A Multiresolution Methodology for Signal-Level Fusion and Data Assimilation with Applications to Remote Sensing

MICHAEL M. DANIEL, STUDENT MEMBER, IEEE, AND ALAN S. WILLSKY, FELLOW, IEEE

Invited Paper

This paper covers the design of multiscale stochastic models that can be used to fuse measurements of a random field or random process provided at multiple resolutions. Such sensor fusion problems arise in a variety of contexts, including many problems in remote sensing and geophysics. An example, which is used in this paper as a vehicle to illustrate our methodology, is the estimation of variations in hydraulic conductivity as required for the characterization of groundwater flow. Such a problem is typical in that the phenomenon to be estimated cannot be measured at fine scales throughout the region of interest, but instead must be inferred from a combination of measurements of very different types, including point measurements of hydraulic conductivity at irregular collections of points and indirect measurements that provide only coarse and nonlocal information about the conductivity field. Fusion of such disparate and irregular measurement sets is a challenging problem, especially when one includes the objective of producing, in addition to estimates, statistics characterizing the errors in those estimates. In this paper, we show how modeling a random field at multiple resolutions allows for the natural fusion (or assimilation) of measurements that provide information of different types and at different resolutions. The key to our approach is to take advantage of the fast multiscale estimation algorithms that efficiently produce both estimates and error variances even for very large problems. The major innovation required in our case, however, is to extend the modeling of random fields within this framework to accommodate multiresolution measurements. In particular, to take advantage of the fast algorithms that the models in [4] admit, we must be able to model each nonlocal measurement as the measurement of a single variable of the multiresolution model at some appropriate resolution and scale. We describe how this can be done and illustrate its effectiveness for an ill-posed inverse problem in groundwater hydrology.

I. Introduction

In this paper we describe a methodology for the efficient, statistically optimal fusion of measurements of a random process or random field for problems in which the measurement data may be of very different types and, in particular, may convey information about the random phenomenon at very different scales. Problems of this type arise in a variety of contexts, perhaps most notably in remote sensing and geophysical applications, in which spatially distributed random fields are to be estimated for a variety of purposes ranging from the simple production of maps of quantities like rainfall distributions to the estimation of spatial quantities to be used in the analysis of complex geophysical processes like ocean currents and subsurface fluid flow.

Geophysical phenomena such as these are typically not accessible to dense, uniform measurement, and one generally must rely on a variety of measurement sources of very different types in order to obtain enough spatial coverage to produce reliable estimates. Furthermore, while some of these measurements may be taken at individual points in the field—e.g., rain gauges, ocean measurements from ships, measurements of subsurface properties in boreholes—these measurements are typically sparse, irregularly sampled, and inadequate by themselves. Consequently, they must be fused with measurements that are indirect and provide nonlocal measurements of the phenomenon of interest over areas that are not adequately covered by the localized point measurements. These indirect observations are usually of varying resolution. An example sensor fusion problem with multiresolution measurements is the estimation of precipitation, which is used for numerical weather prediction (NWP). Precipitation can be measured with rain gauges, radar sensors, and microwave and infrared satellites. The rain gauges provide point samples of precipitation at select locations, while the infrared satellites provide broad but coarse resolution coverage. Climatologists have long recognized that no single measurement source is sufficient for reliable precipitation estimates, and instead all measurements must be incorpo-

Manuscript received June 17, 1996; revised September 21, 1996. This work was supported by Office of Naval Research Grant N00014-91-J-1004, Advanced Research Projects Agency Grant F49620-93-1-0604, and National Science Foundation Grant 9316624-DMS.

The authors are with the Laboratory for Information and Decision Systems at the Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: cafepre@mit.edu).

Publisher Item Identifier S 0018-9219(97)00637-3.

rated [13], [26]. Another geophysical system requiring the assimilation of heterogeneous measurements is ocean currents. Ocean currents are measured with a variety of sensors, including floating buoys, acoustic travel times, satellite altimetry, and direct and indirect observations of temperature and salinity. While the floating buoys can observe fine-scale fluctuations in the ocean currents, their coverage is limited. More comprehensive coverage, albeit at a coarser resolution and limited to the ocean surface, is given by the satellite data. How to fuse the many different measurements in order to produce the most reliable descriptions of ocean currents is a very active research topic [11].

The application used in this paper to illustrate the multiscale methodology is the estimation of hydraulic conductivity for characterizing groundwater flow. Accurately describing the flow of fluids in the earth's subsurface is important due to the prevalence of contaminated soils in or near groundwater supplies. An accurate description of groundwater flow requires an accurate description of hydraulic conductivity, which is a property of the subsurface geology known to be an important determinant of groundwater flow. Geologic properties like hydraulic conductivity can be measured directly only at select well locations. Indirect observations are supplied by tracer travel times, pump tests, acoustic wave propagation (seismics), and measurements of fluid properties like hydraulic head. These observations differ in spatial resolution and support, and each is related to hydraulic conductivity by a physical equation, i.e., a PDE. As illustrated in Section IV, point samples of hydraulic head are essentially observations of a coarse-scale derivative of hydraulic conductivity and are nonlocal in the sense that each head sample is sensitive to the entire conductivity field to be estimated. Again, no single measurement source can provide a reliable estimate of hydraulic conductivity, and all available measurements should be used.

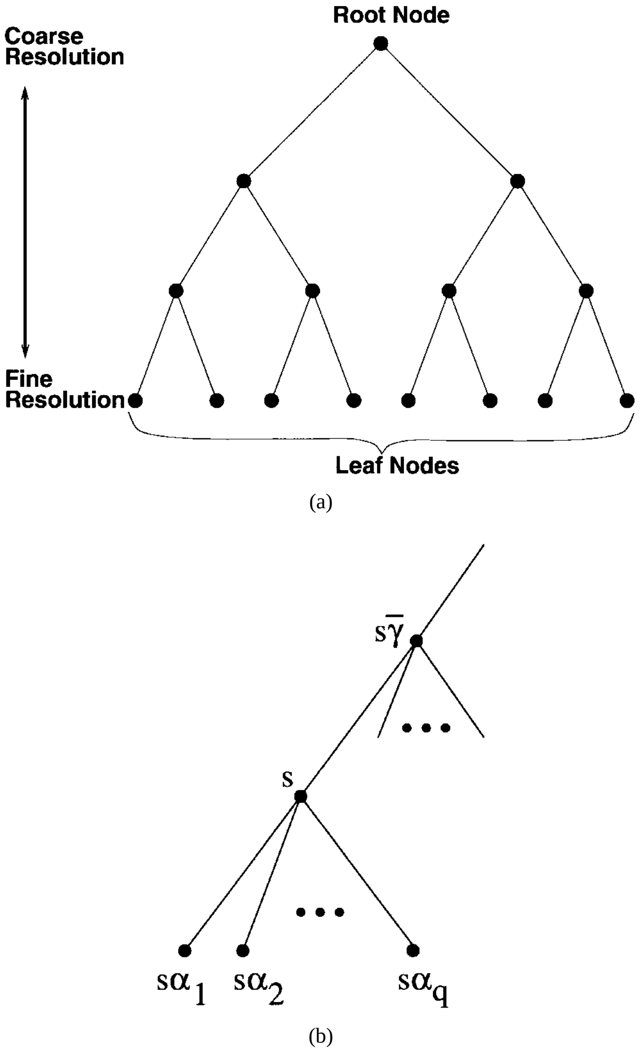
Thus a fundamental objective of our work has been to develop methods for the fusion of such disparate measurement sources, a difficult problem given the nonlocal nature of at least some of the measurement data. Moreover, there are several other features of such geophysical problems that add to the challenge. First, and most importantly, the problems of interest in applications such as these are extremely large, and thus developing computationally efficient algorithms is absolutely crucial. Second, there is generally a strong need for the computation not only of estimates of phenomena but also of error variances for these estimates so that their significance can be assessed. Third, there are often very strong reasons to think about describing phenomena at multiple scales, both because the underlying phenomena in applications such as these generally exhibit variability over wide ranges of scales and also because the available data may support statistically meaningful estimation at different resolutions in different regions, depending on the coverage and nature of the available measurements.

A variety of methods for fusing measurements in such contexts have been used over the years [22], but it is fair to say that computational complexity, especially if error variances are desired, remains a significant and limiting challenge. Several other researchers have attempted to make use of the multiscale nature of the problem by using wavelet decompositions in order to overcome computational limitations, for example [1], [3], [16], and [24]. However these efforts do not address all of the issues of interest here as they either focus only on using wavelets to obtain estimates but not error statistics [1], [3], [16], require regular measurements so that wavelet transforms can be applied [24], or admit only very special nonlocal measurements, namely those that correspond to the explicit direct measurements of wavelet coefficients at particular scales [1]. In contrast, the approach that we develop here computes estimates and error statistics, is directly applicable to arbitrary measurement sets, and allows us to use a wide variety of prior statistical models to describe the statistical variability of the phenomenon.

The basic idea behind our approach is to develop multiscale models for random processes and fields within the class introduced in [4]. These models describe random phenomena using tree structures for which each level of the tree represents a different resolution of the phenomenon of interest. Analogous to one-dimensional (1-D) autoregressive models which evolve recursively in time, these multiscale models evolve recursively in scale. The utility of this class of models is twofold. First, the class has been shown to provide useful models for a wide variety of random processes and fields, such as 1-D Markov processes and two-dimensional (2-D) Markov random fields (MRF's) [19] and self-similar and fractal processes that can be used to model natural phenomena arising in geophysics [8], [9]. Second, and most importantly, just as the Markov property associated with 1-D autoregressive models leads to a highly efficient estimation algorithm, (the Kalman filter), the multiscale models satisfy a Markov property in scale and space which leads to an efficient estimation algorithm. Also, the multiscale estimator automatically, i.e., with no additional computations, produces estimation error covariances. Moreover, the efficiency of this algorithm does not require regular data and in particular can accommodate arbitrarily spaced measurements.

In recent work this model class has met with considerable success, both in demonstrating that very rich classes of stochastic phenomena can be represented within this framework an in applying the associated estimation algorithm to several applications, including computer vision [17] and the optimal interpolation of sea level variations in the North Pacific Ocean from satellite measurements [8]. However, in all of this work, attention has been focused almost exclusively on the finest level of the multiscale representation. That is, in modeling a random phenomenon in this framework the objective has been to ensure that the finest scale of the model has a desired statistical structure. Also, in estimation applications, the measurements that have been considered have all been at the finest level of representation, i.e., they have corresponded to point measurements of the phenomenon. In this context, the variables captured at higher (coarser) levels in the multiscale representation are simply abstract variables that are carried along simply to ensure the statistical structure that allows us to apply the very fast algorithms that these models admit. Nevertheless, these algorithms actually allow measurements and produce estimates at these coarser scales. This suggests that if care were taken to define multiscale models so that these coarser scale variables also represented quantities of interest—specifically nonlocal weighted averages of the phenomenon that correspond to variables that are either measured through indirect measurements or that we wish to estimate—then we would be able to use this same efficient estimation methodology for the fusion of measurements at different resolutions. Achieving this objective, however, is not an obvious or trivial task, and one of the major contributions of this paper is to demonstrate how this can be accomplished. The second contribution is then to use this methodology as the basis for designing extremely efficient and flexible data fusion algorithms that can accommodate measurements at different resolutions and with arbitrary spatial distribution. This capability is demonstrated by estimating hydraulic conductivity from point measurements of conductivity and measurements of head, which provide nonlocal measurements of conductivity through the partial differential equations of groundwater hydrology.

In the next section we provide a brief review the class of multiscale models of [4] and include a description of multiscale models for Markov processes, which will be used throughout this paper to illustrate our results. In Section III, we present a modeling algorithm that begins with a multiresolution model that accurately captures the finest-scale statistics of the phenomenon of interest and augments this model in order to incorporate variables at coarser scales representing nonlocal quantities. We apply this methodology in Sections IV and V and conclude in Section VI.



(a)

(b)

Fig. 1. (a) A binary tree used to index a random process at multiple resolutions and (b) the local labeling of the $q+1$ nodes connected to node s.

II. Multiscale Models

A. Multiscale Models and the Multiscale Estimator

The class of multiscale random processes introduced in [4] is indexed by the nodes of trees organized into scales. The coarsest scale is indexed by the root node, while the finest scale is indexed by the set of leaf nodes. For example, the multiscale process defined on the binary tree illustrated in Fig. 1(a) consists of a set of random vectors $z(s)$ for each node s on the tree. The *scale* of node s, which we denote by $m(s)$, is the distance between node s and the root node of the tree. Define $\bar{\gamma}$ to be the upward (in scale) shift operator, so that $s\bar{\gamma}$ denotes the parent of any node s, as illustrated in (1b). The class of multiscale processes considered in this paper satisfies the following autoregression *in scale*

$$z(s) = A_s z(s\bar{\gamma}) + w(s) \tag{1a}$$

$$w(s) \sim (0, Q_s) \tag{1b}$$

where $z(s)$ is the process value at node s and $z(s\bar{\gamma})$ is the process value at node $s\bar{\gamma}$. The notation $x \sim (m_x, P_x)$ denotes that x is a random vector with mean m_x and covariance P_x.[1] Equation (1) defines an autoregression from coarse to fine scale, with $w(s)$ as the process noise of the autoregression. The autoregression is initialized at the root node $s = 0$ by

$$z(0) \sim (0, P_0). \tag{2}$$

Since $z(0)$ and $w(s)$ are zero-mean, every process value $z(s)$ will be a zero-mean[2] random vector.

The process noise $w(s)$ is assumed to be a white-noise process uncorrelated across scale and space and also uncorrelated with the root node state, i.e., $E[w(s)\, z(0)^T] = 0$. The whiteness of the process noise implies that a multiscale tree model is characterized completely by P_0—the root node covariance—and the autoregression parameters A_s

[1] This paper will focus upon second-order descriptions of random processes and on best linear estimators. Of course, if all variables are Gaussian, this second-order description is also a complete specification and best linear estimators are in fact optimal over the larger class including nonlinear estimators.

[2] The zero-mean assumption is made for simplicity and is easily relaxed by adding a deterministic term to (1a) or (2).

and Q_s for all nodes $s \neq 0$. (A_s and Q_s are not defined for $s = 0$.) More importantly, the whiteness of the process noise leads to a Markov property similar to the Markov property for 1-D autoregressive processes driven by white noise. Specifically, note that any node s, with q_s defined to be the number of children of node s, partitions the tree into $q_s + 1$ subsets of nodes [see Fig. 1(b)]: $\mathcal{S}_{s\alpha_1}, \cdots, \mathcal{S}_{s\alpha_{q_s}}$, and $\mathcal{S}_s^c$, where[3]

$$\begin{aligned} \mathcal{S}_s &\triangleq \text{set of nodes descendent from and including node } s \\ \mathcal{S}_s^c &\triangleq \text{complement of } \mathcal{S}_s \\ s\alpha_i &\triangleq \text{child of node } s, \quad i = 1, \cdots, q_s. \end{aligned}$$

We will also find it useful to write $\mathcal{S}_{s\alpha_{q_s+1}} \triangleq \mathcal{S}_s^c$. The Markov property of multiscale tree processes is that, conditioned on the state $z(s)$, the q_s+1 sets of states partitioned by node s are conditionally uncorrelated. More formally

$$E[z(r)\, z(t)^T \mid z(s)] = E[z(r) \mid z(s)]\, E[z(t) \mid z(s)]^T$$

for all $r \in \mathcal{S}_{s\alpha_i}$, $t \in \mathcal{S}_{s\alpha_j}$, $i \neq j$, and $(i, j) \in [1, q_s + 1] \times [1, q_s + 1]$. Because of this Markov property, the process value $z(s)$ is commonly referred to as the *state* at node s.

The Markov property of the multiscale processes leads to an efficient algorithm for the estimation of $z(\cdot)$ at every node on the tree based upon measurements, each of which is a noise-corrupted observation of $z(\cdot)$ at an individual node of the tree, i.e.,

$$y(s) = C_s z(s) + v(s), \quad v(s) \sim (0, R_s) \tag{3}$$

where $v(\cdot)$ is white and uncorrelated with $z(\cdot)$ at all nodes on the tree. Measurements at coarse-scale nodes will generally be equivalent to measurements of coarse-resolution or nonlocal functions of the finest-scale process. The multiscale estimation algorithm provided in [4] is a generalization of the Kalman filter and Rauch–Tung–Striebel smoother [27] for dynamic systems in time, i.e., processes given by (1) for a tree with $q = 1$. The first sweep of the estimator is a recursion from fine to coarse scale, which is then followed by a recursion from coarse to fine scale. The result is that the linear least-squared error (LLSE) estimate $\hat{z}(s)$ of the state at every node in the tree is computed in $\mathcal{O}(Nd^3)$ computations for a tree which has N nodes[4] and constant state dimension d. Thus the efficiency of the estimator depends primarily upon whether a tree model can be realized with manageable state dimension. As a by-product, the multiresolution estimator also produces the estimation error covariance $E[(z(s) - \hat{z}(s))\,(z(s) - \hat{z}(s))^T]$ at every node.

B. Internal Realizations of Multiscale Models

As defined in [15], an *internal* realization of a multiscale model is one for which each variable of the process is a linear function of the finest-scale process, where the finest-scale process is the states $z(s)$ at the leaf nodes of the tree. If f is a vector containing the finest-scale process of the tree, then each state of an internal realization can be expressed as

$$z(s) = V_s f. \tag{4}$$

Each linear function $V_s f$ will be referred to as an *internal variable* and each matrix V_s as an *internal matrix*. For the example sensor fusion problem given in Section V, the vector f will contain a discretization of the hydraulic conductivity function.

Note that the parameters P_0, A_s, and Q_s of an internal multiscale model can be expressed completely in terms of the internal matrices V_s and the covariance of the finest-scale process, P_f. Specifically, substituting (4) evaluated at $s = 0$ into $P_0 = E[z(0)\, z(0)^T]$ yields

$$P_0 = V_0 P_f V_0{}^T. \tag{5}$$

The parameters A_s and Q_s can then be computed by noting that (1a) is just the optimal prediction of $z(s)$ based upon $z(s\bar{\gamma})$, plus the associated prediction error, i.e.,

$$z(s) = E[z(s) \mid z(s\bar{\gamma})] + w(s). \tag{6}$$

Using standard equations from LLSE estimation, the model parameters follow as

$$A_s = P_{z(s)z(s\bar{\gamma})}\, P_{z(s\bar{\gamma})}^{-1} \tag{7a}$$

$$Q_s = P_{z(s)} - P_{z(s)z(s\bar{\gamma})}\, P_{z(s\bar{\gamma})}^{-1}\, P_{z(s\bar{\gamma})z(s)} \tag{7b}$$

where P_x denotes the covariance of the random vector x, P_{xy} denotes the cross-covariance between x and y, and $P_{x|y}$ denotes the covariance of x after conditioning on y. Finally, the covariances in (7) follow from (4) as

$$P_{z(s)} = V_s P_f V_s{}^T \tag{8a}$$

$$P_{z(s)z(s\bar{\gamma})} = P_{z(s\bar{\gamma})z(s)}^T = V_s P_f V_{s\bar{\gamma}}^T. \tag{8b}$$

Although the algorithm used in Section III assumes that an internal multiscale model is given, it is appropriate to make several comments about these models. The construction of an internal multiscale model for a finest-scale process f with covariance P_f consists of three steps: 1) mapping the components of f to leaf nodes of the tree, which also determines V_s for each of the finest-scale nodes, 2) specifying the internal matrices V_s at the coarser-scale nodes, and 3) computing the model parameters using (5) and (7). Step 1) is generally straightforward, although the mapping can be affected by the nonlocal functions of f to be incorporated at coarser-scale nodes. Also, as we have seen, step 3) is straightforward. Consequently, the core of constructing internal realizations is determining the internal matrices V_s and the resulting covariances in (8).

As discussed in [15], internal multiscale realizations can, in principle, be constructed for a finest-scale random process f with any desired covariance P_f. However, for an arbitrary P_f the ranks of the resulting internal matrices, which equal the dimensions of the corresponding state

[3] The only exceptions are the finest resolution leaf nodes which have no children and the coarsest resolution root node which has no parent.

[4] Note that a tree with N_f nodes nodes at the finest scale has only $\mathcal{O}(N_f)$ nodes, and thus also requires only $\mathcal{O}(N_f d^3)$ computations to be estimated.

vectors $z(s)$, may be quite large and thus negate the computational advantage of the tree model. Fortunately, as developed in [9], [15], [19], there are large classes of processes for which either exact or adequately approximate multiscale realizations can be constructed that have sufficiently low dimension to make the multiscale formalism quite attractive. In the next section, such models for wide-sense Markov processes and MRF's [19] are described and will later be used to illustrate our methodology.

C. Example: Multiscale Models for Markov Processes

A discrete-time process $f[k]$ is a *bilateral* Markov process [7] if, conditioned on the values of $f[k]$ at the boundaries of any interval $I = [k_1, k_2]$, $k_2 > k_1$, the process inside the interval is uncorrelated with $f[k]$ outside the interval. The width of these boundaries depends upon the order, n, of the process. More precisely, define

$$f_b = \{f[k] \mid k \in [k_1 - n, k_1 - 1] \cup [k_2 + 1, k_2 + n]\}$$

to contain $f[k]$ at the boundaries of I for an nth order process. Also define $\tilde{f}[k] = f[k] - E[f[k] \mid f_b]$, which is the process containing the uncertainty in $f[k]$ after conditioning on the boundary values. Then $f[k]$ is said to be nth order bilateral Markov if

$$E[\tilde{f}[l]\,\tilde{f}[m]] = 0 \quad \text{for all } l \in I \text{ and } m \notin I.$$

Similar to the boundary values, define f_p to contain the n present values of $f[k]$, i.e., $f_p = \{f[k], f[k+1], \cdots, f[k+n-1]\}$. An nth order *unilateral* Markov process is one for which

$$E[\check{f}[l]\,\check{f}[m]] = 0 \quad \text{for all} \quad l < k \leq m$$

where $\check{f}[l] = f[l] - E[f[l] \mid f_p]$, i.e., conditioned on the n "present" values of $f[k]$, the past and future are uncorrelated. While not every bilateral Markov process is a unilateral Markov process, every unilateral process is a bilateral [7], so that any method for the multiscale modeling of bilateral Markov processes applies equally well to unilateral Markov processes.

The multiscale models described in [19] are based upon the midpoint deflection algorithm for synthesizing Brownian motion [6]. The basic idea behind the midpoint deflection algorithm is that, given the values of a Markov process at the boundaries of any interval, the midpoint value of this interval can be synthesized independently of any values outside the interval. As an example, consider a first-order Markov process on the interval $[0, N]$. Given $f[0]$ and $f[N]$, then the midpoint value $f[k_0]$, where the "midpoint" k_0 can in fact be anywhere in the interval $[1, N-1]$, can be written as

$$f[k_0] = E[k_0 \mid f_b] + \tilde{f}[k_0] \tag{9a}$$
$$= P_{f[k_0], f_b} P_{f_b}^{-1} f_b + \tilde{f}[k_0] \tag{9b}$$

where $f_b = [f[0]\;\; f[N]]^T$. The second equality in (9) follows from standard LLSE formulas. The covariance matrices in (9b), as well as the covariance of $\tilde{f}[k_0]$, are given by the statistics of the Markov process. Equation (9) can be interpreted as an interpolation from the boundary values plus a deflection $\tilde{f}[k_0]$, as illustrated in Fig. 2(a) for a sample path of Brownian Motion on the interval [0,32].

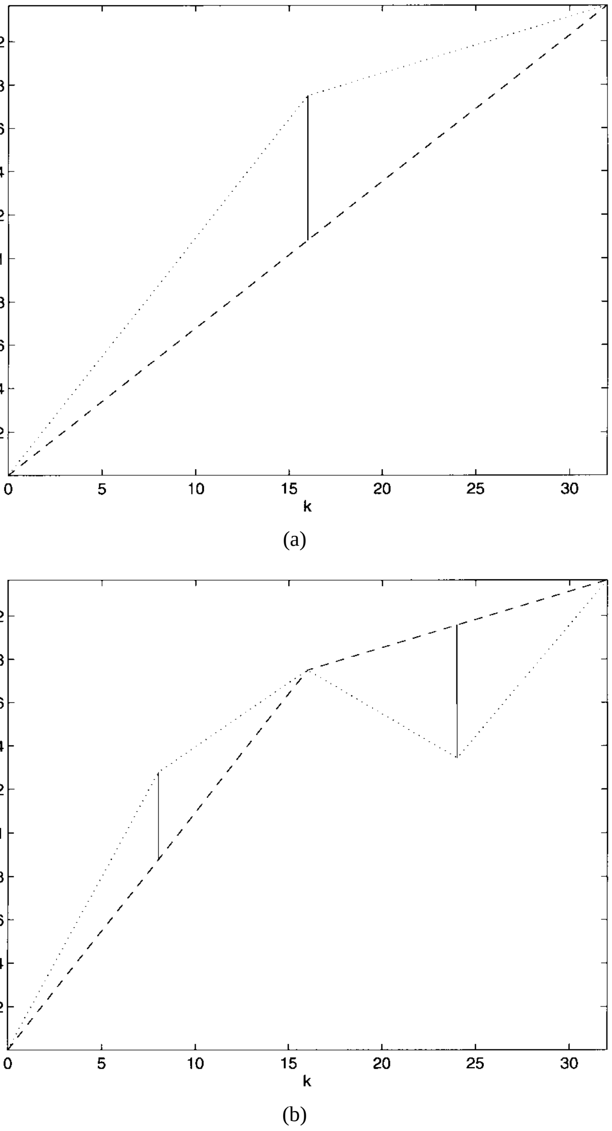


Fig. 2. Steps one (a) and two (b) of the midpoint deflection algorithms for synthesizing Brownian motion. The dashed line provides the interpolation (LLSE estimate) of the process from the present boundary values, the solid line is the deflection of the midpoint(s), and the dotted line is the new interpolation.

Once $f[k_0]$ is determined, we then have the boundary values of the two intervals $I_1 = [0, k_0]$ and $I_2 = [k_0, N]$. The values of the process at the midpoints of these two intervals can again be generated by an interpolation and a deflection, i.e.,

$$f[k_1] = E[f[k_1] \mid f[0], f[k_0]] + \tilde{f}[k_1] \tag{10a}$$
$$f[k_2] = E[f[k_2] \mid f[k_0], f[N]] + \tilde{f}[k_2] \tag{10b}$$

for any "midpoints" $k_1 \in [1, k_0 - 1]$ and $k_2 \in [k_0 + 1, N - 1]$. More importantly, the two deflections $\tilde{f}[k_1]$ and $\tilde{f}[k_2]$ are uncorrelated due to Markovianity, and thus can be generated independently. An example of the interpolation and

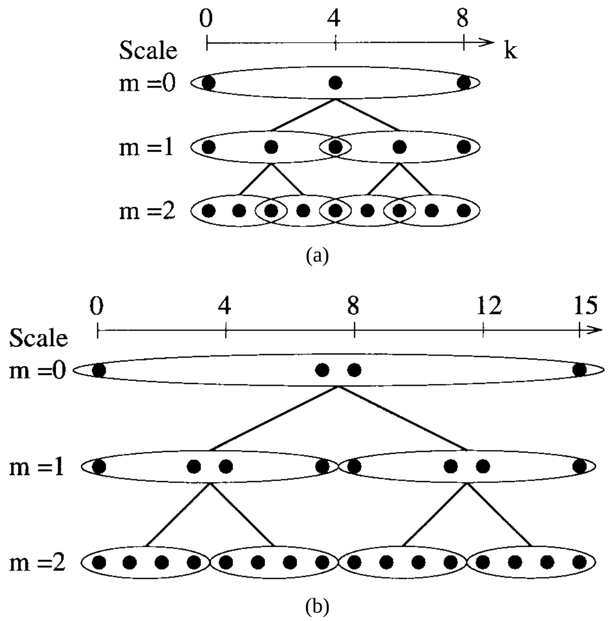


Fig. 3. Multiscale models for first-order Markov processes, where each ellipse represents the samples of the Markov process which comprise the state $z(s)$ at a single node s. For each model, the state $z(s)$ is confined to sample values of the Markov process within the interval I_s. (a) A binary tree with a state dimension of three, and (b) a binary tree with state dimension four.

deflection associated with these two samples is illustrated in Fig. 2(b).

After the first two steps of the midpoint deflection synthesis, $f[k]$ has been computed at the endpoints of four intervals. The synthesis process continues recursively by generating the midpoint values of the four intervals, each of which can be generated independently. In what follows, we describe how this recursive process can be represented by a multiscale autoregression. To simplify notation, assume that $N = 2^M$. Choosing $k_0 = N/2$ as the first midpoint, the state at the root node is given by

$$z(0) = \begin{bmatrix} f[0] \\ f[N/2] \\ f[N] \end{bmatrix}.$$

Modeling the process on a binary tree, the process values at the two descendents of the root node can also be chosen to contain three samples of $f[k]$. Namely, choose

$$z(0\alpha_1) = \begin{bmatrix} f[0] \\ f[N/4] \\ f[N/2] \end{bmatrix} \quad \text{and} \quad z(0\alpha_2) = \begin{bmatrix} f[N/2] \\ f[3N/4] \\ f[N] \end{bmatrix}.$$

The process noise generated when transitioning from scale $m = 0$ to scale one, $w(0\alpha_1)$ and $w(0\alpha_2)$, will contain $\tilde{f}[N/4]$ and $\tilde{f}[3N/4]$, respectively. From Markovianity, these two vectors are uncorrelated with each other, and from the orthogonality of the LLSE they are uncorrelated with $z(0)$. This process can be continued recursively until the variables at a given level of the tree represent the entire interval of interest, as illustrated in Fig. 3(a) for $N = 8$.

The multiscale models for 1-D Markov processes are internal multiscale models, since each state $z(\cdot)$ simply contains samples of $f[k]$. Therefore, the parameters for the scale recursive autoregression follow from (7). However, note that the covariance matrices in (8) can be computed without explicitly computing P_f, the covariance of the entire Markov process which is to be represented at the finest scale of the tree. The ability to compute the model parameters without explicitly forming P_f is especially important for modeling 2-D random fields.

There is considerable flexibility in modeling 1-D Markov processes with the multiscale autoregression. The trees do not have to be binary, and the state dimension can vary from node to node [19]. However, the general procedure for developing an internal multiscale model of a Markov process remains the same, i.e., forming states as samples of subintervals of the Markov process and then deducing the model parameters from (5) and (7). This flexibility also holds in 2-D, where the midpoint deflection algorithm can be generalized to develop internal multiscale models for Markov random fields [19]. A MRF is the 2-D generalization of a 1-D bilateral process. Namely, a wide-sense MRF is a 2-D random process $f[i,j]$ for which the values of f in any connected set Ω are uncorrelated with the values of f outside this set when conditioned on the values of f on the boundary of Ω. Analogous to the multiscale models for 1-D Markov processes, the states of the multiscale models for MRF's contain the values of f on the boundaries of subregions of the entire domain on which f is defined. In other words, if $\{f[i,j] \mid (i,j) \in \Omega_s\}$ is the finest-scale MRF descendent from node s, then $z(s)$ contains the values of $f[i,j]$ on the boundaries of subregions which cover Ω_s. The width of the boundaries again depends upon the order of the MRF. Once these boundaries have been determined, the model parameters can again follow from (5) and (7).

The major difference between the multiscale models for 1-D Markov processes and those for MRF's is that the state dimensions for MRF's grow with the size of the domain of the finest-scale process. The dimension of a state in a multiscale model of 1-D Markov processes depends only upon the order of the process and the number of children, q_s, descending from node s. For 2-D MRF's, the dimension of the state $z(s)$ at a node s corresponding to some 2-D region is proportional to the linear dimension of the boundary of that region. However, in many applications the MRF is only a model of the phenomenon of interest, and thus exactly matching its statistics is unnecessary. Indeed, in [19] low-dimensional multiscale models with approximately the same statistics as MRF's have been used with success.

III. Augmenting the Variables of Internal Multiscale Processes

For internal models, (3) can be rewritten as

$$y(s) = C_s V_s f + v(s). \tag{11}$$

Therefore, given an internal multiscale model, the multiresolution estimator described in [4] can incorporate only measurements of particular linear functions of f, i.e., only functions Gf for which each row of G is in the row space of some internal matrix V_s. For instance, given one of the multiscale models for 1-D Markov processes described in Section II-C, the multiscale estimator cannot incorporate a measurement of the average value of the finest-scale process, since no single state contains the average value of f.

To expand the set of functions represented by the tree model, which will allow the multiscale estimator to fuse multiple-resolution observations of the random process f, one can imagine augmenting the internal variables $V_s f$ with additional linear functions f and then recomputing the model parameters. However, doing this requires considerable care. In particular, the states must be augmented such that both the Markovianity of the tree process is preserved and the resulting model is internal. For example, suppose we naively augment the state of our model at a single node τ in order to include the linear function Gf. That is, suppose $\zeta(s) = z(s)$ for $s \neq \tau$ and

$$\zeta(\tau) = \underbrace{\begin{bmatrix} V_\tau \\ G \end{bmatrix}}_{\mathcal{V}_\tau} f. \tag{12}$$

In general, this augmentation will destroy the Markovianity of the tree. For example, the states $z(\tau\bar{\gamma})$, $z(\tau\alpha_1), \cdots, z(\tau\alpha_q)$ generally are *correlated* with each other after conditioning on $\zeta(\tau)$. The consequences of this correlation are that, for the multiscale model defined by (5) and (7) with $\mathcal{V}_\tau$ substituted for V_τ, the finest-scale process will not have covariance P_f; also, the model will not be internal, i.e., the state at node τ will not be equal to a linear function of the finest-scale process.

The issue here is that the augmentation at node τ introduces some coupling among field values due to the nonlocal nature of the linear function Gf. If the correct statistics are to be maintained, and the state at node τ is to contain the desired function of the finest-scale process, the effect of the coupling must also be propagated to other nodes on the tree.

A. Maintaining the Markov Property of the Internal Variables

For internal realizations, the Markovianity of the variables $V_s f$ guarantees that the finest-scale process of the multiscale model whose parameters are derived from (5) and (7) has the desired covariance [15]. However, as just discussed, augmenting $V_s f$ with additional linear functions of f can destroy this Markov property and thus alter the covariance of the finest-scale process. For instance, consider the multiscale models for 1-D Markov processes described in Section II-C. If the average value of the finest-scale process is added to the state at the root node of the tree, then the deflections $\tilde{f}[k_1]$ and $\tilde{f}[k_2]$ will be correlated. The reason is that the "left" and "right" halves of the finest-scale process are correlated after conditioning on $\zeta(0) = \mathcal{V}_0 f$, (which contains the original state $z(0)$ and the average value of f). This implies that some additional augmentation will be required to account for the correlation.

Instead of directly augmenting $z(\tau)$ in (12) with Gf, another linear function must be found for which 1) the augmented variable $\zeta(\tau)$ still decorrelates the $q_\tau + 1$ sets of internal variables partitioned by node τ, and 2) $\zeta(\tau)$ contains Gf. To understand how this linear function is chosen, it is useful to first examine how the internal variables are chosen in general, i.e., when the finest-scale process is not necessarily a wide-sense Markov process. For internal multiscale realizations of a finest-scale process $f \sim (0, P_f)$, the variables can be assumed to satisfy

$$z(s) = V_s f = W_s f_s \tag{13}$$

where f_s contains the finest-scale process at nodes descendent from s [15]. In other words, the state at node s is a linear function of its descendents. Note that the multiscale models for Markov processes discussed in Section II-C satisfy (13). Equation (13) leads to a simpler form of the Markov property for multiscale trees. Note that, for $i = 1, \cdots, q_s$, each state in the set $\{z(t) \mid t \in \mathcal{S}_{s\alpha_i}\}$ is a linear function only of $f_{s\alpha_i}$. Also, each state in the set $\{z(t) \mid t \in \mathcal{S}_{s\alpha_{q_s+1}}\}$ is a linear function only of $z(s)$ and $f_{s\alpha_{q_s+1}}$, which represents the finest-scale process which does not descend from node s. (When the number of subscripts becomes unwieldy, we will substitute $f_{s^c} \triangleq f_{s\alpha_{q_s+1}}$.) Thus a state $z(s)$ satisfies the multiscale Markov property if and only if it conditionally decorrelates the $q_s + 1$ random vectors $\{f_{s\alpha_i}\}_{1 \le i \le q_s+1}$. Therefore, to augment $z(s)$ with a linear function of f and not alter the covariance of the finest-scale process, we need only ensure that the $q_s + 1$ vectors $\{f_{s\alpha_i}\}_{1 \le i \le q_s+1}$ remain conditionally uncorrelated. To do so we make use of the following corollary of Proposition 5 in [15, Ch. 3].

Corollary: If the $q_s + 1$ vectors $\{f_{s\alpha_i}\}_{1 \le i \le q_s+1}$ are uncorrelated after conditioning on some linear function $V_s f$, they remain uncorrelated after conditioning on $V_s f$ and individual linear functions of $f_{s\alpha_i}$, $i = 1, \cdots, q_s + 1$.

For example, the $q_s + 1$ vectors $\{f_{s\alpha_i}\}_{1 \le i \le q_s+1}$ are uncorrelated after conditioning on $V_s f$, $L_1 f_{s\alpha_1}$, and $L_2 f_{s\alpha_2}$, but they will generally be correlated after conditioning upon $V_s f$ and $L[f_{s\alpha_1}^T, \ f_{s\alpha_2}^T]^T$. Therefore, to add the linear functional $\langle g, f \rangle \triangleq g^T f$ to $z(\tau)$, we first define the following matrix:

$$G_\tau = \begin{bmatrix} g_{\tau\alpha_1}^T & 0 & \cdots & 0 & 0 \\ 0 & g_{\tau\alpha_2}^T & 0 & & 0 \\ \vdots & 0 & \ddots & \ddots & \vdots \\ 0 & & \ddots & g_{\tau\alpha_{q_\tau}}^T & 0 \\ 0 & 0 & \cdots & 0 & g_{\tau\alpha_{q_\tau+1}}^T \end{bmatrix} \tag{14a}$$

where $\langle g_{\tau\alpha_i}, f_{\tau\alpha_i} \rangle$ is the component of $\langle g, f \rangle$ which is only a function $f_{\tau\alpha_i}$, i.e., the sum of all the elements in $G_\tau f$ equals $\langle g, f \rangle$. The variable at node τ can now be augmented as

$$\zeta(\tau) = \underbrace{\begin{bmatrix} W_\tau & 0 \\ \multicolumn{2}{c}{G_\tau} \end{bmatrix}}_{\mathcal{V}_\tau} \underbrace{\begin{bmatrix} f_\tau \\ f_{\tau^c} \end{bmatrix}}_{f} \tag{14b}$$

without altering the Markov property. Note that if g has full support, i.e., if each term $\langle g_{\tau\alpha_i}, f_{\tau\alpha_i}\rangle \not\equiv 0$, then this augmentation requires an additional $q_\tau + 1$ elements in the state at node τ. If some of these terms are zero, then a lower-dimensional augmentation is required. Furthermore, if any of the rows of G_τ are already in the row-space of $[W_\tau \;\, 0]$, these elements are already available in $z(\tau)$ and need not be added. Note also that since the partitioning of f into $f_{\tau\alpha_i}$ is different for each node, one might imagine that there is a best choice for node τ in terms of minimizing the number of terms $\langle g_{\tau\alpha_i}, f_{\tau\alpha_i}\rangle$ which are nonzero and hence minimizing the dimension of the augmentation.

Define the augmented variable at each node by $\zeta(s) = \mathcal{V}_s f$, where $\zeta(s) = z(s)$ if the state at node s is not augmented. The model parameters of the augmented model follow as

$$P_0 = \mathcal{V}_0 P_f \mathcal{V}_0^T \tag{15a}$$

$$A_s = P_{\zeta(s)\zeta(s\bar{\gamma})}\, P^{-1}_{\zeta(s\bar{\gamma})} \tag{15b}$$

$$Q_s = P_{\zeta(s)} - P_{\zeta(s)\zeta(s\bar{\gamma})}\, P^{-1}_{\zeta(s\bar{\gamma})}\, P_{\zeta(s\bar{\gamma})\zeta(s)} \tag{15c}$$

where $P_{\zeta(s)} = \mathcal{V}_s P_f \mathcal{V}_s^T$ and $P_{\zeta(s)\zeta(s\bar{\gamma})} = \mathcal{V}_s P_f \mathcal{V}_{s\bar{\gamma}}^T$. This augmented model will have a finest-scale process with covariance identical to that of the original model.

B. Maintaining an Internal Multiscale Model

The augmentation described in the preceding section, which augments the state at a single node τ, does maintain Markovianity and hence yields a model whose finest-scale process will have the desired covariance P_f. However, this model will not generally be consistent; that is, the element of the state at node τ that is intended to equal $\langle g_{\tau\alpha_i}, f_{\tau\alpha_i}\rangle$ may not be equal to $\langle g_{\tau\alpha_i}, f_{\tau\alpha_i}\rangle$. The reason for this is simple: at node τ we are attempting to *pin* a linear combination of the values descending from node $\tau\alpha_i$. In order to ensure that this value is pinned, information must be propagated from node τ all the way to its descendents at the finest scale. To illustrate this problem and to motivate its solution, consider the following example of augmenting the root node of a multiscale model for a 1-D Markov process with the sum of the finest-scale process.

1) Example: Multiscale Modeling the Sum of a 1-D Markov Process: Consider a 1-D first-order Markov process $f[k]$ on the interval [0,15]. The multiscale model for this process is illustrated in Fig. 3(b). Assume that the 1-D Markov process is to be estimated, using the multiscale estimator, from point measurements of $f[k]$ together with a measurement of the sum $h = (\sum_{k=0}^{15} f[k])$ of the finest-scale process. From Section III-A, we know that $z(0)$ can be augmented with the two linear functions

$$h_1 = \sum_{k=0}^{7} f[k] \quad \text{and} \quad h_2 = \sum_{k=8}^{15} f[k] \tag{16}$$

without altering the Markov property of the tree. Thus if the root node variable is augmented as $\zeta(0) = [z(0)^T,\ h_1,\ h_2]^T$ and no other variables are changed, then the finest-scale process of the multiscale model derived from (15) will have the covariance of the 1-D Markov process.

However, the element in the state at the root node which is intended to contain $h = h_1 + h_2$ will not be equal to the sum of the finest-scale process unless this value is propagated from the root node to the finest-scale. This propagation is accomplished by constraining the scale-to-scale recursion of the multiscale model. For this 1-D Markov example, this means constraining the midpoint deflections by conditioning them on the value of h generated at the root node. This conditioning is accomplished by augmenting the descendents of the root node, except for the finest-scale states which are never augmented, with h. Again, this augmentation must also preserve Markovianity. For example, consider the two children of the root node, nodes $0\alpha_1$ and $0\alpha_2$. The augmentation of these nodes is shown in (17) at the bottom of the page. However, these states contain more information than is needed. For instance, $f[k]$ on the interval [0,7] is uncorrelated with $f[k]$ on the interval [8,15] when conditioned on $z(0\alpha_1)$. Thus the last element of $\zeta(0\alpha_1)$ in (17) contains no additional information about the descendents of nodes $0\alpha_1$. That is, in order to maintain consistency, and hence an internal realization, the state at node $0\alpha_1$ must only be made consistent with h_1, the component of h corresponding to the finest-scale descendents of node $0\alpha_1$. Similarly, the state at node $0\alpha_2$ must only be made consistent with h_2. As a result, the states in (17) can be reduced to (18), as shown on the bottom of the previous page. For this simple example,

$$\zeta(0\alpha_1) = \begin{bmatrix} z(0\alpha_1) \\ \sum_{k=0}^{3} f[k] \\ \sum_{k=4}^{7} f[k] \\ \sum_{k=8}^{15} f[k] \end{bmatrix} \quad \text{and} \quad \zeta(0\alpha_2) = \begin{bmatrix} z(0\alpha_2) \\ \sum_{k=8}^{11} f[k] \\ \sum_{k=12}^{15} f[k] \\ \sum_{k=0}^{7} f[k] \end{bmatrix} \tag{17}$$

$$\zeta(0\alpha_1) = \begin{bmatrix} z(0\alpha_1) \\ \sum_{k=0}^{3} f[k] \\ \sum_{k=4}^{7} f[k] \end{bmatrix} \quad \text{and} \quad \zeta(0\alpha_2) = \begin{bmatrix} z(0\alpha_2) \\ \sum_{k=8}^{11} f[k] \\ \sum_{k=12}^{15} f[k] \end{bmatrix} \tag{18}$$

the augmentation is now complete, and the parameters of the augmented multiscale model can now be derived from (15). The resulting model generates a finest-scale process with the desired covariance P_f and is internally consistent so that, for example, the value of $h_1 + h_2$ at the root node does exactly equal the sum of the finest-scale process.

2) Example: Modeling the Sum at Scale One: Although the sum of the finest-scale process is a function of the entire finest-scale process, it can be advantageous, as shown in the examples of Section V-A, to model this value at a node other than the root node. Consider augmenting the variable at $0\alpha_1$ with h. The augmented variable which preserves the Markov property of $z(0\alpha_1)$ is

$$\zeta(0\alpha_1) = \begin{bmatrix} z(0\alpha_1) \\ \sum_{k=0}^{3} f[k] \\ \sum_{k=4}^{7} f[k] \\ \sum_{k=8}^{15} f[k] \end{bmatrix}. \tag{19}$$

While, as argued in the previous example, the last element of $\zeta(0\alpha_1)$, namely h_2, is unnecessary to maintain Markovianity or consistency with the nodes descending from node $0\alpha_1$, it is necessary if the h is to be captured at this node.

To maintain consistency, the information contained in h_2 at node $0\alpha_1$ must be propagated to the other half of the tree, i.e., that descending from node $0\alpha_2$. To accomplish this, it is necessary that the value of h_2 be available to this part of the process; therefore, the root node must be augmented as $\zeta(0)^T = [z(0)^T, \sum_{k=8}^{15} f[k]]$. The state $z(0\alpha_2)$ can be augmented exactly as in (18). Note that because we have chosen to place the measurement at node $0\alpha_1$, $\zeta(0)$ does not need to include the sum over the "left" half of the tree, (as we do not introduce the constraint on this sum at the root node). Thus in comparison to the augmented model in the preceding subsection, in this case the dimension of $\zeta(0)$ has been reduced while that of $\zeta(0\alpha_1)$ has been increased. The remaining states are identical in the two models. As before, having defined the states, the model parameters can be generated from (15).

C. An Algorithm for Augmenting Internal Multiscale Realizations for a Single Nonlocal Measurement

Using the previous two examples for intuition, we now present a general algorithm for adding linear functions of f to the coarser-scale variables of internal multiscale models. This algorithm applies to a much broader class of processes than those discussed in the previous section. The multiscale model can have an arbitrary number of children per node and the finest-scale process can have any desired covariance—not just that of 1-D Markov process. The algorithm proceeds in two stages: 1) the augmented variables $\zeta(\cdot)$ are created for each node on the tree, and 2) the model parameters are computed from (15) for the augmented process $z^a(\cdot)$.

The algorithm which follows is for adding a single linear functional $\langle g, f\rangle$ to the variable at node τ. This procedure can then be applied recursively to add additional linear functions. The initial step is to determine $\zeta(\tau)$. As discussed in Section III-A, the augmented variable which preserves the Markov property of $z(\tau)$ is given by (14). The next step is to define $\zeta(\cdot)$ for the remaining nodes in the tree to guarantee that the information generated by $z^a(\tau)$ is passed consistently to the finest-scale process. First consider the nodes descendent from node τ. Since all the descendents of node τ are linear functions of f_τ, the entire process descendent from node τ is uncorrelated with f_{τ^c} when conditioned on $z(\tau)$. Thus augmenting any variable descendent from node τ with a linear function of f_{τ^c} will have no effect upon the parameters derived from (15). Consequently, since the linear function $\langle g, f\rangle$ can be decomposed as

$$\langle g, f\rangle = \langle g_\tau, f_\tau\rangle + \langle g_{\tau^c}, f_{\tau^c}\rangle \tag{20}$$

the variables descendent from node τ only need to be made consistent with $\langle g_\tau, f_\tau\rangle$. In fact, because of the conditioning property of $z(s)$, (where s is a descendent of τ), the augmented variable only needs to include $\langle g_s, f_s\rangle$. This augmentation will guarantee that all the process noise added to the descendents of node τ is conditioned on $\langle g, f\rangle$. Therefore, the augmentation of $z(s)$ which preserves Markovianity and maintains consistency is

$$\zeta(s) = \begin{bmatrix} W_s \\ G_s \end{bmatrix} f_s = \mathcal{V}_s f \tag{21a}$$

and

$$G_s = \begin{bmatrix} g_{s\alpha_1}^T & 0 & \cdots & 0 \\ 0 & g_{s\alpha_2}^T & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & g_{s\alpha_{q_s}}^T \end{bmatrix}. \tag{21b}$$

Now consider determining $\zeta(\cdot)$ for nodes not in $\mathcal{S}_\tau$. These nodes must be augmented to make the value of $\langle g_{\tau^c}, f_{\tau^c}\rangle$ consistent with the finest-scale process f_{τ^c}. However, if the support of g is not the entire domain, we may only need to augment a subset of the nodes in $\mathcal{S}_\tau^c$. Specifically, define the direct ancestors of τ as $\tau\bar{\gamma}$, $\tau\bar{\gamma}^2, \cdots$, and let σ be the ancestor closest to node τ for which

$$g^T f = g_\sigma^T f_\sigma. \tag{22}$$

Only nodes descendent from node σ need to be augmented, since, conditioned on $z(\sigma)$, the variables at any node outside the subtree descending from σ are uncorrelated with f_σ and hence with $g^T f$. Consider first the augmentation of a node $s \neq \tau$ on the path connecting τ and σ, (i.e., s is a direct ancestor of τ that is either node σ or a descendent of σ). As always, $\langle g, f\rangle$ can be expressed as

$$\begin{aligned}\langle g, f\rangle = {} & \langle g_{s\alpha_1}, f_{s\alpha_1}\rangle + \langle g_{s\alpha_2}, f_{s\alpha_2}\rangle \\ & + \cdots + \langle g_{s\alpha_{q_s}}, f_{s\alpha_{q_s}}\rangle + \langle g_{s^c}, f_{s^c}\rangle.\end{aligned} \tag{23}$$

While $\langle g_{s^c}, f_{s^c}\rangle$ is not needed at node s to maintain Markovianity, it must be included in $\zeta(s)$ to ensure that this value is passed to the state at node τ. This is a generalization of the description of $\zeta(0\alpha_1)$ in (19), for which the last component of the state was not required for Markovianity but was needed to have the entire linear functional available at node τ. In the more general case here, the last component is needed to have the entire linear functional available at a *descendent* of node s (namely, node τ).

Turning to the other q_s components in (23), all but one must be included in $\zeta(s)$. This component corresponds to the child $s\alpha_i$ of node s for which $\tau \in \mathcal{S}_{s\alpha_i}$, i.e., the child of node s that is either node τ itself or a direct ancestor of node τ. This component can be excluded without disturbing Markovianity or consistency and can be generated at a descendent of node s. This is a generalization of the augmentation of node zero given in Section III-B2), where $z(0)$ only needs to be augmented with $h_2 = \sum_{k=8}^{15} f[k]$ and not with $h_1 = \sum_{k=0}^{7} f[k]$.

The augmented variable $\zeta(s)$ is then given by

$$\zeta(s) = \underbrace{\begin{bmatrix} W_s & 0 \\ & G_s \end{bmatrix}}_{\mathcal{V}_s} \underbrace{\begin{bmatrix} f_s \\ f_{s^c} \end{bmatrix}}_{f} \tag{24a}$$

where the elements of $G_s f$ correspond to all of the elements on the right-hand side of (23) except the one not needed for the augmentation. For example, if $i = 1$ is the term not needed for the augmentation and if the elements of f_s are organized as $f_s^T = [f_{s\alpha_1}^T, f_{s\alpha_2}^T, \cdots, f_{s\alpha_{q_s}}^T]^T$, then

$$G_s = \begin{bmatrix} 0 & g_{s\alpha_2}^T & 0 & \cdots & & 0 \\ 0 & 0 & g_{s\alpha_3}^T & 0 & \cdots & 0 \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ 0 & & & 0 & g_{s\alpha_{q_s}}^T & 0 \\ 0 & 0 & \cdots & & 0 & g_{s^c}^T \end{bmatrix}. \tag{24b}$$

Finally, once we have augmented each of the direct ancestors of τ up to and including σ, the descendents of these nodes must also be augmented. This is exactly the same procedure used to augment the descendents of node τ, i.e., each state is augmented with those elements necessary to maintain both Markovianity and internal consistency. The resulting overall algorithm for state augmentation can then be summarized as follows: for each node $s \in \mathcal{S}_\sigma$ and s not at the finest scale

1) If $s = \tau$, then $\zeta(\tau)$ is given by (14).
2) If $s \neq \tau$ and s is on the path from σ to τ, then $\zeta(s)$ is given by (14).
3) Otherwise, $\zeta(s)$ is given by (21).

For $s \notin \mathcal{S}_\sigma$ or s at the finest scale, then $\zeta(s) = z(s)$. Note that if $\tau = \sigma$, i.e., if the linear function placed at node τ is a function only of f_τ, then the augmentation is simplified. Namely, $\zeta(s)$ is given by (21) for $s \in \mathcal{S}_\tau$, and $\zeta(s) = z(s)$ for $s \notin \mathcal{S}_\tau$.

Once the matrices $\mathcal{V}_s$ have been determined, the final step of the augmentation algorithm is to compute the model parameters from (15). Given the parameters of the original multiscale model, only the parameters for $s \in \{\mathcal{S}_\sigma, \sigma\bar{\gamma}\}$ need to be recomputed for the augmented model.

D. Adding Multiple Nonlocal Measurements

For adding linear functions of f, i.e., multiple linear functionals, to the state at node τ or any other node of a multiscale tree, the state augmentation just described can be applied recursively to individual linear functionals. This recursive procedure can be used to represent measurements of f at multiple resolutions within the tree framework. Note, however, that (15) need only be executed once, after all the linear functionals have been incorporated into the augmented states $\zeta(\cdot)$.

If the states $z(s) = W_s f_s$ of the original multiscale model contain nontrivial functions of the process f or if the states of an internal realization are augmented with a large number of linear functionals, it is likely that that the state dimensions can be reduced before executing (15). Specifically, if any of the internal matrices $\mathcal{V}_s$ have linearly dependent rows, the corresponding states can be compressed by discarding redundant variables. Also, while the matrix $\mathcal{V}_s$ may not have precise algebraic dependency, it may have approximate probabilistic dependency. In other words, $P_{\zeta(s)} = \mathcal{V}_s P_f \mathcal{V}_s^T$ may be singular or ill-conditioned, corresponding to some parts of $\zeta(s)$ being exactly or nearly deterministic (and therefore zero since the processes are assumed to be zero mean). Consequently, an eigenvalue decomposition of $P_{\zeta(s)}$ can identify reduced-order state models that yield good approximations to the desired statistics. In any case, if $\zeta(s)$ has an ill-conditioned covariance matrix, $\zeta(s)$ must be replaced by a reduced-order vector $L\zeta(s)$ with well-conditioned covariance before executing (15).

E. Comments on Computational Complexity

The utility of the multiscale framework is the ability to efficiently provide statistical analysis in the form of optimal estimates and error covariances. This efficiency hinges upon the number of computations required to derive the model parameters from (15) and the number of computations required to implement the corresponding multiscale estimator. Because P_f cannot be explicitly stored in memory when f has very large dimension, the primary obstacle to implementing (15) is the computation of the covariance matrices $P_{\zeta(s)}$ and $P_{\zeta(s)\zeta(s\bar{\gamma})}$. For instance, if the finest-scale process corresponds to a 2-D field of 64-by-64 elements, P_f has approximately 17 million elements, making it infeasible to derive $P_{\zeta(s)}$ directly from P_f and $\mathcal{V}_s$. Instead, for such applications $P_{\zeta(s)}$ and $P_{\zeta(s)\zeta(s\bar{\gamma})}$ must be computed implicitly. This implicit computation is straightforward for stationary random fields, such as the MRF's used in Section V-B.

Assuming the model parameters can be computed efficiently, the remaining question is how the state augmentations described in the previous section affect the computational efficiency of multiscale estimator. Remember that the number of computations required by the multiscale estimator increase cubicly with the dimension of each state of the tree. For each linear functional $\langle g, f\rangle$ placed at node τ by the algorithm of Section III-C, the state

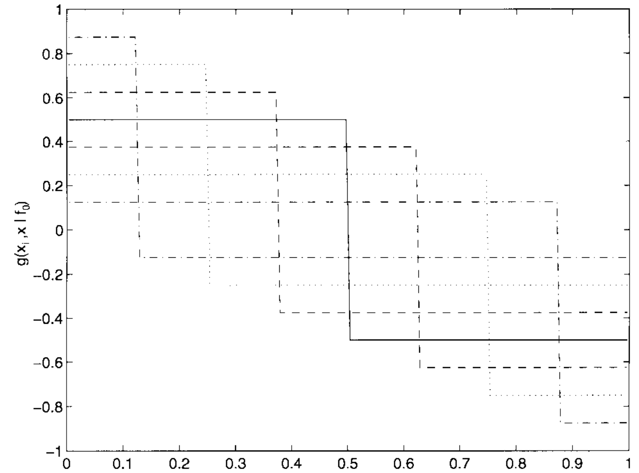


Fig. 4. The Fréchét derivatives $g(\underline{x}_i, \underline{x} \mid f_0)$ at $\underline{x}_i = i/8$, $i = 1, \cdots, 7$, for the 1-D flow equation when linearized about the log conductivity function $f_0 = 0$.

at each node s in the subtree descending from τ will increase by q_s elements.[5] While the effect of this increase is insignificant when adding a single linear functional of f, the effect will be problematic when a large number of linear functionals must be added. Therefore, an important problem is to manage the dimension of the states in the augmented multiscale model. As mentioned in the previous subsection, the states dimensions can be reduced whenever the augmented variables $\zeta(s)$ have ill-conditioned covariance. Furthermore, if one is willing to accept models which only approximate the desired statistics, in many cases the state dimensions can be significantly reduced without significantly altering the accuracy of the estimates. While a complete framework for approximate modeling is not described in this paper, other methods for managing the state dimension are discussed in the examples.

IV. Hydraulic Head as a Coarse-Resolution Function of Conductivity

The application which provided the original motivation for this work [5], and which will be explored in Sections V-A and B, is the estimation of parameters in groundwater flow models. Accurate descriptions of groundwater flow are very important due to the prevalence of contaminated soils in or near groundwater supplies. A standard model for steady-state groundwater flow, assuming no internal water sources, is [20]

$$-\nabla \cdot (e^{f(\underline{x})} \nabla h(\underline{x})) = 0, \quad \underline{x} \in \Omega \tag{25}$$

where f is log hydraulic conductivity,[6] h is piezometric head, and Ω is the region of interest. Piezometric head is a potential, in direct analogy with electrostatic potential.

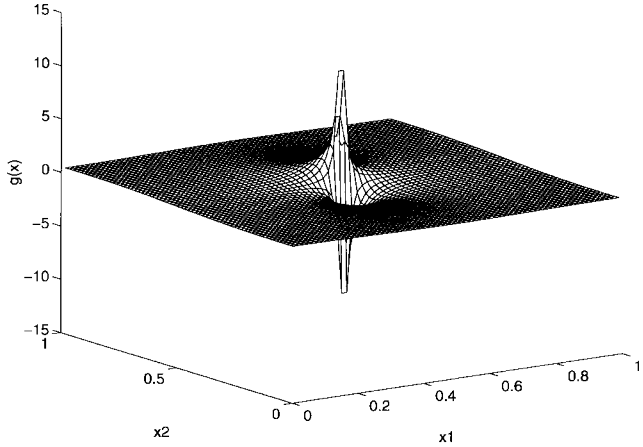


Fig. 5. The Fréchét derivatives $g(\underline{x}_i, \underline{x} \mid f_0)$ for the 2-D flow equation at $\underline{x}_i = (0.5, 0.5)$ when linearized about the log conductivity function $f_0 = 0$.

The problem for groundwater hydrologists is to estimate flow parameters like the function $f(\underline{x})$ from irregularly distributed measurements of conductivity, head, contaminant concentrations, tracer tests, and the like, all providing observations of $f(\underline{x})$ at different resolutions [10], [14], [20], [22], [25]. For simplicity, we consider using the multiscale framework for the fusion of measurements at two resolutions—point measurements of head and hydraulic conductivity. We also assume that the boundary conditions of the flow model are known. These assumptions serve only to simplify the analysis, and do not reflect limitations of our approach.

Note each sample (point value) of head is a nonlinear and nonlocal function of the entire hydraulic conductivity function. Because the multiscale estimator is only able to incorporate linear measurements of the unknown process $f(\underline{x})$, the head measurements must be linearized. This linearization is given by computing the Fréchét derivative of each observed head sample $h(\underline{x}_i)$ with respect to the entire conductivity function on the domain Ω. The value of $h(\underline{x}_i)$ based upon the linearization about the conductivity function $f_0(\underline{x})$ is

$$h(\underline{x}_i) \approx h_0(\underline{x}_i) + \int_\Omega g(\underline{x}_i, \underline{x} \mid f_0)(f(\underline{x}) - f_0(\underline{x}))d\underline{x} \tag{26}$$

where $g(\underline{x}_i, \underline{x} \mid f_0)$ is the Fréchét derivative and h_0 is the solution to (25) when $f = f_0$. The Fréchét derivative is given by [22]

$$g(\underline{x}_i, \underline{x} \mid f_0) = -e^{f_0(\underline{x})}(\nabla h_0(\underline{x}) \cdot \nabla G(\underline{x}_i, \underline{x} \mid f_0)) \tag{27}$$

where $G(\underline{x}_i, \underline{x} \mid f_0)$ is the Green's function [12] of (25) for $f = f_0$. If y_i is the initial noisy measurement of head at $\underline{x}_i$, the linearized head measurement is given by

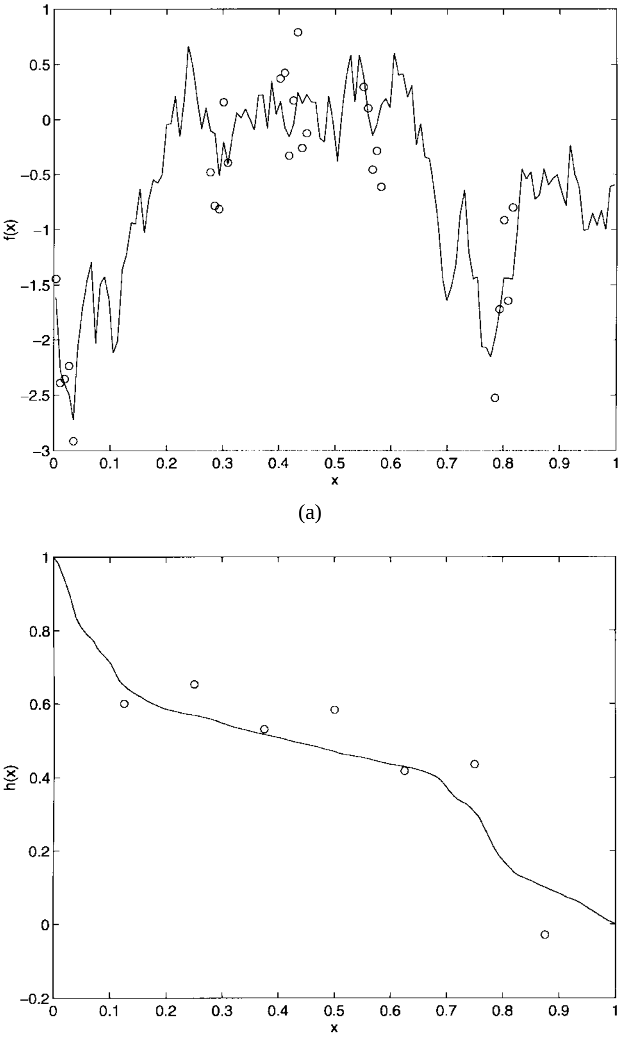
$$y_i - h_0(\underline{x}_i) = \int_\Omega g(\underline{x}_i, \underline{x} \mid f_0)(f(\underline{x}) - f_0(\underline{x}))d\underline{x} + v_i \tag{28}$$

where the measurement noise v_i also includes errors in the linearization. When discretized, (28) can be expressed in the form of (11).

For the constant background conductivity $f_0 = 0$ and boundary conditions $h(0) = 1$ and $h(1) = 0$, the Fréchét derivatives for the 1-D flow equation are illustrated in

[5] If τ is a descendent of σ, where σ is defined in Section III-C, then the only difference is that the state dimension at node τ increases by $q_\tau + 1$ and the state dimension at node σ increases by $q_\sigma - 1$.

[6] Because hydraulic conductivity is often log-normally distributed [22], it is often easier to work directly with the logarithm of hydraulic conductivity.



(a)

(b)

Fig. 6. (a) A sample path of the log conductivity function and (b) the corresponding head function. The noisy measurements are indicated by o's.

Fig. 4. Note that the Fréchét kernels are nonzero over the entire domain, indicating that each head measurement is sensitive to the entire conductivity function f. A Fréchét derivative for the 2-D flow equation is illustrated in Fig. 5 for $f_0 = 0$. Define $\underline{x} = (x_1, x_2)$ to be the 2-D spatial coordinate. The boundary conditions assumed in this case are that $h = 1$ along $x_2 = 1$ and $h = 0$ along $x_2 = 0$, and that the flux normal to the boundaries $x_1 = 0$ and $x_1 = 1$ is equal to zero. The 2-D Fréchét derivative indicates that $h(\underline{x}_i)$ is more sensitive to local conductivity values in 2-D than in 1-D. However, both the 1-D and 2-D Fréchét derivatives illustrate that head samples essentially measure a coarse-scale derivative of f.

V. Applying the Multiscale Framework to Groundwater Flow

In this section,[7] the multiscale estimator is used to fuse measurements of log conductivity and head into an

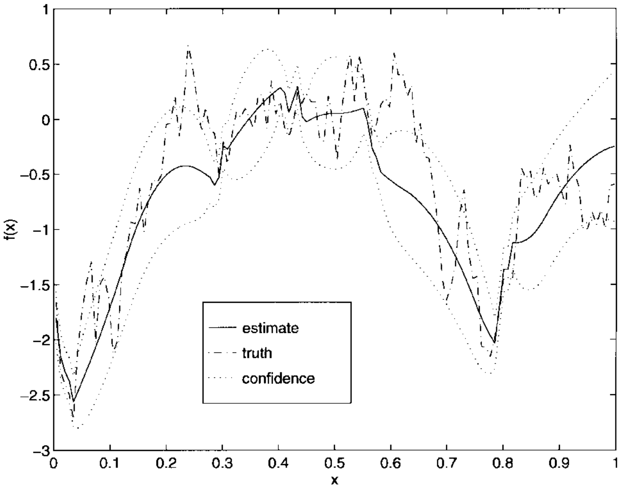


Fig. 7. LLSE estimate of the log conductivity function (solid line) along with the one standard deviation confidence intervals (dotted lines). The true conductivity function is also provided (dashed line).

estimate of the log conductivity function. The log conductivity function is assumed to be wide-sense Markov, and thus can be realized using the multiscale models of Section II-C. (Remember that the class of multiscale models is *not* restricted to having a Markov process or MRF at the finest scale.) The following examples serve three purposes. First, we demonstrate the utility of the multiscale framework for data fusion problems. Second, we provide some enhancements to the algorithm of Section III-C for managing the growth in state dimension due to the addition of nonlocal measurements. Third, the conductivity estimates and the corresponding error variances provide some insight into the problems encountered in automatic flow modeling. In particular, we demonstrate why the incorporation of additional measurements like tracer travel times and pump tests is necessary for producing reliable estimates of hydraulic conductivity.

A. One-Dimensional Flow

For steady-state flow in 1-D, consider estimating log conductivity on the interval $x \in [0, 1]$. Assume that $f(x)$ is a 1-D first-order Markov process with zero mean and covariance

$$E[f(x)f(x+r)] = e^{-5|r|}. \tag{29}$$

Samples of this 1-D Markov process can be mapped to the finest scale of one of the multiscale models described in Section II-C. In particular, assume a binary tree with six scales, four samples per state, and $N = 128$ elements at the finest scale. [The model for $N = 16$ is illustrated in Fig. 3(b).] A sample path of f and the corresponding head function are illustrated in Fig. 6, along with the noisy point measurements. The Fréchét derivatives of the seven head measurements are illustrated in Fig. 4, i.e., the head samples are linearized about the mean of the conductivity function, $m_f = 0$.

[7] The MATLAB code used to generate the examples in this section can be obtained by anonymous ftp at the Website lids.mit.edu in the directory pub/ssg/code/groundwater.

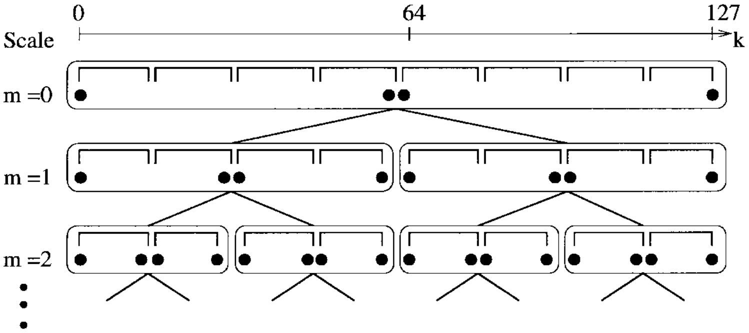


Fig. 8. The states of the first three scales of the multiscale model for a 1-D Markov process after the inclusion of the seven linear functionals illustrated in Fig. 4. The brackets in each state represent local averages of the finest scale process.

The variables of the multiscale representation of the 1-D Markov process can be augmented so that all of the head measurements are modeled at the root node of the tree. The estimate $\hat{f}(x)$ of the finest-scale process of the multiscale model is given in Fig. 7. The multiscale estimator also computes the estimation estimation error variances $E[(\hat{f}(x) - f(x))^2]$, which are included in Fig. 7 in the form of confidence intervals. The confidence intervals are equal to the LLSE estimate plus or minus a single standard deviation of the estimation error. As would be expected, most of the true conductivity function lies within the confidence interval.

A closer look at the state augmentation for incorporating the head measurements illustrates how the state dimensions of the augmented model can be reduced, even when no approximations are made. Assume that all seven head measurements are placed at the root node of the tree by recursively applying the algorithm of Section III-C. The seven head measurements are represented by the inner products $\langle g^i, f\rangle$, where $\langle g^i, f\rangle$ is the 128 sample Riemann sum approximation of $\int_{x=0}^{1} g(x_i, x \mid f_0) f(x) dx$. Using a naive application of the augmentation algorithm, the dimension of each state of the multiscale tree will increase by 14. However, all of these dimensions can be reduced. To see this, first note that the seven Fréchét derivatives can be represented by linear combinations of the eight local averages

$$a_i = \int_{(i-1)/8}^{i/8} f(x) dx, \quad i = 1, \cdots, 8.$$

The state $z(0)$ can be augmented with each of these local averages without destroying the Markov property of this state. Secondly, $z(0\alpha_1)$ only needs to be augmented with $a_1, \cdots, a_4$, and $z(0\alpha_2)$ only needs to be augmented with $a_5, \cdots, a_8$. Finally, note that the discontinuities of all seven Fréchét derivatives lie at the boundaries of the eight finest-scale intervals partitioned by the four nodes at scale $m = 2$. Over each of these intervals, the Fréchét derivatives are constant, and thus linearly dependent. This "local linear dependence" means that $\langle g_s^i, f_s\rangle \propto \langle g_s^j, f_s\rangle$ for all nodes s at scales $m(s) > 2$. Therefore, the augmentation of any state $z(s)$ for $m(s) \geq 2$ is given by the two local averages over the two finest-scale intervals descendent from node s. This augmentation is illustrated in Fig. 8. The seven measurements are thus incorporated with only a minor increase in the state dimension, especially at the finer scale nodes. These increases are considerably less than would be predicted from a repeated application of the algorithm of Section III-C, and are due to the local linear dependence of the kernels g^i over the finest-scale intervals partitioned by the nodes of the tree. Thus one can imagine modifying the structure of the tree models, i.e., tailoring the descendents of each node, to maximize this linear dependence and minimize the effect of the augmentation on the estimation algorithm.

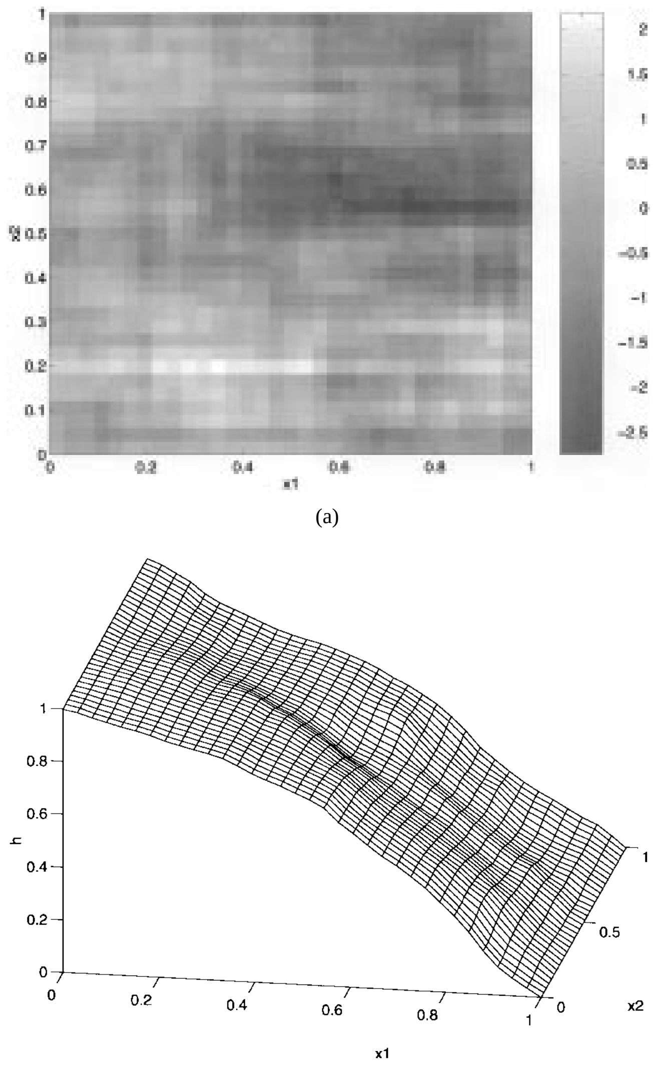
Another way to reduce the effect of the state augmentation on the multiscale estimator is to distribute the measurements at various nodes on the tree. One problem with placing all the measurements at a single node is that the dimension of this node can become quite large, and the computations required by the multiscale estimator increase cubicly with each state dimension. For this example, keep $\langle g^4, f\rangle$ at the root node, but place $\langle g^2, f\rangle$ and $\langle g^6, f\rangle$ at nodes $0\alpha_1$ and $0\alpha_2$ and place $\langle g^i, f\rangle$ for $i = 1, 3, 5, 7$ at the four nodes at scale $m = 2$. In this case, by repeatedly applying the algorithm of Section III-C and also accounting for local linear dependence, the dimension of the state at the root node increases by only two, the states at scales $m = 1$ and $m = 2$ increase by three, and the remaining states for scales $m > 2$ increase by two. Thus a redistribution of the coarse-resolution functionals leads in this case to nontrivial computational savings.

B. Two-Dimensional Flow

For steady-state flow in 2-D, consider estimating $f(\underline{x})$ on the square domain $\Omega = [0, 1] \times [0, 1]$. Assume that $f(\underline{x})$ is a Markov Random Field with zero mean and covariance

$$E[f(\underline{x}) f(\underline{x} + \underline{r})] = \sigma^2 e^{-|\underline{r}|^T d} \tag{30}$$

where $|\underline{r}|^T = [|r_1|, \ |r_2|]$. Assume that $\sigma^2 = 1$ and that $d^T = [5/3, 6]$. The rectangularly spaced samples of this



(a)

(b)

Fig. 9. (a) A sample path of the log conductivity function and (b) the corresponding head function.

process form a discrete-index Markov random field,[8] which can be mapped to the finest scale of the multiscale models for MRF's discussed in Section II-C. In particular, assume a quad-tree $(q_s = 4)$ with five scales and 33-by-33 elements at the finest scale. A sample path of this process is given in Fig. 9(a). Note that the conductivity function is anisotropic, with stronger correlation in the horizontal than the vertical direction. Such horizontal stratification is typical of groundwater aquifers which arise from sedimentary deposition. The corresponding head function is given in Fig. 9(b), assuming the following boundary conditions: $h = 1$ along $x_1 = 0$, $h = 0$ along $x_1 = 1$, and the water flux normal to the boundaries $x_2 = 1$ and $x_2 = 0$ is equal to zero. Note that the head function is considerably smoother than the conductivity function.

The locations of the measurements from which the conductivity function is to be estimated are illustrated in

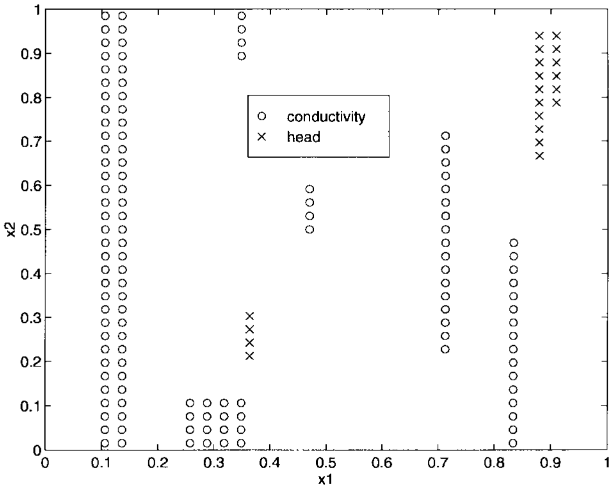


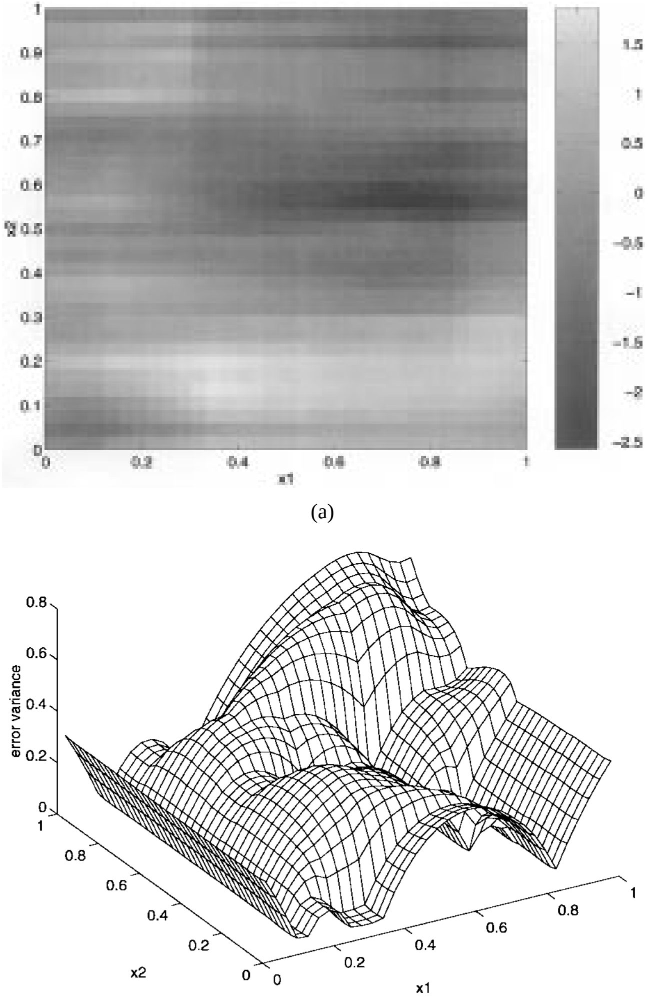
Fig. 10. The locations of the conductivity measurements (○'s) and head measurements (×'s).

Fig. 10. This measurement geometry mimics that given by full and partially penetrating wells. The noise added to the conductivity measurements has standard deviation $\sigma_f = 0.32$, while the noise added to the head measurements has standard deviation $\sigma_h = 0.071$. For incorporation into the multiscale framework, the head measurements are linearized about the conductivity mean, $m_f = 0$. The variables of the multiscale model are augmented with the linearized head measurements according to Section III-C. However, similar to the Fréchét kernels in 1-D, the 2-D kernels $g(\underline{x}_i, \underline{x} \mid m_f)$ are probabilistically dependent over some of the finest-scale domains partitioned by the tree nodes. As a consequence, by placing the head measurements somewhat arbitrarily about the states at scales $m = 1$ and $m = 2$, the maximum increase in the dimension of any state is 37. (The state dimensions can be reduced further if slight errors in the model can be tolerated.) The estimate of the finest-scale process, $\hat{f}(x)$, and the corresponding estimation error variance are plotted in Fig. 11.

C. Discussion of Examples and Method

For both 1-D and 2-D flow, the head samples contribute relatively little information about the hydraulic conductivity function. For 1-D flow, the contribution of the head measurements is evidenced by the "mild" inflections in the conductivity estimate of Fig. 7 near the locations of the head measurements; yet the reduction in uncertainty due to a head measurement is less than that due to a conductivity measurement. For 2-D flow, the insensitivity of head samples to hydraulic conductivity variations is evidenced in regions of dense head measurements by both the smoothness of the 2-D head function illustrated in Fig. 9(b) and the error variance shown in Fig. 11(b). Note that the error variance in regions of dense conductivity measurements decreases much more dramatically than in areas of dense head measurements. This insensitivity of the head measurements to local variations in hydraulic

[8] In general, sampling a continuous-index MRF does not produce a discrete-index MRF [2].



(a)

(b)

Fig. 11. (a) The LLSE estimate of the log conductivity function in Fig. 9(a), and (b) the variance of the estimation errors.

conductivity is due in part to the poor linearization given by the mean conductivity function as the point of linearization, and is improved by successively relinearizing about the current conductivity estimate. However, the real problem is that the estimation of hydraulic conductivity from head is an extremely ill-posed problem [21]. For the examples in this paper, the head measurements were incorporated only to verify that the multiscale framework can be used to fuse measurements of varying spatial resolution. (In our present work we are using the multiscale framework to also incorporate tracer travel time data.)

Another feature of the hydraulic conductivity estimates $\hat{f}(x)$ is that they have fine-scale fluctuations only where such variations can be inferred from the data, e.g., near the line $x_1 = 0.12$ in Fig. 11(a). In terms of the error variance, conductivity is estimated with fine-scale features only in areas where the uncertainty is small. This suggests reducing the number of parameters used to describe the conductivity estimate in areas where the estimate is smooth. Because the multiscale estimator produces both estimates and error variances at each scale of the process, an estimate with space-varying resolution can be selected which possesses some optimal trade-off between resolution, number of parameters, and variance. Within a full nonlinear optimization, where for each iteration the head measurements are linearized about the current conductivity estimate, this approach might eventually be used to reduce the number of parameters used to re-linearize the head measurements, and hence reduce the number of computations require for the inversion.

As for the efficiency of the multiscale framework, for the 33-by-33 example the multiscale estimator requires slightly fewer computations (380 M-flops) than does a direct solution to the normal equations (470 M-flops) for producing the LLSE estimate and the corresponding error variances. However, this differential will become more pronounced for larger domains or if the number of measurements is increased. (The 33-by-33 example was the largest sized problem for which we could directly compute on a workstation the solution to the normal equations and the corresponding error variance.) Also, while the number of computations required to implement the normal equations increases cubicly with the number of finest-scale measurements, the number of finest-scale measurements has only minimal effect upon the number of computations required by the multiscale estimator [23]. Note that the overall sampling density of the finest-scale process for our 2-D example is low, as illustrated in Fig. 10, so the multiscale framework will compare more favorably as the number of finest-scale measurements increases.

VI. Conclusion

In this paper, we showed how the efficient multiscale estimation framework can be extended to data fusion problems. The basic idea is to model each measurement as a point value of the multiscale process, where the location in scale and space of this point value depends upon the resolution and region of support of the measurement. The method we proposed was to augment the variables of an internal multiscale model with linear functions of the finest-scale process derived from the nonlocal functions which are to be measured. Furthermore, we showed how probabilistic dependence among the nonlocal measurements and the existing state variables $z(s) = W_s f_s$ leads to reduced-order augmentations, and therefore to efficient solutions of the corresponding data fusion problems. In particular, the multiresolution framework was used to estimate hydraulic conductivity from point observations of hydraulic conductivity and head.

In terms of the data fusion problems described, the multiscale framework is attractive for a number of reasons. First, the resolution and spatial distributions of the measurements can be completely arbitrary, which poses problems for many of the standard tools used to solve large estimation problems. For instance, Fourier based methods for implementing the normal equations assume that the measurements are both regularly spaced and at a single resolution. Second, the multiresolution estimator provides error variances without any increase in computations. These error variances are useful for evaluating the reduction in uncertainty supplied by the measurements, or for selecting the location and resolution of additional measurements. Also, the estimation errors can be modeled by a multiscale tree whose parameters are a by-product of the multiscale estimator [18]. These error models can be used to efficiently compute multiple conditional simulations of the hydraulic conductivity function, which are useful for characterizing the behavior of groundwater aquifers. In contrast, conditional simulations computed by standard implementations of the LLSE estimator require the Cholesky factorization of a large and nonstationary estimation error covariance. Third, note that the multiscale estimator can be used within a complete nonlinear inversion. Each iteration of a Gauss–Newton optimization is equivalent to implementing a LLSE estimator [22], where the nonlinear measurements are linearized at each iteration about the present estimate of the conductivity function. Therefore, the multiscale modeling and estimation can be used to efficiently implement each iteration of the optimization. (In the examples, we considered only the initial iteration, with the mean as the point of linearization.) Furthermore, multiscale estimates have been shown to be useful for both accelerating and improving the accuracy of these nonlinear optimizations [16].

The challenge for the multiscale framework in the context of data fusion is to apply it to truly large problems, e.g., to estimate f over a large spatial domain Ω and to incorporate other measurements, like contaminant concentrations. This will require developing a comprehensive framework for managing the dimensions of the state augmentations. This is a complicated problem, since, as shown in the examples, the dimension of the augmentations is affected by the support of the nonlocal measurements, the location at which the measurements are placed on the tree, and the regions descending from each tree node. Another method for managing the state dimensions is to consider building approximate models which trade off statistical fidelity for computational efficiency. Because the prior model is usually only a description of the phenomena of interest, approximating the statistics is reasonable and necessary when the problems are large enough.

Acknowledgment

The authors would like to thank D. McLaughlin of the MIT Parsons Laboratory and D. Rossi of Schlumberger–Doll Research for guidance on the groundwater application, and P. Fieguth of the University of Toronto for the use of his multiscale estimator code.

References

[1] K. E. Brewer and S. W. Wheatcraft, "Chapter inculding multiscale information in the characterization of hydraulic conductivity distributions," in *Wavelets in Geophysics*. New York: Academic, 1994, pp. 213–248.

[2] R. Chellappa and A. Jain, Eds., *Markov Random Fields*. New York: Academic, 1993.

[3] K. C. Chou and A. S. Willsky, "A multi-resolution, probabilistic approach to two-dimensional inverse conductivity problems," *Signal Process.*, vol. 18, 1989.

[4] K. C. Chou, A. S. Willsky, and A. Benveniste, "Multiscale recursive estimation, data fusion, and regularization," *IEEE Trans. Automat. Contr.*, vol. 39, no. 3, pp. 464–478, 1994.

[5] M. M. Daniel, A. S. Willsky, D. B. McLaughlin, and D. J. Rossi, "A multi-resolution approach for imaging hydraulic conductivity," in *IEEE Int. Conf. on Image Process.*, Washington, DC, Oct. 1995.

[6] M. H. A. Davis, *Linear Estimation and Stochastic Control*. London: Chapman and Hall, 1977.

[7] H. Derin and P. A. Kelly, "Discrete-index Markov-type random processes," *Proc. IEEE*, vol. 77, Oct. 1989.

[8] P. W. Fieguth, W. C. Karl, A. S. Willsky, and C. Wunsch, "Multiresolution optimal interpolation and statistical analysis of TOPEX/POSEIDEN satellite altimetry," *IEEE Trans. Geosci. Remote Sensing*, vol. 33, pp. 280–292, Mar. 1995.

[9] P. W. Fieguth and A. S. Willsky, "Fractal estimation using models on multiscale trees," *IEEE Trans. Signal Process.*, pp. 1297–1300, May 1996.

[10] L. G. Gelhar, *Stochastic Subsurface Hydrology*. Englewood Cliffs, NJ: Prentice-Hall, 1993.

[11] M. Ghil and P. Malanotte–Rizzoli, "Chapter data assimilation in meteorology and oceanograpy," *Advances in Geophysics*. New York: Academic, 1991, vol. 33, pp. 141–266.

[12] M. D. Greenberg, *Application of Green's Functions in Science and Engineering*. Englewood Cliffs, NJ: Prentice-Hall, 1971.

[13] G. J. Huffman *et al.*, "Global precipitation estimates based on a technique for combining satellite-based estimates, rain gauge analysis, and NWP model precipitation information," *J. Climate*, vol. 8, pp. 1284–1295, 1995.

[14] D. W. Hyndman, J. M. Harris, and S. M. Gorelick, "Coupled seismic and tracer test inversion for aquifer property characterization," *Water Resources Res.*, vol. 30, no. 7, pp. 1965–1977, July 1994.

[15] W. W. Irving, "Multiscale stochastic realization and model identification with applications to large-scale estimation problems," Ph.D. dissertation, MIT, Aug. 1995.

[16] J. Liu, "A multiresolution method for distributed parameter estimation," *SIAM J. Scientif. Computing*, vol. 14, no. 2, pp. 389–405, Mar. 1993.

[17] M. R. Luettgen, W. C. Karl, and A. S. Willsky, "Efficient multiscale regularization with applications to the computation of optical flow," *IEEE Trans. Image Process.*, vol. 3, pp. 41–64, Jan. 1994.

[18] M. R. Luettgen, W. C. Karl, and A. S. Willsky, "Multiscale smoothing error models," *IEEE Trans. Automat. Contr.*, vol. 40, pp. 173–175, Jan. 1995.

[19] M. R. Luettgen, W. C. Karl, A. S. Willsky, and R. R. Tenney, "Multiscale representations of Markov random fields," *IEEE Trans. Signal Proc.*, vol. 41, p. 3377, Dec. 1993.

[20] G. de Marsily, *Quantitative Hydrogeology: Groundwater Hydrology for Engineers*. San Diego, CA: Academic, 1986.

[21] D. B. McLaughlin and L. B. Reid, "Estimating continuous aquifer properties from field measurements: The inverse problem for groundwater flow and transport," in *Computational Methods in Water Resources*. New York: Kluwer, 1994, pp. 777–784.

[22] D. B. McLaughlin and L. R. Townley, "A reassessment of the groundwater inverse problem," *Water Resources Res.*, vol. 32, no. 5, pp. 1131–1161, 1996.

[23] D. Menemenlis, P. W. Fieguth, C. Wunsch, and A. S. Willsky, "A fast optimal interpolation algorithm for mapping hydrographic and other oceanographic data," *J. Geophys. Res.*, 1996.

[24] E. L. Miller, "A multiscale approach to sensor fusion and the solution of linear inverse problems," *Applied and Comp. Harmonic Anal.*, vol. 2, pp. 127–147, 1995.

[25] D. S. Oliver, "Incorporation of transient pressure data into reservoir characterization," *In Situ*, vol. 18, no. 3, pp. 243–275, 1994.

[26] I. Primus, "Scale-recursive estimation of precipitation using remote sensing data," M.S. thesis, MIT, June 1996.

[27] H. E. Rauch, F. Tung, and C. T. Striebel, "Maximum likelihood estimates of linear dynamics systems," *AIAA J.*, vol. 3, no. 8, Aug. 1965.

Michael M. Daniel (Student Member, IEEE) received the B.S. degree from the University of California at Berkeley in 1990 and the S.M. degree from the Massachusetts Institute of Technology (MIT), Cambridge, MA, in 1993.

He is a Teaching Assistant at MIT, where he teaches the circuits, signals, and systems and discrete-time signal processing courses. He has worked at Schlumberger–Doll Research in Ridgefield, CT, and the Cambex Corporation in Waltham, MA. He recently co-authored *Computer Explorations in Signals and Systems using MATLAB*. His research interests include signal and image processing, stochastic processes and uncertainty analysis, and geophysical inverse problems.

Mr. Daniel is a member of SIAM, Sigma Xi, Tau Beta Pi, and Eta Kappa Nu.

Alan S. Willsky (Fellow, IEEE) received both the S.B. and Ph.D. degrees from the Massachusetts Institute of Technology (MIT), Cambridge, MA, in 1969 and 1973, respectively.

He joined the MIT faculty in 1973 and his present position is Professor of Electrical Engineering. From 1974 to 1981, he served as Assistant Director of the Laboratory for Information and Decision Systems at MIT. He is also a founder and member of the board of directors for Alphatech, Inc., Burlington, MA. His research interests are in problems involving multidimensional and multiresolution signal processing and imaging, discrete-event systems, and the asymptotic analysis of control and estimation systems. He is the author of a monograph, *Digital Signal Processing and Control and Estimation Theory*, and is co-author of *Signals and Systems*. He has held visiting positions at Imperial College, l'Université de Paris-Sud, and INRIA (France). He has been an associate editor of several journals, has served as a member of the Board of Governors and Vice President for Technical Affairs of the IEEE Control Systems Society, was program chairman for the 1981 Bilateral Seminar on Control Systems held in the People's Republic of China, and was a special guest editor in 1992 for the IEEE TRANSACTIONS ON INFORMATION THEORY.

Dr. Willsky was named a Distinguished Member of the IEEE Control Systems Society in 1988. In 1975 he received the Donald P. Eckman Award, in 1979 the Alfred Noble Prize, and in 1980 the Broder S. Thompson Memorial Prize recognizing a paper excerpted from his monograph. He was program chair for the 17th IEEE Conference on Decision and Control,