjoint analysis: theory and applications, Met. and Atm. Phys. 50 (1-3) (1992) 3–20.
[27] A. Sandu, L. Zhang., Discrete second order adjoints in atmospheric chemical transport modeling, J. Comput. Phys. 227 (12) (2008) 5949–5983.
[28] A. Griewank, et al., On automatic differentiation, Mathematical Programming: recent developments and applications 6 (1989) 83–107.
[29] R. Giering, T. Kaminski., Recipes for adjoint code construction, ACM Trans. Math. Software 24 (4) (1998) 437–474.

[30] A. S. A. Cioaca, M. Alexe., Second-order adjoints for solving PDE-constrained optimization problems, Optim. Meth. Softw. 27 (4-5) (2011) 625–653.
[31] C. Zhu, R. Byrd, P. Lu, J. Nocedal, L-BFGS-B: a limited memory fortran code for solving bound constrained optimization problems, Dept. of Electrical Engineering and Computer Science, Northwestern Univ., TR NAM-11, Evanston, IL.
[32] W. Arnoldi, The principle of minimized iterations in the solution of the matrix eigenvalue problem, Quart. Appl. Math 9 (1) (1951) 17–29.
[33] C. Lanczos, An iteration method for the solution of the eigenvalue problem of linear differential and integral operators, United States Governm. Press Office, 1950.