# Real-Time Ensemble Control with Reduced-Order Modeling

The MIT Faculty has made this article openly available. Please share how this access benefits you. Your story matters.

<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>As Published</td>
<td><a href="http://dx.doi.org/10.1137/130921878">http://dx.doi.org/10.1137/130921878</a></td>
</tr>
<tr>
<td>Publisher</td>
<td>Society for Industrial and Applied Mathematics</td>
</tr>
<tr>
<td>Version</td>
<td>Final published version</td>
</tr>
<tr>
<td>Accessed</td>
<td>Wed Nov 21 14:24:09 EST 2018</td>
</tr>
<tr>
<td>Citable Link</td>
<td><a href="http://hdl.handle.net/1721.1/92548">http://hdl.handle.net/1721.1/92548</a></td>
</tr>
<tr>
<td>Terms of Use</td>
<td>Article is made available in accordance with the publisher's policy and may be subject to US copyright law. Please refer to the publisher's site for terms of use.</td>
</tr>
<tr>
<td>Detailed Terms</td>
<td></td>
</tr>
</tbody>
</table>
REAL-TIME ENSEMBLE CONTROL WITH REDUCED-ORDER MODELING

BINGHUAI LIN† AND DENNIS MCLAUGHLIN†

Abstract. The control of spatially distributed systems is often complicated by significant uncertainty about system inputs, both time-varying exogenous inputs and time-invariant parameters. Spatial variations of uncertain parameters can be particularly problematic in geoscience applications, making it difficult to forecast the impact of proposed controls. One of the most effective ways to deal with uncertainties in control problems is to incorporate periodic measurements of the system’s states into the control process. Stochastic control provides a convenient way to do this, by integrating uncertainty, monitoring, forecasting, and control in a consistent analytical framework. This paper describes an ensemble-based approach to closed-loop stochastic control that relies on a computationally efficient reduced-order model. The use of ensembles of uncertain parameters and states makes it possible to consider a range of probabilistic performance objectives and to derive real-time controls that explicitly account for uncertainty. The process divides naturally into forecast/update and forecast/control steps carried out recursively and initialized with a prior ensemble that describes parameter uncertainty. The performance of the ensemble controller is investigated here with a numerical experiment based on a solute transport control problem. This experiment evaluates the performance of open- and closed-loop controllers with full and reduced-order models as well as the performance obtained with a controller based on perfect knowledge of the system and the nominal performance obtained with no control. The experimental results show that a closed-loop controller that relies on measurements consistently performs better than an open-loop controller that does not. They also show that a reduced-order forecasting model based on offline simulations gives nearly the same performance as a significantly more computationally demanding full-order model. Taken together, these results confirm that reduced-order ensemble closed-loop control is a flexible and efficient option for uncertain spatially distributed systems.

Key words. model order reduction, ensemble Kalman filter, real-time control, model predictive control

AMS subject classifications. 93E11, 93E20, 93B11

DOI. 10.1137/130921878

1. Introduction. Efforts to control dynamic systems can often benefit from frequent measurements of system behavior. Such measurements can help compensate for inaccurate predictions or other errors by revealing the actual system’s response to control actions. If we have a perfect model of the system’s response there is no need to rely on measurements since they do not add any new information. But in realistic situations, where models are imperfect and system response is uncertain, proper use of measurement information can greatly improve system performance. The roles of uncertainty and measurement feedback are closely related and deserve to be addressed from a single unified perspective. Stochastic control theory offers such a perspective by considering how control strategies should be designed to deal with uncertainty and to benefit from observations [4].

†Department of Civil and Environmental Engineering, Massachusetts Institute of Technology, Cambridge, MA 02139 (blin@mit.edu, dennism@mit.edu).
Here we consider an approach to stochastic control that is especially useful for real-time operation of complex spatially distributed systems with uncertain inputs. Such systems are often encountered in geoscience applications, including management of water and petroleum resources, contaminant remediation, and carbon sequestration. Our particular approach to stochastic control uses ensembles of possible inputs and states to represent uncertainty, to determine how to make best use of available measurements, and to select control strategies that are likely to achieve particular objectives.

Figure 1 illustrates one possible approach to ensemble real-time control. The true system is subject to uncertain inputs that lead to uncertainties in the system states. We suppose that controlled inputs are applied to this system at discrete times defined over a specified control horizon. Each pass around the control loop in the figure describes operations that take place between two successive control times. We also suppose that noisy measurements of some of the system states are available at the control times. Although these measurements are known up to and including the current time, the states they depend upon are uncertain.

We suppose that the system uncertainties can be described by a prior ensemble of system states composed of many replicates generated from different combinations of possible inputs. In the diagram double lines indicate ensembles while single lines indicate individual variables that may be specified or uncertain, depending on context. The ensemble of possible system states is updated with new measurements whenever they become available. At each control time the current updated ensemble

**Fig. 1. Basic components of an ensemble feedback control procedure.**
serves as an initial condition for a forecast to the next time and beyond, to the end of the control horizon. Also at each time a single deterministic control is applied to the true system. This control is chosen to minimize a cost-related measure of performance that depends on the entire forecast ensemble and explicitly accounts for uncertainty. A stochastic approach typically leads to more conservative strategies than would be obtained with a controller that assumes the system response is perfectly known.

The procedure outlined above is a “closed-loop” ensemble controller in the sense that there is feedback from the true system through the measurements and forecast to the controller. This is in contrast to an “open-loop” ensemble controller in which there is no feedback either because measurements are not collected or because they have no impact on subsequent controls. Figure 1 is reduced to open-loop ensemble control when the measurement update operation is omitted. We can expect closed-loop control to work better than open-loop control when the true system behavior is uncertain since the closed-loop controller can continually adjust as the true system responds, sometimes in unexpected ways [16].

Candidate updating techniques suitable for ensemble real-time control include various forms of the ensemble Kalman filter [9, 10] and the particle filter [11]. These techniques are appropriate for many nonlinear problems and also offer considerable flexibility for describing prior uncertainty. Here we compute the replicates needed to derive real-time controls from an unbiased square root version of the ensemble Kalman filter [14]. There are many methods for deriving the control strategy from the forecast ensemble. We use a stochastic version of model predictive control that derives the current control with a sequential quadratic programming algorithm. The gradients required by this algorithm are computed with an adjoint approach [15].

A model of the true system is used to generate forecast ensembles required by both the updating and optimization steps of the real-time control algorithm. The number of model simulations needed is on the order of the number of ensemble replicates multiplied by the number of optimization search iterations, multiplied by the number of control times. Generally speaking, larger problems require more replicates and more search iterations as well as more computation time for each model simulation. The computational effort associated with these simulations can be substantial, to the point where it discourages use of an ensemble approach or at least greatly reduces the size of the ensembles used to guide control decisions. It is difficult to properly describe uncertainty and to derive accurate updates when the ensemble size becomes too small.

In this paper we address the computational limitations of ensemble control by using a low-dimensional reduced-order model to make forecasts for both update and optimization operations. The reduced-order model is derived from a higher-dimensional full-order model with a nonlinear extension of principal orthogonal decomposition. The nonlinear reduced-order model is able to account for variable interactions that are not considered in linear model reduction approaches. It captures the key dynamical properties of the full-order model and also reproduces the probabilistic characteristics of ensembles obtained from full-order simulations.

Our ensemble control procedure is tested on a numerical experiment motivated by a practical environmental control problem: cost-effective removal of dissolved contaminants from a groundwater aquifer. The controls of interest in this problem are well pumping rates and the primary sources of uncertainty are poorly known geological properties. The experimental results clearly reveal that a closed-loop stochastic controller that relies on real-time measurements gives better performance than an
open-loop controller without measurements. They also show that closed-loop control
with our nonlinear reduced-order model gives nearly the same performance as control
with a more computationally demanding full-order model. Finally, the experiment
suggests that it is possible to obtain a robust control strategy that reduces perfor-
mance uncertainty with little or no increase in average cost.

Our discussion begins with a full-order formulation of the stochastic real-time con-
trol problem. We then consider a revised formulation that relies on an approximate
reduced-order model. This is followed by a description of the numerical experiment
and a presentation of some typical results. We conclude with a consideration of limi-
tations and possible enhancements of the approach.

2. Problem formulation and methodology.

2.1. Full-order formulation. We formulate the real-time control problem by
following Figure 1 from the true system through measurement collection and updating,
to forecasting and control, back to the true system. The spatially distributed system
to be controlled is described by a set of partial differential equations discretized over
an appropriate spatial grid. The discretization produces the following set of nonlinear
equations, which are solved simultaneously:

\[
\frac{dx}{dt} = F(x, w, \alpha, \omega, u), \quad x(t_0) = x_0 \text{ specified},
\]

\[
G(x, w, \alpha, \omega, u) = 0.
\]

The spatially discretized dynamic states (prognostic variables) are assembled in the
\(N_x\) dimensional vector \(x\). These variables are described by a set of \(N_x\) ordinary dif-
fferential equations (2.1) derived from partial differential equations with both time
and space derivatives. Another set of dynamic variables (diagnostic variables) are
assembled in the \(N_w\) dimensional vector \(w\). These variables are described by a set
of \(N_w\) algebraic equations (2.2) derived from partial differential equations with only
space derivatives. The algebraic functions \(F(x, w, \alpha, \omega, u)\) and \(G(x, w, \alpha, \omega, u)\), with
dimensions \(N_x\) and \(N_w\), respectively, are generated by the discretization procedure
and account for the effects of known boundary conditions. It is convenient for control
purposes to divide the system inputs into an \(N_\alpha\) dimensional vector of time-invariant
but spatially variable model parameters \(\alpha\), an \(N_\omega\) dimensional vector of time-varying
uncontrolled inputs \(\omega\), and an \(N_u\) dimensional vector \(u\) of time-varying control vari-
ablevars that are adjusted to achieve a specified objective. In some cases, it is useful to
include the initial condition \(x_0\) in the parameter vector.

In the applications of interest here we suppose that the parameter \(\alpha\) is the only
uncertain model input. The uncontrolled time-varying input \(\omega\) is assumed to be per-
fectly known and the control \(u\) is to be determined. The functions \(F\) and \(G\) used to
describe the system are also assumed to be perfectly known. This problem formulation
is sometimes called a perfect model scenario. In the real-time control example consid-
ered here parameter errors dominate and the perfect model assumption is a reasonable
approximation. In other applications it may be necessary to explicitly consider un-
certainties in \(\omega\). This extension is feasible within the conceptual framework presented
here, although it complicates the computations.

It is convenient to work with an \(N_y = N_x + N_\alpha\) dimensional augmented state
vector \(y\) that combines the prognostic variables and time-invariant parameters:

\[
y = \begin{bmatrix} x \\ \alpha \end{bmatrix}.
\]
The state equations can then be written in an augmented form as

\[
\begin{align*}
\frac{dy}{dt} &= F_a(y, w, u) = \begin{bmatrix} F(x, w, \alpha, \omega, u) & 0 \end{bmatrix}, \quad y(t_0) = y_0 \text{ specified}, \\
G_a(y, w, u) &= G(x, w, \alpha, \omega, u) = 0,
\end{align*}
\]

where the diagnostic vector \( w \) has dimension \( N_w \). The vector function \( F_a \) has dimension \( N_x + N_\alpha \) while \( G_a \) has dimension \( N_w \). The dependence on \( \omega \) has been omitted in the augmented functions to simplify notation. Note that (2.3) relies on the assumption that \( \alpha \) is time invariant.

We define the control horizon to be the time interval that extends from the current time \( t_k = k \Delta t \) to a specified end time \( T = N_T \Delta t \), where \( k = 0, \ldots, N_T - 1 \). For simplicity, we suppose that this horizon is divided into \( N_T - k \) control intervals of equal duration \( \Delta t \). We seek a set of controls \( u_{k:N_T-1} = [u_k, u_{k+1}, \ldots, u_{N_T-1}] \) to be applied over these control intervals, where each \( u_l \) is an \( N_u \) dimensional vector of scalar control variables. We use the colon notation (such as \( k : N_T - 1 \) in \( u_{k:N_T-1} \)) as shorthand for a sequence of variables defined at different times. The control sequence is constrained to be piecewise constant, with each control value held constant within its respective control interval.

We suppose here that the measurements for closed-loop control are obtained at the times \( t_{1: N_T-1} \) (i.e., at the beginning of each control period after the first). For simplicity, we also suppose that these measurements are linearly related to the states as follows:

\[
(2.5) 
\begin{align*}
    z_k &= M_k y_k + v_k, \quad k = 1, \ldots, N_T - 1,
\end{align*}
\]

where \( z_k \) is an \( N_y \) dimensional measurement vector, \( v_k \) is a zero-mean temporally uncorrelated \( N_y \) dimensional random noise vector with a specified covariance matrix \( R_k \), and \( M_k \) is a specified \( N_y \) by \( N_x \) dimensional measurement matrix.

A stochastic approach to real-time control accounts, at least in an approximate way, for uncertainties about the system to be controlled. In order to do this we adopt a Bayesian approach and suppose that the uncertain augmented state vector is characterized by a randomly generated prior probability distribution or, alternatively, by an ensemble of samples (or replicates) drawn from that distribution. This ensemble is updated whenever measurements become available. As mentioned in the introduction, our updated ensemble at the current time \( t_k \) is obtained from a version of the ensemble Kalman filter [14] and our forecast ensemble is derived from a simulation model based on (2.3) and (2.4). The forecast ensemble \( y^f_{k:N_T|k} \) obtained from the measurement update at \( t_k \) is indexed by replicate \( j = 1, \ldots, N_R \) and describes the system’s response over \([t_k : T]\), for a specified control sequence \( u_{k:N_T-1} \) and for \( k = 1, \ldots, N_T - 1 \). Here the vertical bars indicate conditioning on measurements (e.g., the replicate \( y^f_{k+1|k} \) is a sample from a probability distribution conditioned on measurements collected through \( t_k \)). Since there are no measurements at \( t_0 (k = 0) \), variables with a 0 after the vertical bar are unconditioned. The detailed computations required to obtain the forecasts are described below.

In the version of real-time control described here the effectiveness of a proposed control sequence is measured by the cost \( J_k \) of operating the system from \( t_k \) to \( T \) (i.e., the cost-to-go), for \( k = 0, \ldots, N_T - 1 \). Costs before the current time have already been incurred and are not considered in the selection of current and future controls.
We suppose that the cost-to-go is separable so it can be written as the sum of the costs incurred over the remaining control intervals:

\[
J_k(u_{k:N_T-1}, y_{k:N_T} | k) = \sum_{l=k}^{N_T-1} \gamma_{l|k}(y_{l|k}, y_{l+1|k}, u_l),
\]

where \(\gamma_{l|k}(y_{l|k}, y_{l+1|k}, u_l)\) is the cost incurred over the control interval \([t_l, t_{l+1}]\), conditioned on measurements collected through \(t_k\). Note that \(\gamma_{l|k}\) is uncertain, for a given control sequence, by virtue of its dependence on the uncertain conditional states \(y_{k:N_T|k}\). We can account for this uncertainty by constructing, for a given control sequence, an ensemble of cost-to-go replicates \(J^j_k(u_{k:N_T-1}, y^j_{k:N_T} | k)\) from an ensemble \(y^j_{k:N_T|k}\) of conditional states obtained, for example, from an ensemble Kalman filter:

\[
J^j_k(u_{k:N_T-1}, y^j_{k:N_T} | k) = \sum_{l=k}^{N_T-1} \gamma^j_{l|k}(y^j_{l|k}, y^j_{l+1|k}, u_l).
\]

These replicates can be used to estimate the probability density of the cost-to-go obtained for a particular control sequence.

Equation (2.7) reveals an assumption implicit in our definition of cost. In this equation the ensemble \(\gamma^j_{l|k}\) of costs incurred in control interval \([t_l, t_{l+1}]\) is conditioned only on measurements collected through \(t_k\). However, when this cost is actually incurred measurements will have been collected through \(t_l\) and the uncertainty will be less than suggested by the \(\gamma^j_{l|k}\) ensemble of (2.7). Derivation of a control based on (2.7) essentially ignores the value of information gained from updates with measurements collected in the future. Although these measurements are unknown at time \(t_k\) we do know that they will add information, as reflected in the measurement equation of (2.5). We can contrast a stochastic control approach based on (2.7) to stochastic dynamic programming, which accounts for the information obtained from all measurements, past, present, and future, when deriving a real-time decision rule [4]. Although a control algorithm based on (2.7) is suboptimal compared to stochastic dynamic programming, it is much less computationally demanding, especially for large distributed parameter problems.

With the cost-to-go ensemble specified we can consider how to select a single control \(u_k\) to apply to the true system at \(t_k\). Although the current state and the cost-to-go are random the current control must be deterministic. One option for obtaining this control is to minimize the sample mean of the cost-to-go, obtained by averaging over all replicates in the cost-to-go ensemble:

\[
V_k(u_{k:N_T-1}) = \frac{1}{N_R} \sum_{j=1}^{N_R} J^j_k(u_{k:N_T-1}, y^j_{k:N_T} | k).
\]

Minimization of the deterministic performance function \(V_k\) at time \(k\Delta t\) yields a set \(u^*_{k:N_T-1|k}\) of optimal controls for the entire control horizon.

We can now combine the minimization of (2.8) at each time with the computation of the required forecasts and updates in a real-time recursion that specifies all required computations. At each time \(t_k = k\Delta t\), for \(k = 1, \ldots, N_T - 1\), carry out an iterative forecast/update and forecast/optimization for each replicate \(j = 1, \ldots, N_R\), as follows.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.
**Forecast/Update.** Use the most recent optimal control $u_{k-1|k-1}^*$ and updated ensemble $y_{k-1|k-1}$ to compute a one-step-ahead forecast ensemble $y_{k|k-1}^j$. The forecast is obtained by solving the following equations over the time interval $[t_{k-1}, t_k]$:

\begin{align}
\frac{dy^j(t)}{dt} &= F_a(y^j, w^j, u_{k-1|k-1}^*), \quad y^j(t_{k-1}) = y_{k-1|k-1}^j \quad \text{from update at } t_{k-1}, \\
G_a(y^j, w^j, u_{k-1|k-1}^*) &= 0.
\end{align}

Then update the ensemble average state and the deviations from the average with new measurements taken at $t_k$, as follows:

\begin{align}
y_{k|k} &= y_{k|k-1} + A_k \left[ z_k - M_k y_{k|k-1} \right], \\
Y_{k|k} &= Y_{k|k-1} T_k, \\
y_{k|k}^j &= y_{k|k} + \tilde{y}_{k|k}.
\end{align}

Here $y_{k|k-1}$ and $y_{k|k}$ are the arithmetic averages of the forecast and updated augmented state replicates $y_{k|k-1}^j$ and $y_{k|k}^j$. The columns of the $N_y$ by $N_{\text{rep}}$ matrices $Y_{k|k-1}$ and $Y_{k|k}$ are composed of the ensemble perturbations $\tilde{y}_{k|k-1}^j = y_{k|k-1}^j - y_{k|k-1}$ and $\tilde{y}_{k|k}^j = y_{k|k}^j - \bar{y}_{k|k}$, for $j = 1, \ldots, N_R$. The $N_y$ by $N_{\text{rep}}$ dimensional matrix $A_k$ is the Kalman gain and the $N_{\text{rep}}$ by $N_{\text{rep}}$ matrix $T_k$ is a weighting matrix that transforms forecast ensemble perturbations into updated ensemble perturbations. Both of these matrices are computed from $Y_{k|k-1}$ and the measurement error covariance matrix $R_k$ [14]. Note that the diagnostic variables need not be updated in the Kalman filter since they can always be derived from the updated states.

**Forecast/Optimization.** After the updated ensemble at $t_k$ has been derived perform an iterative search for the optimal control sequence $u_{k:N_T-1|k}^*$ defined by

\begin{align}
u_{k:N_T-1|k}^* &= \arg\min \{ V_k(u_{k:N_T-1}) \}
\end{align}

subject to the following constraints:

$$u^L \leq u_{k:N_T-1} \leq u^U,$$

where $u^L$ and $u^U$ are, respectively, lower and upper bounds on the control variables. The conditional states $y_{k:N_T|k}^j$ needed to evaluate $V_k(u_{k:N_T-1})$ for the proposed control $u_{k:N_T-1}$ are defined by the following set of equality constraints (forecast equations):

\begin{align}
\frac{dy^j(t)}{dt} &= F_a(y^j, w^j, u_{k:N_T-1}), \quad y^j(t_k) = y_{k|k}^j \quad \text{from update at } t_k, \\
G_a(y_{l+1}^j, w_{l+1}^j, u_l) &= 0, \quad l = k, \ldots, N_T - 1.
\end{align}

The optimization problem may be solved with a variety of techniques. If a gradient-based method is used it is convenient to derive the gradients from an adjoint procedure, by using Lagrange multipliers to augment $V_k(u_{k:N_T-1})$ with (2.15) and (2.16).

The update recursion given in (2.9) through (2.13) must be initialized at $k = 1$ with a set of unconditional augmented state replicates $y_{0|0}^j$ and with an initial control $u_{0|0}^*$. The initial (or prior) state replicates divide into parameters $\alpha_{0|0}^j$ sampled from a specified prior parameter distribution and dependent variables $x_{0|0}^j$ sampled...
from a specified prior distribution of initial states. These prior replicates need to be carefully generated so that they are consistent with the spatial structure likely to be encountered in a given application. For this reason, the prior replicate generation process should be application specific rather than generic. An example is illustrated in the subsurface transport example discussed in section 3.

The initial control \( u^*_{0,0} \) is obtained by solving the following unconditional optimization problem:

\[
(2.17) \quad u^*_{0,N_T-1|0} = \arg\min \{ V_0(u_{0:N_T-1}) \}
\]

subject to the following constraints:

\[
\begin{align*}
(2.18) & \quad u^L \leq u_{0:N_T-1} \leq u^U, \\
(2.19) & \quad \frac{dy^j(t)}{dt} = F_a(y^j, w^j, u_{0:N_T-1}), \quad y^j(t_0) = y^j_{0|0} \text{ from initial condition (prior) at } t_0, \\
& \quad G_a(y^j_{l+1|l}, w^j_{l+1}, u_l) = 0, \quad l = k, \ldots, N_T - 1.
\end{align*}
\]

The initial control \( u^*_{0|0} \) is the first entry in the optimal sequence \( u^*_{0:N_T-1|k} \).

Although the optimization problem of (2.14) provides a sequence \( u^*_{k,N_T-1|k} \) of optimal control values for the entire control horizon, these values are only conditioned on measurements taken through the current time \( t_k \). Also, only the first value \( u^*_{k,k|k} \) in the sequence is actually needed at this time. Consequently, we can discard the future controls \( u^*_{k+1:N_T-1|k} \) derived from (2.14) and apply only \( u^\text{opt}_k = u^*_{k,k|k} \) to the true system at \( k\Delta t \). Discarded controls for the later times \( (k+1)\Delta t, \ldots, (N_T-1)\Delta t \) will eventually be replaced by new optimization solutions derived from new information obtained at these times. This approach is a version of model predictive control (MPC), which is widely used because it performs well, is easy to implement, and is computationally efficient.

The expressions given in this section define all of the operations in the real-time stochastic control loop of Figure 1. In the next section we consider an approximate but computationally efficient reduced-order formulation of this real-time control procedure.

### 2.2. Reduced-order formulation

The stochastic real-time control algorithm described above requires multiple evaluations of the system equations given in (2.9) through (2.16): approximately one evaluation for each ensemble replicate for each iteration of the optimization search algorithm at each control/measurement time. In most distributed parameter applications, including the solute transport problem examined in this paper, the most computationally demanding part of these evaluations is the solution of the forward model equations given in (2.1) and (2.2). The model equations are typically high dimensional since they are derived by discretizing a set of partial differential equations on a fine spatial grid. This is not necessarily the most efficient way to characterize the true system for stochastic control applications.

The high cost of the model solutions has the practical effect of limiting the ensemble size that can be used in such applications. This, in turn, limits the effectiveness of the control procedure by compromising the accuracy of the measurement update operation, which relies on sample moments of the augmented state vector, and by compromising the accuracy of the optimization procedure, which relies on sample moments of the cost-to-go. There is clearly a tradeoff between the accuracy benefits
obtained by using a high-dimensional model with a small ensemble versus the statistical benefits obtained by using a lower-dimensional model with a large ensemble. In order to investigate this tradeoff we examine real-time control performance with the reduced-order system model described in [13].

Reduced-order models of distributed parameter systems have frequently been derived by applying linear techniques such as principle orthogonal decomposition (POD). Examples include [1, 5, 7, 20]. Several recent studies have proposed more advanced model reduction methods that may be suitable for nonlinear stochastic optimization applications. These include adaptive procedures that improve the model reduction process by successively updating linear approximations [17] and trust-region methods [3]. Most of these model reduction methods require computationally expensive online modifications of the reduced-order model.

Here we present a nonlinear reduced-order modeling approach that is implemented primarily offline. It uses a POD expansion to approximate the full-order model state but substitutes this expansion into a nonlinear approximation of the spatially discretized dynamic equations. The resulting nonlinear reduced-order model captures key dynamic properties of the true system but is much less computationally demanding than a full-order model obtained by discretizing the distributed equations on a fine grid. This approach to model reduction moves expensive computations offline so the online real-time control operation can use ensembles of reasonable size. The key elements of the approach are summarized below. Further details are provided in [13] and illustrated in our example.

Our model reduction procedure replaces the uncertain model variable vectors appearing in (2.1) and (2.2) with low-dimensional linear combinations of specified basis vectors [19], as follows:

\begin{align}
\alpha & \approx \Theta \hat{\alpha}, \\
x & \approx \Phi_x \hat{x}, \\
w & \approx \Phi_w \hat{w},
\end{align}

where \( \Theta \) is an orthonormal \( N_{\alpha} \times N_{\alpha} \) matrix whose columns are a selected set of discrete cosine transform (DCT) basis vectors and \( \Phi_x \) and \( \Phi_w \) are orthonormal \( N_x \times N_x \) and \( N_w \times N_w \) matrices whose columns are POD basis vectors. These expressions may be combined to give a concise approximation for the \( N_y = N_\alpha + N_x \) dimensional augmented state vector:

\( y \approx \Psi_a \hat{y}, \)

where:

\( \hat{y} = \begin{bmatrix} \hat{x} \\ \hat{\alpha} \end{bmatrix}, \quad \Psi_a = \begin{bmatrix} \Phi_x & 0 \\ 0 & \Theta \end{bmatrix}. \)

The reduced-order model is formulated in terms of the low-dimensional augmented variables \( \hat{y} \). The DCT basis vectors used in the \( \hat{\alpha} \) expansion are obtained from standard periodic functions frequently used for image compression. The POD basis vectors used in the \( \hat{x} \) and \( \hat{w} \) expansions are the leading singular vectors of so-called snapshot matrices that are designed to convey the dominant dynamical properties of the system. Construction of the snapshot matrices is discussed in more detail below.
When deriving the reduced-order model it is convenient to decompose the vector functions $F_a$ and $G_a$ in (2.3) and (2.4) into two types of nonlinear terms: (i) $F_{aq}$ and $G_{aq}$, which are linear and/or quadratic in the state and control variables, and (ii) $F_{an}$ and $G_{an}$, which are neither linear nor quadratic. The resulting full-order model equations can be written in a residual form as

\begin{align}
\Delta F &= \frac{dy}{dt} - F_{aq}(y, w, u) - F_{an}(y, w, u) = 0, \quad y(t_0) = y_0 \text{ specified,} \\
\Delta G &= G_{aq}(y, w, u) + G_{an}(y, w, u) = 0.
\end{align}

(2.25) \hspace{2cm} (2.26)

The quadratic versus nonquadratic distinction made here is especially useful in problems where the dominant nonlinearities involve products of states and/or products of states and controls. Examples include diffusion problems with uncertain coefficients and problems that involve the advection of scalar or vector quantities (such as the example discussed later in this paper). Such problems often generate quadratic nonlinearities.

We can expand the linear/quadratic terms $F_{aq}$ and $G_{aq}$ in second-order Taylor series (which are exact) about a nominal state $y_0$, $w_0$, and control $u_0$ to obtain

\begin{align}
\Delta F &= \frac{dy}{dt} - F_{aq}(y_0, w_0, u_0) - F_{aq,y} \delta y - F_{aq,w} \delta w \\
&\hspace{1cm} - F_{aq,u} \delta u - F_{aq,yu}(\delta y \otimes \delta w) \\
&\hspace{1cm} - F_{aq,yw}(\delta y \otimes \delta u) - F_{an}(y, w, u) = 0, \quad y(t_0) = y_0 \text{ specified,} \\
\Delta G &= G_{aq}(y_0, w_0, u_0) + G_{aq,y} \delta y + G_{aq,w} \delta w + G_{aq,u} \delta u \\
&\hspace{1cm} + G_{aq,yu}(\delta y \otimes \delta w) + G_{aq,yw}(\delta y \otimes \delta u) + G_{an}(y, w, u) = 0,
\end{align}

(2.27) \hspace{2cm} (2.28)

where $\delta y = y - y_0$ and $\delta u = u - u_0$. Here we assume that the only non-zero quadratic terms are $\delta y \otimes \delta w$ and $\delta y \otimes \delta u$. This is a valid assumption for our subsurface solute transport example. In other applications where there are additional non-zero quadratic terms the Taylor expansion can be extended as appropriate.

Note that the Kronecker product $\Gamma \otimes \Pi$ for an $m$ by $n$ matrix $\Gamma$ and an $r$ by $w$ matrix $\Pi$ is an $mr$ by $nw$ matrix defined as

\[
\Gamma \otimes \Pi = \begin{bmatrix}
\Gamma_{11} \Pi & \cdots & \Gamma_{1n} \Pi \\
\vdots & \ddots & \vdots \\
\Gamma_{m1} \Pi & \cdots & \Gamma_{mn} \Pi
\end{bmatrix}.
\]

For example, $\delta y \otimes \delta u$ in (2.27) is an $N_yN_u$ by 1 dimensional column vector composed of products of individual elements of $\delta y$ and $\delta u$. Here $F_{aq,y}$, $G_{aq,y}$, $F_{aq,w}$, $G_{aq,w}$, $F_{aq,u}$, and $G_{aq,u}$ are first derivative coefficient matrices while $F_{aq,yw}$, $G_{aq,yw}$, $F_{aq,yu}$, and $G_{aq,yu}$ are second derivative coefficient matrices. The scalar derivatives that form the elements of these matrices are arranged to be consistent with the ordering implied by the Kronecker products. Since $F_{aq}$ and $G_{aq}$ are quadratic the first derivatives depend linearly on $y_0$, $w_0$, and $u_0$ while the second derivatives are constants.

The reduced-order model approximates (2.27) and (2.28) by replacing the full-order states and parameters by the basis function expansion of (2.23). When this is done the full-order model equations cannot be satisfied exactly but can only be satisfied in a weak sense. That is, only the projections of the residuals $\Delta F$ and $\Delta G$ onto the subspaces spanned by the associated basis vectors can be zero. A reduced-order set of model equations is obtained by substituting the basis vector expansions
into (2.27) and (2.28), projecting each equation onto the appropriate subspace (i.e.,
by premultiplying the residual by either \( \Psi_a^T \) or \( \Phi_w^T \)) and rearranging terms:

\[
\frac{d}{dt}(\Psi_a \hat{y}) = \Psi_a^T F_{aq}(y_0, w_0, u_0) + \Psi_a^T F_{aq,y} \Psi_a \delta \hat{y} + \Psi_a^T F_{aq,w} \Psi_w \delta \hat{w} + \Psi_a^T F_{aq,u} \delta u + \Psi_a^T F_{an}(\Psi_a \hat{y}, \Psi_w \hat{w}, u) = 0, \quad y(t) = y_0 \text{ specified},
\]

(2.29)

\[
\Phi_w^T G_{aq}(y_0, w_0, u_0) + \Phi_w^T G_{aq,y} \Psi_a \delta \hat{y} + \Phi_w^T G_{aq,u} \Psi_w \delta \hat{w} + \Phi_w^T G_{aq,u} \delta u + \Phi_w^T G_{aq,yu}(\Psi_a \delta \hat{y} \otimes \Psi_w \delta \hat{w}) + \Phi_w^T G_{aq,y}(\Psi_a \delta \hat{y} \otimes \delta u) + \Phi_w^T G_{an}(\Psi_a \hat{y}, \Psi_w \hat{w}, u) = 0.
\]

The properties of the Kronecker product and the orthonormality of \( \Phi_w \) and \( \Psi_a \) can be used to write these equations as a nonlinear reduced-order model that depends on the low-dimensional state vector \( \hat{y} \) rather than the high-dimensional vector \( y \):

\[
\frac{d\hat{y}}{dt} = \hat{F}_a(\hat{y}, \hat{w}, u) = \left[ \Psi_a^T F_{aq}(y_0, w_0, u_0) \right] + \left[ \Psi_a^T F_{aq,y} \Psi_a \right] \delta \hat{y} + \left[ \Psi_a^T F_{aq,w} \Psi_w \right] \delta \hat{w} + \left[ \Psi_a^T F_{aq,u} \right] \delta u + \left[ \Psi_a^T F_{an}(\Psi_a \hat{y}, \Psi_w \hat{w}, u) \right] \delta \hat{y} \otimes \delta u,
\]

(2.31)

\[
\hat{G}_a(\hat{y}, \hat{w}, u) = \left[ \Phi_w^T G_{aq}(y_0, w_0, u_0) \right] + \left[ \Phi_w^T G_{aq,y} \Psi_a \right] \delta \hat{y} + \left[ \Phi_w^T G_{aq,u} \Psi_w \right] \delta \hat{w} + \left[ \Phi_w^T G_{aq,yu}(\Psi_a \otimes \Psi_w) \right] \delta \hat{y} \otimes \delta \hat{w} + \left[ \Phi_w^T G_{aq,y}(\Psi_a \otimes \Psi_w) \right] \delta \hat{y} \otimes \delta u + \left[ \Phi_w^T G_{an}(\Psi_a \hat{y}, \Psi_w \hat{w}, u) \right] = 0,
\]

(2.32)

where \( I_u \) is the identity matrix which has the same dimension as \( u \).

In both of these equations all the coefficient terms in square brackets can be derived offline, from the full-order model equations, before measurements are collected or controls are applied. These offline terms depend only on the nominal state and control trajectories, which are specified in advance and are not changed during the control process. Methods for selecting the nominals are application dependent and discussed in more detail in section 3. The coefficient matrices are sparse, with non-zero elements arranged according to the spatial connectivity of the computational grid. It should be noted that the second-order reduced state vector products (such as \( \delta \hat{y} \otimes \delta \hat{w} \)) are large (dimension \( N_y N_w \)) but many of the elements in these vectors may not contribute, depending on the application, and can be omitted since they are multiplied by zeros in the coefficient matrices. Although the quadratic terms contribute more to the reduced-order models’ computational effort than the linear terms, they still provide a saving relative to a full-order calculation.

The final terms without brackets appearing in (2.31) and (2.32) cannot generally be written as the simple product of a predetermined coefficient matrix and reduced-order states. The full state must be reconstructed online whenever these terms are evaluated. There are a number of ways to deal with this requirement, ranging from neglecting the final terms, through approximating them with a procedure such as the
discrete empirical interpolation method (DEIM) [6], to computing them explicitly. The last of these approaches is illustrated in the example of section 3. In any case, there is no need to run the full-order model online. The nonlinear reduced-order model given in (2.31) and (2.32) can be used to forecast the reduced-order state for any specified control and initial ensemble, including the ensemble produced by a Kalman filter update. An approximation of the full-order state can be reconstructed as needed by substituting the reduced-order state into (2.23).

The accuracy of the reduced-order model depends largely on how well the DCT and POD basis vectors capture dominant sources of variability in the parameters and states, respectively. All of the DCT vectors associated with an image of a given size are prespecified. The particular DCT vectors included in the matrix \( \Psi \) are members of the subset that explains the greatest fraction of the variability encountered across the prior ensemble [12]. By contrast, the POD basis vectors are not prespecified but depend on the dynamical properties of the system. In our example these vectors are derived from a large number of full-order snapshot simulations that describe system behavior over a range of conditions. The simulation results are assembled in a so-called snapshot matrix. Each column of this matrix is a vector of simulated states or, in some cases, a vector of derivatives of states with respect to a particular uncertain parameter, with each entry taken at a different location. The POD basis vectors included in the matrix are the leading singular vectors of the snapshot matrix.

The snapshot simulations are run for many different possible input parameters and controls. These inputs define the range of conditions that the reduced-order model is designed to reproduce. It is reasonable to draw the snapshot parameter values from the prior ensemble, since this ensemble conveys our prior understanding of parameter variability. In a similar way, it is reasonable to use snapshot control sequences that optimize performance for the parameter values included in the prior ensemble. Each snapshot control sequence can be obtained by optimizing the open loop cost-to-go \( J_k \) for one of the replicates in the snapshot parameter set. Our numerical experiments indicate that snapshots typically give acceptable reduced-order model performance if they span the range of actual conditions reasonably well. In practice, the performance of the reduced-order model should be tested for each application to make sure that its various approximations and assumptions are justified.

A reduced-order approximation of the full order real-time control algorithm is obtained by writing the control equations developed in section 2.1 in terms of the low-dimensional augmented state vector \( \hat{y} \). At each time \( t_k = k\Delta t \), for \( k = 1, \ldots, N_T - 1 \), carry out an iterative forecast/update and forecast/optimization, for each replicate \( j = 1, \ldots, N_R \), as follows.

**Forecast/Update.** Use the most recent optimal control \( u_{k-1|k-1}^* \) and updated ensemble \( \hat{y}_{k-1|k-1}^j \) to compute a one-step-ahead forecast ensemble \( \hat{y}_{k|k-1}^j \). The forecast is obtained by solving the following equations over the time interval \([t_{k-1}, t_k]\):

\[
\begin{align*}
\frac{d\hat{y}^j}{dt} &= \hat{F}_a(\hat{y}^j, \hat{w}^j, u_{k-1|k-1}^*), \quad \hat{y}^j(t_{k-1}) = \hat{y}_{k-1|k-1}^j \quad \text{from update at } t_{k-1}, \\
\hat{G}_a(\hat{y}^j, \hat{w}^j, u_{k-1|k-1}^*) &= 0.
\end{align*}
\]

Set \( \hat{y}_{k|k-1} = \hat{y}^j(t_k) \) and \( \hat{w}_{k|k-1} = \hat{w}^j(t_k) \). Then update the augmented state ensemble average and the deviations from the average with new measurements taken at \( t_k \), as follows:
\[ (2.35) \quad \hat{y}_{k|k} = \hat{y}_{k|k-1} + \hat{A}_k [z_k - M_k \Psi_a \hat{y}_{k|k-1}], \]
\[ (2.36) \quad \hat{Y}_{k|k} = \hat{Y}_{k|k-1} + \hat{T}_k, \]
\[ (2.37) \quad \hat{y}^j_{k|k} = \hat{y}_{k|k} + \hat{y}^j_{k|k}, \]

where all reduced-order variables (with hats) are defined in the same ways as their counterparts in section 2.1. The updated diagnostic variables are derived from \( \hat{y}^j_{k|k} \) using (2.32).

**Forecast/Optimization.** After the updated ensemble at \( t_k \) has been derived perform an iterative search for the optimal control sequence \( u^*_{k:N_T-1|k} \) defined by

\[ (2.38) \quad u^*_{k:N_T-1|k} = \text{argmin} \{ V_k(u_{k:N_T-1}) \} \]

subject to the following constraints:

\[ u^L \leq u_{k:N_T-1} \leq u^U, \]

where \( u^L \) and \( u^U \) are, respectively, lower and upper bounds on the control variables. The conditional states \( y^j_{k:N_T|k} \) needed to evaluate \( V_k(u_{k:N_T-1}) \) for the proposed control \( u_{k:N_T-1} \) are defined by the following set of equality constraints (reduced-model forecast equations):

\[ (2.39) \quad \frac{d\hat{y}^j(t)}{dt} = \hat{F}_a(\hat{y}^j, \hat{w}^j, u_{k:N_T-1}), \quad \hat{y}^j(t_k) = \hat{y}^j_{k|k} \text{ from update at } t_k, \]
\[ (2.40) \quad \hat{G}_a(\hat{y}^j_{l+1}, \hat{w}^j_{l+1}, u_l) = 0, \quad l = k, \ldots, N_T - 1, \]
\[ (2.41) \quad y^j_{k:N_T|k} = \Psi_a \hat{y}^j_{k:N_T|k}. \]

As in the full-order case, the update recursion must be initialized at \( k = 1 \) with a set of unconditional augmented state replicates \( \hat{y}^j_{0|0} = \Psi^T y^j_{0|0} \) and with an initial control \( u^*_{0|0} \). The initial states are obtained from the prior ensemble and the control is derived offline from (2.17) through (2.19). Also, an MPC approach is adopted and only the first value \( u^*_{k|k} = u^*_{k|k} \) in the control sequence from (2.38) is retained and applied to the true system at \( k \Delta t \).

The reduced-order control algorithm described above provides an alternative to the full-order algorithm. Note that the control vector has the same dimension in the full- and reduced-order formulations. However, the reduced-order optimization, forecasting, and update constraint equations are formulated in terms of the low-dimensional vector \( \hat{y} \) rather than the high-dimensional vector \( y \). This provides an important computational advantage that becomes even more significant as the system dimension increases.

**3. Numerical experiment.** The purpose of our numerical experiment is to test the performance of a stochastic real-time control algorithm under controlled conditions. We adopt an approach that accounts for uncertainty in system parameters and measurements but assumes that the equations used to describe the system are perfectly known (i.e. we adopt a perfect model assumption). The numerical experiment relies on a virtual true system simulated with a typical set of spatially variable model parameters. Virtual measurements of the true system response are derived by adding random noise to the true states, as indicated in Figure 1. Note that the true states depend on the controls which are, in turn, derived from the noisy virtual measurements. Consequently, different measurement error realizations give different true states as well as different measurements.
In the discussion that follows we are particularly interested in (1) the merits of using closed-loop control (with measurement updates) versus open-loop control (without measurement updates), (2) the performance of controllers based on full versus reduced-order models of the true system, and (3) the tradeoff between mean performance and performance uncertainty. Our investigation relies on a subsurface solute transport (groundwater remediation) problem that illustrates many of the key issues encountered in spatially distributed real-time control applications.

3.1. Experimental setup. Our experiment considers control of subsurface flow and transport in a thin isotropic confined aquifer with a constant layer thickness, contained in the rectangular horizontal domain shown in Figure 2. In the absence of control a solute originating along the bottom boundary travels upward towards the water supply well at P4. Controlled pumping at wells P1, P2, P3 pulls the solute away from P4 and out of the system, incurring both pumping and treatment costs. The best control satisfies water quality requirements at the supply well while minimizing the cost of remediation.

We suppose that measurements of the system response are from fully screened wells and that a two-dimensional vertically averaged description is appropriate. The corresponding flow and transport model can be written as

\[
\begin{align*}
\frac{\partial h(r,t)}{\partial t} &= -\nabla \cdot q - \sum_{i=1}^{N_w} Q_i \delta(r-r_i), \\
q(r,t) &= -K(r)d\nabla h(r,t), \\
\frac{\partial c(r,t)}{\partial t} &= -\nabla \cdot (gc) + \theta D \nabla^2 c - \sum_{i=1}^{N_w} Q_i c \delta(r-r_i).
\end{align*}
\]
Here \( h(r,t) \) is the piezometric head, \( q(r,t) \) is the two-dimensional Darcy velocity, \( c(r,t) \) is the solute concentration, \( s \) is the specific storage, \( Q_i(t) \) is the volumetric flux pumped from the aquifer at well \( i \), \( \delta(r - r_i) \) is a two-dimensional spatial Dirac delta function, \( K(r) \) is the isotropic hydraulic conductivity, \( d \) is the constant aquifer thickness, and \( \theta \) is the porosity. To simplify the transport equation we assume the dispersion coefficient \( D \) is constant and isotropic in all directions. The head and concentration equations are accompanied by initial and boundary conditions that are discussed in more detail below. Note that the only nonlinearities in these equations are products of the unknown dependent variables \( h, q_x, q_y, \) and \( c \), the unknown parameter \( K \), and their respective spatial gradients. That is, these equations are all bilinear.

If we discretize the above model equations over space as described in [13] the result is

\[
\begin{align*}
(3.4) & \quad s \frac{\partial h(t)}{\partial t} - Bq(t) = S_1(t), \\
(3.5) & \quad B^T h(t) + A(\eta)q(t) = S_2(t), \\
(3.6) & \quad \theta \frac{\partial c(t)}{\partial t} + E(q)c(t) + Lc(t) = S_3(t),
\end{align*}
\]

where \( h \) is an \( N_h \) dimensional column vector containing the unknown grid cell pressure heads, \( q \) is an \( N_q = N_{qx} + N_{qy} \) dimensional column vector containing the \( x \) and \( y \) components of the unknown grid cell Darcy velocities, and \( c \) is an \( N_c \) dimensional vector containing unknown grid cell solute concentrations. The components of the \( N_h \) by \( N_q \) matrix \( B \) in (3.4) and (3.5) are constant while the components of the \( N_q \) by \( N_q \) matrix \( A(\eta) \) in (3.5) depend linearly on the components of the \( N_c \) dimensional hydraulic resistivity vector \( \eta \), which is defined by \( \eta_i = 1/K_i \) for each grid cell \( i \). Although our model and Kalman filtering algorithms are formulated in terms of resistivity we report results in terms of the more familiar log hydraulic conductivity.

The \( N_h \) and \( N_q \) dimensional right-hand-side vectors \( S_1 \) and \( S_2 \) depend linearly on the flow boundary conditions, including those imposed at the pumping wells. The matrix \( E(q) \) in (3.6) depends nonlinearly on the velocity vector \( q \) when upwinding is used to mitigate numerical dispersion. The matrix \( L \) is constant if the dispersion coefficients are assumed to be constants that do not depend on velocity. If the solute is assumed to exit primarily through the wells and the flow rates at the wells are prescribed the pumping term in (3.6) can be incorporated into the matrix \( E \) and \( S_3 = 0 \). The bilinear terms of the spatially discretized transport equation, which involve simple products of velocity and concentration, are included in the quadratic operator \( F_{aq} \) while the terms affected by upwinding are incorporated into the nonlinear term \( F_{an} \).

In this example the augmented state vector \( y \) defined in section 2.2 is composed of the prognostic variables \( h \) and \( c \), and the parameter \( \eta \). The diagnostic variable \( w \) contains the components of the velocity vector \( q \). The definitions of the vector functions \( \hat{F}_a(\hat{y}, \hat{w}, u) \) and \( \hat{G}_a(\hat{y}, \hat{w}, u) \) for the example follow from (3.4) through (3.6).

The discretized flow equation system given in (3.4) and (3.5) can be further simplified if we suppose that the pumping rates are constant between control times and that the specific storage is small enough for the flow system to reach an approximate steady state between these times. That is, the time for the velocity field to respond to changes in pumping rates is small compared to the time between measurement and pumping updates. In this case we only need to solve a steady state version of (3.4) for constant \( h \) and \( q \) over each interval.

For our numerical example a \( 320 \times 320 \times 1 \)-m computational domain of Figure 2 is discretized into a \( 32 \times 32 \times 1 \) uniform computational grid of \( 10 \times 10 \times 1 \)-m cells.
The spatially uniform porosity has a known value of 0.2. The dispersion coefficient is assumed to be an isotropic flow-independent constant over the whole domain. The flow boundary conditions are zero water flux on the east and west sides and constant head boundaries of 10 m and 0 m on the south and north sides. These conditions impose a regional flow from south to north with local fluctuations induced by well pumping and log hydraulic conductivity variations.

The transport boundary conditions are zero solute flux on the east and west sides and zero solute concentration on the south side, except during injection at the solute source located between $x = 150$ m and $x = 170$ m in the middle of the southern boundary. Solute is injected on this boundary at a constant rate of $5 \text{ m}^3/\text{day}$ and a constant concentration of $50 \text{ mg/L}$ for the first 200 days. Outward solute transport is permitted across the northern boundary but the simulation time is limited to keep the solute boundary flux small. The initial concentration is zero everywhere in the domain.

These boundary and initial conditions generate a plume of finite extent that travels south to north, gradually moving past the wells but staying largely in the computational domain during the 400-day simulation period. The simulation period is divided into 4 control intervals of 100 days each. The model time step is 10 days. Only wells P1, P2, and P3 are controlled, while well P4 is used for drinking water supply and has a constant pumping rate of $10 \text{ m}^3/\text{day}$. Since the dispersion tensor is isotropic and constant and the flow system is at steady state over each control interval the transport equation is linear, with coefficients that change instantly when the pumping rates change. This simplification facilitates the reduced-order modeling process.

In the full-order control algorithm the controlled pumping rates are adjusted to minimize the objective function given in (2.8), which is constructed from the cost-to-go function in (2.7), with the current cost function for replicate $j$ defined as

\begin{equation}
\gamma^j_l (y_{l\mid k}^j, \mathbf{y}^j_{l+1\mid k}, \mathbf{u}_l) = \sum_{i=1}^{3} u_{l,i}^2 + w \cdot c^j_{l,4\mid k},
\end{equation}

where $u_{l,i}$ is the pumping rate at well $i = 1, 2, 3$ over the control interval starting at $t_l$ and $c^j_{l,4\mid k}$ is the concentration forecast at time $t_l$, conditioned on measurements taken through $t_k$. This concentration forecast is one of the elements in the full-order augmented state vector $y_{l\mid k}$. The current cost is a weighted combination of the sum of squared pumping rates at the remediation well, which serves as a surrogate for the remediation pumping cost, and the solute concentration at P4. Ideally, both quantities should be as small as possible. The lower and upper bounds for the controlled pumping rates (negative values indicate pumping out of system) are

$$-50 \text{ m}^3/\text{day} \leq u_{l,i} \leq 0, \quad i = 1, 2, 3.$$

The expressions given here also apply to the reduced-order version of the objective function, with the equation numbers and variables replaced with their reduced-order counterparts wherever applicable.

An ensemble approach to real-time characterization and control relies on prior parameter and initial state ensembles that describe our imperfect knowledge of system properties before measurements are collected. In a subsurface application these prior ensembles should be geologically realistic and appropriate for the site under investigation while also properly conveying, through replicate-to-replicate variability,
uncertainty about the true parameters. One flexible option for generating realistic prior replicates is to use multipoint geostatistical techniques based on training images [8]. Training images can be obtained from local experts or constructed from field data. In any case, they are intended to provide useful qualitative information on spatial structures likely to be found in the true system.

The 50 prior resistivity replicates used in our example are generated from a specified training image with the multipoint algorithm FILTERSIM [21]. We use this algorithm to define binary replicates that delineate boundaries between permeable and less permeable geological facies. Resistivity values are assigned to each of the two facies and the resulting binary resistivity images are approximated with a truncated DCT expansion that uses 100 DCT coefficients, about 10% of the number of pixels. The same expansion is used to obtain the basis functions ($\Theta$ columns) for the reduced-order hydraulic resistivity approximation. Since the DCT expansion is approximate the replicates obtained from the retained coefficients are no longer binary but exhibit gradual transitions between higher and lower resistivity areas.

Figure 3(a) shows the single replicate designated as the true field, plotted in log hydraulic conductivity units. Figure 3(b) shows the binary training image used to generate the true replicate as well as the 50 replicates in the prior ensemble. Some typical members of this ensemble are shown in Figure 3(c). The training image is characterized by a single high-conductivity diagonal channel that is reflected, to various degrees, in the true field and in most of the prior replicates. Note that the true replicate is not included in the prior ensemble.

An ensemble Kalman filter is used to update the augmented state replicates with point measurements of head and concentration. These measurements are collected at the nine locations indicated by the blue diamonds in Figure 2. The measurement locations include the four pumping wells. All measurements are taken at the end of the four control steps so there are four updates during the 400-day simulation.

The reduced-order model is constructed from standard DCT basis vectors for the model’s parameters (resistivities) and POD basis vectors for the model’s dynamic states (heads, velocities, and concentrations). The POD basis vectors for each
dynamic state (head, two velocity components, and concentration) are the leading singular vectors of a snapshot matrix composed of full-order simulation results for different combinations of parameter values and control sequences. These snapshots are selected to give a representative picture of the range of possible system responses. Each control sequence used for snapshot generation minimizes the deterministic cost-to-go for a particular parameter replicate. The head and velocity snapshots include the relevant state as well as derivatives of the state with respect to the 1024 hydraulic resistivities. Each of the resulting head and velocity snapshot matrices has 82320 columns, of which 400 (4 control steps for 100 combinations of 10 replicates and 10 controls) are state vectors and 81920 (4 control steps for 1024 resistivities for 20 combinations of 2 replicates and 10 controls) are derivative vectors. The number of columns in the concentration snapshot matrix is 4000 (40 time steps for 100 combinations of 10 replicates and 10 controls). No derivatives are included in this matrix.

The leading snapshot matrix singular vectors that are retained as POD basis vectors account for 99%, 93%, 93%, and 99% of the energy in the head, x velocity, y velocity and concentration, respectively. This gives reduced-order dimensions of 56, 103, 97, and 194 for the four state variables, each of which originally has a full-order dimension of about 1024. The specific snapshot simulation specifications and POD truncation criteria used here provide a comprehensive picture of full-order model response but could easily be modified, if desired.

3.2. Experimental results. The results for our numerical experiment are organized around several test cases defined as follows.
- Deterministic control: perfect knowledge of hydraulic conductivity, best possible control performance with no uncertainty.
- Full-order ensemble closed-loop control: uncertain parameters, measurement updates included.
- Reduced-order ensemble closed-loop control: uncertain parameters, measurement updates included.
- Full-order ensemble open loop control: uncertain parameters, no measurement updates.
- Reduced-order ensemble open loop control: uncertain parameters, no measurement updates.
- No control: reference case with no pumping at control wells P1, P2, P3. Constant pumping for water supply only at well P4.

These cases cover a range of control options, varying from a best case with control based on perfect information about the true system to a worst case with no control. The remaining cases represent intermediate options that all account for uncertainty but use different amounts of information and make different approximations. We examine performance of these cases for \( \mu = 0 \) in (2.8). This gives a control strategy that minimizes the mean cost-to-go.

Figure 4 shows the optimal well control sequences and corresponding true cost for each of the controlled cases defined above. Here the true cost is defined as

\[
J_{\text{tot}}(u_{0:N_T-1}^{\text{opt}}, y_{0:N_T-1}) = \sum_{t=0}^{N_T-1} \gamma_t(y_t, y_{t+1}, u_t^{\text{opt}}) = \sum_{t=0}^{N_T-1} \left[ \sum_{i=1}^{3} (u_{t,i}^{\text{opt}})^2 + w \cdot c_{t,4} \right],
\]
where the state $c_{i,4}$ is the P4 water supply well concentration obtained at time $t_l$ from the true hydraulic resistivity and from the sequence of optimal controls $u_{k|N,F}^{opt}$ generated by the real-time algorithm. Figure 5 shows spatial plots of the true solute concentrations obtained by applying the above six control sequences to the true system (simulated with the full-order model). Figure 6 shows the true concentration time series at the P4 water supply well for all six sequences. The behavior of the various control options is best understood by examining all three figures.

The deterministic control case indicated with dark blue in Figure 4 is characterized by minimal pumping at P1 but significant pumping at P2 and P3 during the second and third control steps. In this case the control strategy allows the natural flow field to move the solute plume to the diagonal permeable channel in the first control interval. Then the solute is extracted through pumping from P2 and P3 during the second and third control intervals. Pumping at these wells is reduced during the fourth control interval since the savings in pumping cost outweighs the penalty incurred from the small concentration at P4.

The full- and reduced-order closed-loop controllers shown in cyan and green, respectively, pump mostly at P2 and P3 in the third control interval, lagging somewhat behind the perfect deterministic controller. The most distinctive difference between the full- and reduced-order strategies is more pumping from P1 in the third control interval for the reduced-order case. This probably occurs because the reduced-order model ensemble locates the permeable channel somewhat closer to P1. The combination of slightly higher concentration at water supply well P4 and more pumping at P1 gives a reduced-order cost that is somewhat higher than the corresponding full-order.
Fig. 5. Spatial distributions of true solute concentration at times 0, 100, 200, 300, 400 days for six control strategies with well locations indicated by black (well P1), red (well P2), cyan (well P3), and magenta (well P4) crosses.

cost. It is interesting to note that the perfect deterministic control gives lower total cost but higher late-time concentration at P4 than either of the closed-loop controls. The lower cost reflects the fact that the peak deterministic pumping rates are lower than those used in the two closed-loop cases.

The full- and reduced-order open-loop controllers shown in orange and brown, respectively, do not benefit from measurements and distribute pumping more evenly over time. This increases the actual cost since concentrations at water supply well P4 are significantly higher. The open-loop controls tend to pump earlier than the corresponding closed-loop controls, suggesting that they predict that the plume will arrive at the permeable channel earlier. The reduced-order control again pumps more at P1 but the overall impact on cost is minor compared to the corresponding full-order case.
Figure 5 reveals that the five controlled solute plumes have similar shapes. However, Figure 6 shows that the closed- and open-loop controllers give significantly different concentrations at P4. Overall, the closed-loop controllers work very well, compensating for uncertainty that is not present in the deterministic case. In fact, the deterministic and closed-loop controllers perform very similarly. The cost for the closed-loop cases is about 10% higher than for the perfect deterministic case. This can be viewed as the penalty paid for the moderate level of uncertainty encountered in this particular problem. The discussion below focuses on some of the properties of the closed-loop controllers.

Figure 7 compares the true log hydraulic conductivity to the ensemble means obtained from the full- and reduced-order closed-loop control algorithms. The mean log conductivity at time 0 is nearly uniform since it averages over many prior replicates with channels placed in different locations (see Figure 3(c)). This representation of geological structure is the one used by the open-loop controllers and is the reason these controllers do not perform as well as their closed-loop counterparts, which have access to better geological information as early as the beginning of the second control interval. The well-defined channel apparent in the closed-loop mean log conductivity plots indicates that this channel is in the right location in most of the conditional replicates. Figure 7 also shows that the reduced-order controller gives an ensemble mean log conductivity that is comparable in quality to the full-order version.

The control decisions obtained from the closed-loop ensemble algorithms rely on forecasts of solute plume movement. Figure 8 compares the true concentration contours for the two closed-loop controllers to the corresponding ensemble means at various times. Note that the true plumes are different for the full- and reduced-order cases because different controls are applied to the true system in these two cases. The
Figure 7. Spatial distributions of log hydraulic conductivity: (a) true log conductivity; (b) log of ensemble mean conductivity for full- and reduced-order models.

This figure suggests that the full- and reduced-order models give roughly comparable mean characterizations of the corresponding true plumes.

Further insight is provided by Figure 9, which shows time series of the true concentrations and ensembles for each of the four ensemble controllers. The benefits of measurement updating are apparent from a comparison of the open- and closed-loop concentrations. The closed-loop ensemble means (dashed curves) derived from the ensemble (gray curves) closely track the true concentration (dark curves) in P2 and P3 and gradually converge to the true in P4. The ensemble spread reflects the ensemble Kalman filter’s assessment of uncertainty, which should properly represent the true system uncertainty. The effects of measurement updating are revealed in Figure 9 by abrupt changes in individual concentration replicates at update times. The measurement updates usually, but not always, bring the replicates closer to the true concentration. The closed-loop full-order model gives an ensemble that is somewhat more centered on the true value, with fewer outliers, but the reduced-order model does well overall. The full and reduced-order open-loop results are almost identical, indicating that the reduced-order model is able to capture the system’s dynamical behavior and uncertainty over the range of prior resistivity values.

Figure 10 shows the effect of measurement updating on the location and shape of the permeable channel and solute plume. Parts (c) and (d) compare the scattered prior log conductivity replicates to the more clustered conditional replicates obtained after updating. Parts (e) and (f) show a similar comparison for the solute plume. In both cases there is much less variability across the ensemble after updating. The smaller variability that remains reflects the residual uncertainty in log conductivity and concentration.

Our numerical experiment clearly demonstrates the benefits of closed-loop control, which significantly improves performance by correcting for the effects of parameter
uncertainty. The results also show that the reduced-order model proposed here is an acceptable substitute for a much more expensive full-order model. In our example the reduced-order model achieves control performance comparable to the full-order case while requiring only 20% of the computation time. The computational benefit of a reduced-order model will probably be much greater for larger problems. We have restricted our attention here to a full-order problem with about 4000 unknowns because of the substantial computational requirements needed to obtain full-order results for a performance comparison. The primary cost of reduced-order control is the offline derivation of the reduced-order coefficient matrices. Online (real-time) computational costs are a small fraction of the offline costs. This implies that reduced-order ensemble control is particularly attractive when the time available for online computation is limited and it is feasible to move substantial amounts of computation offline.

4. Discussion and conclusions. The numerical experiment described in this paper demonstrates the feasibility of ensemble closed-loop control with a reduced-order model for a particular solute transport problem. The general concepts presented in section 2 and applied to the example in section 3 are broadly applicable.
In particular, it is possible in many different contexts to use (i) an ensemble of possible states to describe uncertainty, (ii) a Kalman filter to update the ensemble with measurements, (iii) a reduced-order model to make ensemble forecasts, and (iv) a stochastic MPC procedure. The key question is how well these concepts will work in any particular application. Ultimately, this question needs to be answered with problem-specific experiments. However, we have observed in our solute transport example some outcomes that have broad significance.

The basic advantages of closed-loop deterministic control are well known. The advantages of stochastic control are less extensively documented but it seems apparent from our experiments that a stochastic approach provides considerable flexibility to choose controls that account for uncertainty in a systematic way. It is more difficult to assess the general feasibility and benefit of using a reduced-order model for ensemble real-time control. The importance of computational issues is related to the control and system response time scales. In general, it is desirable to change controls over time scales comparable to the time it takes for the system to respond to typical changes in uncertain inputs. If the system states change significantly over a control
interval it is more difficult to monitor and control performance. On the other hand, if measurements and control decisions are made frequently there is less time to carry out the extensive computations required by an ensemble real-time control algorithm. So the need for reduced-order modeling depends on the system response and control time scales as well as the computational time required to run a full-order model. In problems involving relatively slowly responding geological systems, such as the one considered here, the time required to carry out ensemble control computations with a full-order model may be significantly less than the time available to make a decision. On the other hand, in problems involving faster responding atmospheric systems a reduced-order model may be essential. The type of model reduction used

---

**Fig. 10.** Spatial distributions for ensemble closed-loop control with reduced-order model: (a) true log conductivity field; (b) true concentration on day 400; (c) five typical prior log conductivity replicates; (d) five corresponding updated log conductivity replicates on day 400; (e) concentration on day 400 for the five prior log conductivity replicates; (f) corresponding concentration for the five updated log conductivity replicates.
here transfers computational effort from online to offline operations. When such a transfer is needed to make the ensemble approach feasible, reduced-order modeling should be seriously considered. Otherwise, it may not be necessary.

The decision to use reduced-order modeling also depends on the feasibility of adequately reproducing the dynamic behavior of the true system with a simplified approximation. The reduced-order model used in our example performs well for a particular true system with a dominant high permeability channel but cannot be expected to perform as well over a wider range of conditions unless it is rederived from a new set of snapshot matrices. The success of this process depends on the conditions selected for the snapshot simulations. If these conditions cover too narrow a range of possibilities the reduced-order model will perform well only in certain limited situations. If the range of snapshot simulations is too wide the reduced-order model may not work particularly well for any condition, even one falling within this range.

The approach to reduced-order modeling described here is only one of many possibilities. We selected it largely because it is well suited for our example problem. In other applications it may be preferable to use alternative model reduction techniques such as the DEIM or methods that require online adjustments to the reduced-order approximation [2, 6, 18]. Although there is still much room for experimentation, our research suggests that the basic concept of combining ensemble measurement updating, forecasting, and optimization with reduced-order modeling has much merit and can lead to effective control strategies that are able to deal with uncertainty.

Acknowledgments. We thank Ruben Juanes, Youssef M. Marzouk, and Karen Willcox for their comments.

REFERENCES


