DDDAS Predictions for Water Spills

Craig C. Douglas[1,4], Paul Dostert[2], Yalchin Efendiev[3], Richard E. Ewing[3],
Deng Li[1], and Robert A. Lodder[1]

[1] University of Kentucky, Lexington, KY 40506-0046
[2] University of Arizona, Tucson, AZ 85721-0089
[3] Texas A & M University, College Station, TX 77843-3368
[4] Yale University, New Haven, CT 06520-8285, USA
douglas-craig@cs.yale.edu

Abstract. Time based observations are the linchpin of improving predictions in any dynamic data driven application systems. Our predictions are based on solutions to differential equation models with unknown initial conditions and source terms. In this paper we want to simulate a waste spill by a water body, such as near an aquifer or in a river or bay. We employ sensors that can determine the contaminant spill location, where it is at a given time, and where it will go. We estimate initial conditions and source terms using better and new techniques, which improves predictions for a variety of data-driven models.

1 Introduction

In this paper, our goal is to predict contaminant transport, where the contamination is, where the contaminant is going to go, and to monitor the environmental impact of the spill for contaminants in near coastal areas. Sensors measure the contaminant concentration at certain locations. Here, we discuss the quality of the predictions when the initial conditions and the source terms are updated as data is injected.

From a modeling viewpoint, one of the objectives in dynamic data driven application systems (DDDAS) is to improve the predictions as new data is obtained [1]. Data represents the information at different length scales and the precision can vary enormously. Numerous issues are involved in DDDAS, many of which are described in [2,3,4,5,6].

We investigate contaminant transport driven by convection and diffusion. When new sensor based data is obtained, the initial conditions and source terms are updated. The initial conditions are constructed in a finite dimensional space. Measurements often contain errors and uncertainties. We have studied elsewhere approaches where these uncertainties can be taken into account.

The first set of measurements allows recovery of an approximation to the initial conditions. An objective function is used to update initial data in the simulation. The mesh of sensor locations is quite coarse in comparison to the solution's mesh, leading to an ill-posed problem. Prior information about the initial data lets us regularize the problem and then update the initial conditions. We use a penalty

M. Bubak et al. (Eds.): ICCS 2008, Part III, LNCS 5103, pp. 54–63, 2008.

method whose constants depend on time and can be associated with the relative errors in the sensor measurements.

We also update the source terms, which is quite important when there is still a source of the contamination present in the ongoing measurements. We construct an objective functional that lets us perform an update on the source term in a finite dimensional space.

A number of numerical examples are presented that show that the update process is quite important. Using the correct choice of penalty terms demonstratively improves the predictions.

2 Contaminant Concentration Model

Let C be the concentration of a contaminant and assume that the velocity v, obtained from shallow water equations, is known. We assume that the sensors measure the concentration at some known locations. Note that the velocity field and the diffusion coefficients are unknown and need to be estimated in the most general case.

Contaminant transport is modeled using the convection-diffusion equation,

$$\frac{\partial C}{\partial t} + v \cdot \nabla C - \nabla \cdot (D \nabla C) = S(x,t) \text{ in } \Omega.$$

Given measured data, we estimate the initial condition $C(x,0) = C^0(x)$ and the source term $S(x,t)$.

3 Reconstructing Initial Conditions

We begin by obtaining initial data based on sensor readings and forward simulations assuming both a rectangular subdomain $\Omega_c \subset \Omega$ and that our initial condition is in a finite dimensional function space whose support is contained in Ω_c. This finite dimensional space on Ω_c is equipped with a set of linearly independent functions $\{\tilde{C}_i^0(x)\}_{i=1}^{N_c}$. For some $\alpha = (\alpha_1, \alpha_2, \cdots, \alpha_{N_c})$ our initial condition in this space is then represented by

$$\tilde{C}^0(x) = \sum_{i=1}^{N_c} \alpha_i \tilde{C}_i^0(x).$$

We can assume that $\tilde{C}_i(x,t)$ is the solution of our equation using the initial condition $\tilde{C}_i^0(x)$ leading to a solution of the form,

$$\tilde{C}(x,t) = \sum_{i=1}^{N_c} \alpha_i \tilde{C}_i(x,t).$$

Running the forward simulation code for each initial condition $\tilde{C}_i^0(x)$ lets us find $\tilde{C}_i(x,t)$. Let N_c be the number of initial conditions, N_s the number of output concentration locations, and N_t the number of time steps where concentrations are sampled. We save values $\left\{\tilde{C}_i(x_j,t_k)\right\}_{i=1,j=1,k=1}^{N_c,N_s,N_t}$. The problem is underdetermined for a single set of sensor measurments since $N_s < N_c$ normally. However, $N_s N_t >> N_c$ since data from these sensor locations are collected many times. By using all of the given sensor values at each of the recorded times, we attempt to solve the least squares system to recover the initial conditions and improve the overall predictions made by the model.

If we devise a method so that just a simple least squares problem is solved for each new set of sensor data, then we can solve a much smaller least sqaures problem than in the case when all of the given data is used at once. Further, should not enough data be incorporated at once, solving smaller problems is faster than solving the entire problem over again. We want a model that is improved using smaller quantities of data at any given time.

Once we have collected the data from our forward simulations, the α_i need to be calculated. We want to find the best α such that $\tilde{C}(x,t) \approx C(x,t)$, so we minimize the difference between our simulated concentration and the values at the sensors. Assume there are $N_s < N_c$ sensors in Ω, which leads to minimizing the following objective function:

$$F(\alpha) = \sum_{j=1}^{N_s} \left(\sum_{i=1}^{N_c} \alpha_i \tilde{C}_i\left(x_j,t\right) - \gamma_j(t) \right)^2 + \sum_{i=1}^{N_c} \kappa_i \left(\alpha_i - \beta_i\right)^2,$$

where κ are penalty coefficients for an *a priori* vector β, which will be updated during the simulation to achieve higher accuracy. The minimization of this function gives rise to the linear system $A\alpha = R$, where for $m,n = 1,\cdots,N_c$, and

$$A_{mn} = \textstyle\sum_{j=1}^{N_s} \tilde{C}_m\left(x_j,t\right) \tilde{C}_n\left(x_j,t\right) + \delta_{mn}\kappa_m, \quad \text{and}$$
$$R_m = \textstyle\sum_{j=1}^{N_s} \tilde{C}_m\left(x_j,t\right) \gamma_j(t) + \kappa_m \beta_m.$$

This linear system is clearly symmetric, positive definite and is solved using a direct inversion since it is such a small matrix.

Since we have $\tilde{C}_i\left(x_j,t\right)$ at given time steps $\{t_k\}_{k=1}^{N_t}$ we first solve the linear system $A\alpha^1 = R$, where A and R are evaluated at time t_1 and α^1 refers to the values of α_i at t_1. Initially, we begin with a given value of β whose value can be adapted, though generally we initially choose $\beta = 0$. Once α^1 is determined, β is replaqced with α^1 and we solve the linear system again to determine α^2. We continue this process until we get α^{N_t}, which is the most accurate reconstruction of the initial data.

4 Reconstructing the Source Term

Consider the situation when the source term S is given for a particular fixed time interval at a particular location. Assume there is no initial concentration and all of the concentration in the domain comes from the source term so that

$C(x,0) = 0$ on the whole domain. Now consider a subdomain Ω_C where the source term is defined. Now assume that our region Ω is discretized with N grid points and that the subdomain Ω_C is discretized with N_C grid points. On Ω_C assume that there are basis functions $\{\delta_k\}_{k=1}^{N_C}$ which are nonzero on the k^{th} part of the subdomain and zero otherwise. Assume that $S \approx \tilde{S} = \sum_{k=1}^{N_C} \alpha_k \, \delta_k\left(x,t\right)$ where $S = \tilde{S} = \delta_k\left(x,t\right) = 0$ for some $t > \hat{t}$, i.e., the basis functions are nonzero for the same time interval as S and the source is zero after an initial time period. This is the case when there is an immediate spill that takes a short period of time for all of the contaminant to leak into the water. Using this $\tilde{S}$, our equation becomes

$$\frac{\partial \tilde{C}}{\partial t} - L\left(\tilde{C}\right) = \tilde{S} = \sum_{k=1}^{N_C} \alpha_k \, \delta_k\left(x,t\right)$$

since $\tilde{S}$ is a linear combination of δ_k. We solve

$$\frac{\partial \psi_k}{\partial t} - L\left(\psi_k\right) = \delta_k\left(x,t\right)$$

for ψ_k and each k. We denote the solution to this equation as $\{\psi_k\left(x,t\right)\}_{k=1}^{N_c}$ for each k. Under these assumptions, the solution to

$$\frac{\partial C\left(x,t\right)}{\partial t} - L\left(C\left(x,t\right)\right) = S\left(x,t\right), \, C\left(x,0\right) = 0 \ x \in \Omega$$

is approximated by

$$C\left(x,t\right) \approx \tilde{C}\left(x,t\right) = \sum_{k=1}^{N_C} \alpha_k \, \psi_k\left(x,t\right).$$

Once again, assume that there are $N_s < N_c$ sensors spread within the domain. Choose a source term $\tilde{S}$ and run the forward problem for this particular source while recording the values of the concentration at each sensor location. These values are given by $\{\gamma_j\left(t\right)\}_{j=1}^{N_s}$. Once this equation has have solved for each of the source terms, α_k can be reconstructed by solving

$$F\left(\alpha\right) = \sum_{j=1}^{N_s}\left(\sum_{k=1}^{N_c} \alpha_k \psi_k\left(x,t\right) - \gamma_k\left(t\right)\right)^2 + \sum_{k=1}^{N_c} \kappa_k\left(\alpha_k - \beta_k\right)^2,$$

where κ are penalty coefficients for an a vector β. Minimize this function and solve the corresponding linear system. Note that this is the same exact minimization that was needed for the initial condition recovery problem.

4.1 Solving the Source Term and Initial Condition Problem

We split the solution into two parts in order to predict contaminant transport in the presence of unknown initial conditions and source terms. The first part is due to unknown initial condition and the second one is due to unknown source terms.

We briefly repeat the situation when the source term S is zero, where we assumed there is an initial concentration and solved

$$\frac{\partial C}{\partial t} - L\left(C\right) = 0,\, C\left(x,0\right) = C^{0}\left(x\right).$$

Consider a discretized subdomain Ω_C where the initial condition is nonzero and assume there is a linearly independent set of functions defined by $\{\varphi_i\}_{i=1}^{N_D}$ on the subdomain given by $C^0\left(x\right) \approx \tilde{C}^0\left(x\right) = \sum_{i=1}^{N_D} \lambda_i \varphi_i^0\left(x\right)$. Now solve

$$\frac{\partial \varphi_i}{\partial t} - L\left(\varphi_i\right) = 0,\, \varphi_i\left(x,0\right) = \varphi_i^0\left(x\right)$$

for each i. The solution to this equation for each i is given by $\varphi_i\left(x,t\right)$. We approximate the solution of $\frac{\partial C}{\partial t} - L\left(C\right) = 0,\, C\left(x,0\right) = C^0\left(x\right)$ by $\tilde{C}\left(x,t\right) = \sum_{i=1}^{N_D} \lambda_i \varphi_i\left(x,t\right)$.

For the second step, we solve the problem for ψ and each k with an unknown source term,

$$\frac{\partial \psi}{\partial t} - L\left(\psi\right) = \delta_k\left(x\right),\, \psi\left(x,0\right) = 0$$

and denote the solution for eazch k as $\{\psi_k\left(x,t\right)\}_{k=1}^{N_c}$. Hence, the solution to our original problem with both the source term and initial condition is given by

$$\tilde{C}\left(x,t\right) = \sum_{i=1}^{N_D} \lambda_i \varphi_i\left(x,t\right) + \sum_{k=1}^{N_c} \alpha_k \psi_k\left(x,t\right).$$

We need to verify that this is really the solution. Compute

$$L\left(\tilde{C}\right) = \sum_{i=1}^{N_D} \lambda_i L\left(\varphi_i\left(x,t\right)\right) + \sum_{k=1}^{N_c} \frac{\partial}{\partial t} \alpha_k L\left(\psi_k\left(x,t\right)\right) \text{ and}$$

$$\frac{\partial}{\partial t}\tilde{C}\left(x,t\right) = \sum_{i=1}^{N_D} \lambda_i \frac{\partial}{\partial t}\varphi_i\left(x,t\right) + \sum_{k=1}^{N_c} \frac{\partial}{\partial t}\alpha_k \psi_k\left(x,t\right).$$

So

$$\begin{aligned}\frac{\partial \tilde{C}}{\partial t} - L\left(\tilde{C}\right) &= \sum_{i=1}^{N_D} \lambda_i \left[\frac{\partial}{\partial t}\varphi_i\left(x,t\right) - L\left(\varphi_i\left(x,t\right)\right)\right] + \\ &\quad \sum_{k=1}^{N_c} \frac{\partial}{\partial t}\alpha_k\left[\psi_k\left(x,t\right) - L\left(\psi_k\left(x,t\right)\right)\right] \\ &= \sum_{k=1}^{N_c} \frac{\partial}{\partial t}\alpha_k\left[\psi_k\left(x,t\right) - L\left(\psi_k\left(x,t\right)\right)\right] = \sum_{k=1}^{N_c} \alpha_k \delta_k\left(x\right).\end{aligned}$$

Similarly, $\tilde{C}(x,0) = \sum_{i=1}^{N_D} \lambda_i \varphi_i(x,0) = \tilde{C}^0(x)$. Hence, we have verified that

$$\tilde{C}(x,t) = \sum_{i=1}^{N_D} \lambda_i \varphi_i(x,t) + \sum_{k=1}^{N_c} \alpha_k \psi_k(x,t)$$

really solves our original equation with both an initial condition and source term.

4.2 Reconstruction of Initial Condition and Source Term

After running the forward simulation for each initial basis function and source basis function, we minimize

$$\begin{aligned} F(\alpha,\lambda) = \sum_{j=1}^{N_s} \left[\left(\sum_{k=1}^{N_c} \alpha_k \psi_k(x_j,t) + \sum_{k=1}^{N_D} \lambda_k \varphi_k(x_j,t) - \gamma_j(t) \right)^2 \right] + \\ \sum_{k=1}^{N_c} \tilde{\kappa}_k \left(\alpha_k - \tilde{\beta}_k \right)^2 + \sum_{k=1}^{N_D} \hat{\kappa}_k \left(\lambda_k - \hat{\beta}_k \right)^2 . \end{aligned} \tag{1}$$

For $N = N_c + N_d$, $\mu = [\alpha_1, \cdots, \alpha_{N_c}, \lambda_1, \cdots, \lambda_{N_D}]$, $\eta(x,t) = [\psi_1, \cdots, \psi_{N_c}, \varphi_1, \cdots, \varphi_{N_D}]$, $\beta = \left[\tilde{\beta}_1, \cdots, \tilde{\beta}_{N_c}, \hat{\beta}_1, \cdots, \hat{\beta}_{N_D}\right]$, $\kappa = [\tilde{\kappa}_1, \cdots, \tilde{\kappa}_{N_c}, \hat{\kappa}_1, \cdots, \hat{\kappa}_{N_D}]$, we minimize

$$F(\mu) = \sum_{j=1}^{N_s} \left[\left(\sum_{k=1}^{N} \mu_k \eta_k(x_j,t) - \gamma_j(t) \right)^2 \right] + \sum_{k=1}^{N} \kappa_k (\mu_k - \beta_k)^2 . \tag{2}$$

This is the same minimization that we had previously, which leads to solving a least squares problem of the form $A\mu = R$, where

$$A_{mn} = \sum_{j=1}^{N} \eta_m(x_j,t)\, \eta_n(x_j,t) + \delta_{mn}\kappa_m \text{ and } R_m = \sum_{j=1}^{N} \eta_m(x_j,t)\, \gamma_j(t) + \kappa_m \beta_m .$$

Sensor values are recorded only at discrete time steps $t = \{t_j\}_{j=1}^{N_t}$. μ is first estimated using the sensor values at $t = t_1$. Then each successive set of sensor values is used to improve the estimate of μ.

5 Numerical Results

We have performed extensive numerical studies for initial condition and source term estimation [7]. The numerical results convincingly demonstrate that the predictions can be improved by updating initial conditions and source terms.

Each problem has commonality:

- An initial condition is defined on a domain of $[0, 1] \times [0, 1]$.
- Sensor data is recorded at given sensor locations and times and is used to reconstruct the initial condition.
- Biquadratic finite elements are used in both the forward simulation and the reconstruction.
- For our initial condition expansion

  $$C^n(x) = \sum_{i=1}^{N} c_i^n \varphi_i(x).$$

  We assume φ_i are either piecewise constants or bilinears defined on a given subdomain with its own grid (i.e., there are two different grids): a large (fine) grid where the forward simulation is run and a small (coarse) grid defined only on a given subdomain where we are attempting a reconstruction.
- All velocities are $[2, 2]$ on each cell. Thus our flow is from the lower left corner to the upper right corner of the 2D grid for each problem.
- We sample sensor data every 0.05 seconds for 1.0 seconds at the following five locations: $(.5, .5), (.25, .25), (.25, .75), (.75, .25)$, and $(.75, .75)$.

5.1 Reconstruction Using Piecewise Constants

We attempt to reconstruct a constant initial condition with support on $[0, .2] \times [0, .2]$. We choose an underlying grid defined only on a subregion where we define the basis functions used in the reconstruction.

First, let this grid be exactly where there is support for the function. For example, if we have a 2×2 grid, then we define 4 piecewise constants on $[0, .2] \times [0, .2]$. Hence, support would be on the following four subdomains: $[0, .1] \times [0, .1]$, $[.1, .2] \times [0, .1]$, $[0, .1] \times [.1, .2]$, and $[.1, .2] \times [.1, .2]$. The region is divided similarly for different sized grids.

Second, let the subdomain be larger than the support of the initial condition, e.g., choose $[0, .4] \times [0, .4]$ as the subdomain. Hence, the "effective" area of the basis functions is reduced by a factor of 4 each.

Choose a 2×2 subgrid for the basis functions on $[0, .2] \times [0, .2]$ so that there are only 4 basis functions in the reconstruction. As can be seen in Fig. 1, the reconstructed initial condition are quite good: the initial condition reconstruction only needed two sets of sensor measurements.

Using the same strategy for dividing the domain up into a small number of equal parts we find slightly worse results for larger grids. This seems natural considering that we are using the same amount of information, but we are attempting to reconstruct more functions. Clearly there should be larger errors as the number of functions is increased unless we add more sensor data. Experiments show that this type of folklore is true.

Consider a case with 9 sensor locations instead of 5: $(.25, .25)$, $(.25, .5)$, $(.25, .75)$, $(.5, .25)$, $(.5, .5)$, $(.5, .75)$, $(.75, .25)$, $(.75, .5)$, and $(.75, .75)$. We use

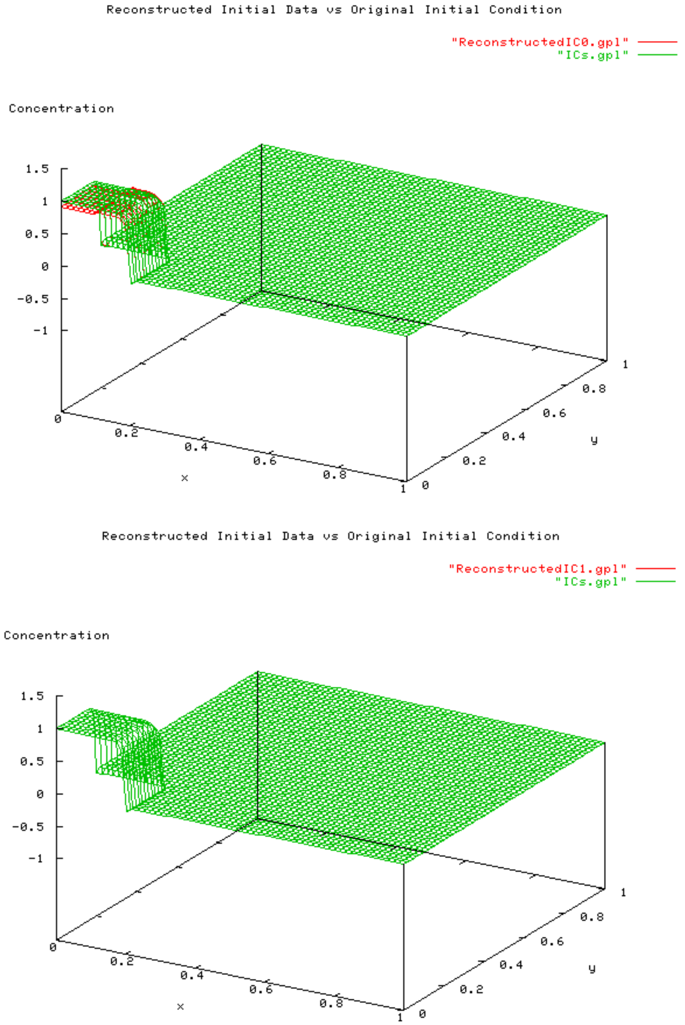

Fig. 1. Five sensors case

bilinears with a 2 × 2 grid so that there are 16 basis functions instead of 4. The accuracy is far higher than when we used the same parameters with only 5 sensors, as can be seen in Fig. 2.

Consider a case with just 2 sensor locations instead of 5: (.5, .5) and (.25, .5). The accuracy is far lower than when we used the same parameters with only 5 sensors, as can be seen in Fig. 3.

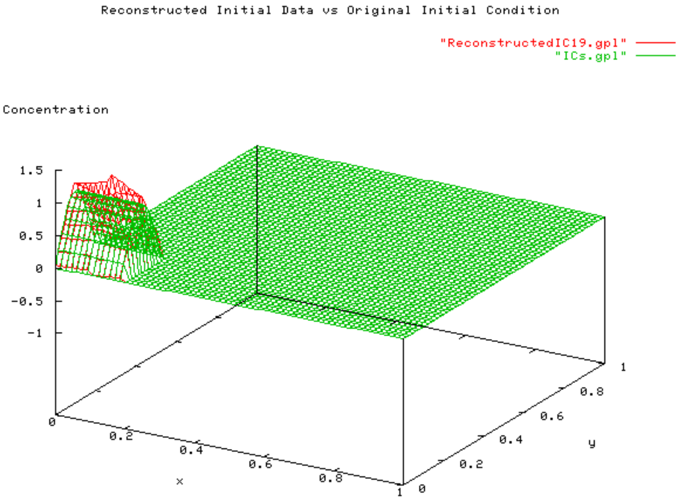

Fig. 2. Nine sensors case

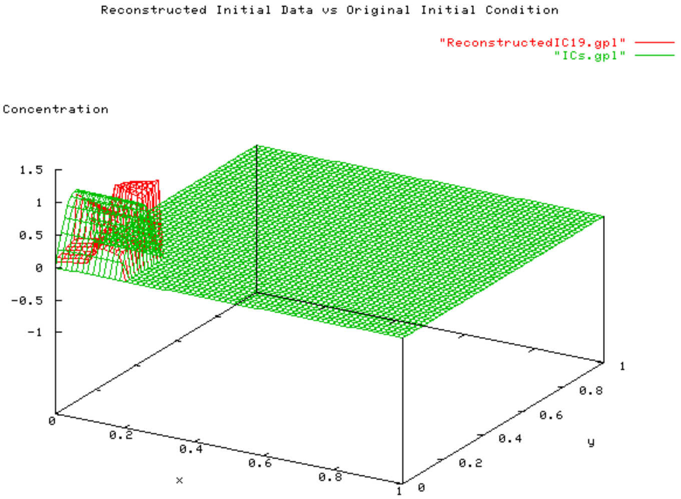

Fig. 3. Two sensors case

For future work, we need to test the proposed methods for numerous other initial conditions and source terms update conditions.

Acknowledgements. This research was supported in part by NSF grants EIA-0219627, EIA-0218229, CNS-0540178, and CNS-0540136.

References

1. Darema, F.: Introduction to the ICCS 2007 Workshop on Dynamic Data Driven Applications Systems. In: Shi, Y., van Albada, G.D., Dongarra, J., Sloot, P.M.A. (eds.) ICCS 2007. LNCS, vol. 4487, pp. 955–962. Springer, Heidelberg (2007)
2. Dostert, P.: Uncertainty Quantification Using Multiscale Methods for Porous Media Flows. PhD thesis, Texas A & M University, College Station, TX (December 2007)
3. Douglas, C., Cole, M., Dostert, P., Efendiev, Y., Ewing, R., Haase, G., Hatcher, J., Iskandarani, M., Johnson, C., Lodder, R.: Dynamic Contaminant Identification in Water. In: Alexandrov, V.N., van Albada, G.D., Sloot, P.M.A., Dongarra, J. (eds.) ICCS 2006. LNCS, vol. 3993, pp. 393–400. Springer, Heidelberg (2006)
4. Douglas, C., Cole, M., Dostert, P., Efendiev, Y., Ewing, R., Haase, G., Hatcher, J., Iskandarani, M., Johnson, C., Lodder, R.: Dynamically identifying and tracking contaminants in water bodies. In: Shi, Y., van Albada, G.D., Dongarra, J., Sloot, P.M.A. (eds.) ICCS 2007. LNCS, vol. 4487, pp. 1002–1009. Springer, Heidelberg (2007)
5. Douglas, C., Efendiev, Y., Ewing, R., Ginting, V., Lazarov, R., Cole, M., Jones, G., Johnson, C.: Multiscale interpolation, backward in time error analysis for data-driven contaminant simulation. In: Sunderam, V.S., van Albada, G.D., Sloot, P.M.A., Dongarra, J. (eds.) ICCS 2005. LNCS, vol. 3515, pp. 640–6470. Springer, Heidelberg (2005)
6. Douglas, C., Efendiev, Y., Ewing, R., Ginting, V., Lazarov, R.: Dynamic data driven simulations in stochastic environments. Computing 77, 321–332 (2006)
7. Dostert, P.: http://math.arizona.edu/~dostert/dddasweb (last visited 2/1/2008)