Environmental and Resource Modeling and Decision Analysis
Homework 1
<Problem Identification>

D08625001
彭柏豪

Dealing with biological data, that is such as managing forest always facing the problem of incomplete data or imperfect data. Measurements and model always accompany with error from everywhere. We can't make decisions under the such uncertainty. Traditional regression model return a value that somehow represent the mean or the expected value. However, it is not easy to derive the error, namely the uncertainty from this kind of model. In the future, I think the return of a state-of-art model shouldn't return not only a single expected value of the observations, but also the error. Then first come to my mind is returning distribution. Instead of several value return, the model return a whole distribution. Distribution is the product of the probability model. In term of probability, there cotain events and sample space. Thus, we should first get many many events to construct the sample space.

After 1940s, with the imporvement of computer equitment. If the problem consist of latent error, we can simulate this kind of problem by using the capability of computer. This simulation method is called Monte Carlo Method, namely statistical analogy method. It is a numerical calculation method, using random number or pseudo random number and lots of computer calculations to solve problems. I believe it will be the ways to modern modeling, or the way to construct sample space.

There is a method call "Particle Filter" which is based on Monte Carlo Method. It can estimate the situation of a dynamic system through a series of random sample from lossy and incomplete observation sequence. In dynamic system, we need two model, one for describing the situation when time changing and the other one for describing the noise at every states. Both of them represent by probability. At the beginning, particle filter can only handle the linear problem with the Gaussian noise. But after several improvement, particle filter can now handle the nonlinear problem with any type of noise. And this method can help decision making under uncertainty.

This project will base on *Propagating probability distributions of stand variables using sequential Monte Carlo methods* (2009 Jeffrey H.) which seen growth of trees as dynamic system and using particle filter method for predicting the basal area of the forest stand. Reproduce the result, learning the technique of Bayes modeling and seeking the breakthrough of growth and yield model are the goal of this project and hopely can be finished in this semester.

Propagating probability distributions of stand variables using sequential Monte Carlo methods

JEFFREY H. GOVE

United States Department of Agriculture Forest Service, Northern Research Station, 271 Mast Road, Durham, NH 03824, USA
E-mail: jgove@fs.fed.us

Summary

A general probabilistic approach to stand yield estimation is developed based on sequential Monte Carlo filters, also known as particle filters. The essential steps in the development of the sampling importance resampling (SIR) particle filter are presented. The SIR filter is then applied to simulated and observed data showing how the 'predictor–corrector' scheme employed leads to a general probabilistic mechanism for updating growth model predictions with new observations. The method is applicable to decision making under uncertainty, where uncertainty is found in both model predictions and inventory observations.

Introduction

Forest managers are frequently faced with the prospect of making decisions using imperfect data. Measurements and model predictions can contain error from numerous sources, including sampling, measurement, classification, model estimation and misspecification, to name but a few. These errors may act in ways that are often difficult to quantify and may affect the decision variables in a variety of ways both linearly and non-linearly. In the formulation of optimal planning models for decision making, the manager must also determine whether the sources of uncertainty may accumulate additively or multiplicatively in the system under consideration. This need to incorporate uncertainty into the decision-making process has been recognized and numerous methods have been developed to accommodate it into optimal planning models (Hof *et al.*, 1988; Pickens and Dress, 1988; Gove and Fairweather, 1992; Kangas and Kangas, 1999). One of the major sources of uncertainty in the decision-making process comes from the error in the estimation of stand variables from inventories and model projections. Other sources of uncertainty often include those concerning future prices, interest rates, and changing ownership patterns. While these other sources of error are important, here we concentrate on the evolution of uncertainty in estimates of stand variables through time. That is, the quantification of the probability distribution of stand variables through the combination of model predictions and inventory adjustments over time. Such information not only plays an important role in optimal planning efforts but also can be used in a variety of other less rigorous decision processes such as the determination of stand stocking levels for silvicultural activities and the scheduling of stand examinations.

Much effort has gone into the problem of the estimation of forest stand variables and other quantities in a way that allows for an assessment

Forestry, Vol. 82, No. 4, 2009. doi:10.1093/forestry/cpp009
Advance Access publication date 23 May 2009

of the error in the estimates to be approximated as well. Traditional design-based sampling methods allow for the estimation of sampling error associated with means and totals in the form of standard errors (SEs). Similarly, simulation approaches might be used, for example to quantify the error in predictions from growth models. Too often, however, the steps of stand projection via models and sampling are thought of as disjoint events. And this is where the efforts of compositing and similar strategies have attempted to address merging both model and sampled estimates (Burk *et al.*, 1982; Fairweather and Turner, 1983; Green and Strawderman, 1988). The underlying theme of these efforts is to derive a melded estimate with associated error approximation resulting from the combination of the two different estimates.

In a more general approach to the quantification of error, the entire probability distribution of a stand variable could be estimated. This is the topic of probabilistic state estimation, which was recognized early on by engineers. In the evolution of the state through time, assumptions concerning Gaussianity and linearity are often made, and the methods that have been developed largely are used to propagate the mean and covariance of some vector of state variables. In so doing, these filtering methods seamlessly composite model predictions with incoming measurements in a two-step process. When the Gaussian assumption is tenable, the propagation of the mean and covariance establish the joint probability distribution of the state. When it is not tenable, there will be some associated bias in the estimates. It would make sense in such cases to actually estimate the form of the distribution itself, not just the mean and variance; other methods exist in such cases as will be detailed below.

Propagating the stand distribution through time has the added benefit that it can be used in Bayesian decision making. Ståhl *et al.* (1994) recognized the utility of such an approach by developing a probability-based technique for incorporating inventory planning into the silvicultural decision-making process. Underlying the planning component was a probabilistic framework for propagating the mean and variance of the state through time, where a Gaussian assumption was made on all distributions. These authors used Bayes' rule to incorporate inventory information with assumed sampling error, into a prediction prior using a discretization process. In a related study, Nyström and Ståhl (2001) showed how different components of error could be incorporated into the state density propagation employing a Monte Carlo simulation approach. They also noted that measurements could be incorporated through Bayes' rule and illustrated the conceptual effect this can have in decision making. However, both of these studies appear to be limited in that the methods used may not generalize well. For example, Ståhl *et al.* (1994) note that using distributions other than the Gaussian would quickly lead to intractable calculations. Similarly, it is unclear how Bayes' rule might be employed to perform a measurement update on a prediction prior composed of thousands of Monte Carlo simulated predictions as in Nyström and Ståhl (2001), unless perhaps some parametric form were assumed and estimated. Moreover, as mentioned by Ståhl *et al.* (1994), the generalization of these methods to more than one state variable may present computational challenges.

In this paper, a general state space framework for estimating forest growth and yield in a probabilistic context is introduced that allows for nonlinear, non-Gaussian assumptions, as well as integration of new inventory information with model predictions. With regard to forest stands, the state of the system refers to any variables that are recognized as being important to the quantification and classification of the stand under consideration for estimation purposes. The system state might include such stand-level variables as basal area, number of stems, volume and biomass over all species, or broken down by species; a more detailed state vector might also include parameters for quantifying the stand diameter distribution, for example. Stand-level projection systems might employ these state variables as both dependent and independent variables, and sampling methods would target these state variables to be estimated from an inventory, either directly or indirectly.

The state space approach allows the time course of measurements and model predictions to be incorporated together in a sequential manner, taking into account both model and sampling errors. In this type of system, the disconnect of inventories and models is not present, since the two get fused as part of the sequential march through time. The

general framework that allows this probabilistic data–model fusion stems largely from the engineering literature on filtering and smoothing. Engineering applications have long taken the probabilistic approach to state estimation. The most well-known of the filtering methods is the Kalman filter (Kalman, 1960), which sequentially propagates the mean and covariance of the system state through a two-step process of prediction and update. The prediction step applies the process models to the prior state estimate, while updating corrects this prediction when the new measurement arrives. This 'predictor–corrector' structure of the Kalman filter is found in many subsequent filters of this class. The Kalman filter does not specifically place a Gaussian assumption on the states; however, the fact that it propagates the mean and covariance of the state would suggest that the closer the underlying state to a Gaussian, the better. A more formidable restriction to the Kalman filter is that of linearity: both measurements and process model predictions have to be a linear function of the states.

Because many real-world problems are non-linear and non-Gaussian, numerous extensions or new filters have been developed over the years to better handle such problems. Some examples include the extended (Gelb, 1974, p. 182) and unscented Kalman filters (Julier and Uhlmann, 2004). Shortly after the Kalman filter's introduction, Ho and Lee (1964) showed how the general non-linear non-Gaussian filtering problem could be formulated in a Bayesian setting. Unfortunately, there are only a few special cases to this general filtering problem that have known solutions. This is because it entails knowing the exact form of the distributions involved in order to solve the resulting integrals. Recently, a new class of filters has been put forth that relies on a combination of importance sampling and Monte Carlo simulation to circumvent the integration difficulties encountered in such problems. This class of filters, collectively known as particle filters, provide a straightforward solution to the problem of propagating probability distributions of stand variables through time.

In this paper, we present the probabilistic formulation to the filtering problem and its solution using one particular form of particle filter known as the bootstrap, or sampling importance resampling (SIR) filter. Examples are provided showing the details of the filter's application to a problem concerning the estimation of basal area yield of eastern white pine (*Pinus strobus* L.).

Particle filtering

Particle filtering is a general state space method for the sequential estimation problem of assimilating model predictions with incoming measurements. Let the unobserved system state be given by the vector $\mathbf{x}_t$ with dimension $(n_x \times 1)$, for all time $t = 0, \ldots, T$. The sequence of states is assumed to be a first-order Markov process, which implies that the entire past history of information in $\mathbf{x}_t$ is contained in $\mathbf{x}_{t-1}$. More formally, the distribution of the state $\mathbf{x}_t$ conditional on the entire history of states $\mathbf{x}_{0:t-1} = \mathbf{x}_0, \ldots, \mathbf{x}_{t-1}$ is completely determined by $\mathbf{x}_{t-1}$. Assume that the process dynamics are given by $\mathbf{f}_t$, which can be non-linear, then the model governing the evolution of the states is

$$\mathbf{x}_t = \mathbf{f}_{t-1}(\mathbf{x}_{t-1}, \mathbf{v}_{t-1}), \tag{1}$$

where $\mathbf{v}_t$ is the process noise. The process noise allows, for example for process model misspecification, estimation error or other disturbances in the process model.

Similarly, let $\mathbf{y}_t$ $(n_y \times 1)$ be the measurements or observations at time t. Note that in general, $\mathbf{y}_t$ does not have to be present at each time period, nor does it have to arrive in the estimation process at periods of equal intervals. This will become clear in the next section. If the measurement model is given by $\mathbf{h}_t$, which may also be non-linear, then the state is related to the measurements via the measurement equation

$$\mathbf{y}_t = \mathbf{h}_t(\mathbf{x}_t, \mathbf{n}_t), \tag{2}$$

where $\mathbf{n}_t$ is the measurement noise. The measurement noise not only might be composed of measurement error but also additionally could include components of sampling, or classification error, to name a few.

Equations (1) and (2) make up the dynamic state space formulation of the problem. Note that the only assumptions are that both of these noise processes are white (a white noise process is one in which the random variables are mutually independent (Gelb, 1974, p. 42)) and that their probability density functions (PDFs) are known

and can be sampled from. Note particularly that the noise random variables are not required to be Gaussian.

The random noise components to the models above allow us to state the general state space model in probabilistic terms as

$$p(\mathbf{x}_t|\mathbf{x}_{t-1}) \text{ transition density} \quad (3)$$

$$p(\mathbf{y}_t|\mathbf{x}_t) \text{ measurement density (likelihood),} \quad (4)$$

with $p(\mathbf{x}_0) = p(\mathbf{x}_0|\mathbf{y}_0)$ as the initial condition. Since both the process dynamics and measurement equations are stochastic, it is not possible to infer the exact state from the measurements. Therefore, the goal is to estimate the posterior distribution $p(\mathbf{x}_t|\mathbf{y}_{1:t})$ at time t from all the measurements $\mathbf{y}_{1:t}$.

The general Bayesian solution

The general Bayesian filtering problem can now be formulated. As with the Kalman filter, it consists of prediction and update steps. First, the prediction step yields an estimate of the dynamic prior, or prediction density at time t, based on all of the measurements through time $t-1$ (Gordon *et al.*, 1993)

$$p(\mathbf{x}_t \mid \mathbf{y}_{1:t-1}) = \int p(\mathbf{x}_t \mid \mathbf{x}_{t-1}) p(\mathbf{x}_{t-1} \mid \mathbf{y}_{1:t-1}) \mathrm{d}\mathbf{x}_{t-1}. \quad (5)$$

In the update step, the measurement is assimilated with the prediction density through the application of Bayes' rule, yielding the posterior at time t. It can be shown that (e.g. Simon, 2006, p. 464)

$$p(\mathbf{x}_t \mid \mathbf{y}_{1:t}) = \frac{p(\mathbf{y}_t \mid \mathbf{x}_t) p(\mathbf{x}_t \mid \mathbf{y}_{1:t-1})}{p(\mathbf{y}_t \mid \mathbf{y}_{1:t-1})}, \quad (6)$$

where the normalizing density in the denominator is often termed the data evidence and depends on the likelihood (4), and the dynamic prior (5); that is $p(\mathbf{y}_t \mid \mathbf{y}_{1:t-1}) = \int p(\mathbf{y}_t \mid \mathbf{x}_t) p(\mathbf{x}_t \mid \mathbf{y}_{1:t-1}) \mathrm{d}\mathbf{x}_t$.

In the linear, Gaussian case ($\mathbf{f}_t$ and $\mathbf{h}_t$ linear; $\mathbf{v}_{t-1}$, $\mathbf{n}_t$ additive Gaussian), it can be shown that the Kalman filter is the solution to this filtering problem given in equations (5) and (6). Unfortunately, a general solution to the above Bayesian filtering problem is not available because the integrals involved are generally high dimensional and complex; this has spawned a number of Bayesian algorithms that are suboptimal approximations (Arulampalam *et al.*, 2002; Ristic *et al.*, 2004, pp. 6–8). A more general approach is to use Monte Carlo integration methods to evaluate the intractable integrals in equations (5) and (6). The Monte Carlo approach has the advantage that it can be applied in non-linear, non-Gaussian state space models.

Sequential importance sampling

In what follows, the general idea of the presentation is to motivate the concept of a particle filter through its development from the Bayesian filtering problem above using sequential importance sampling (SIS). The details are lengthy, and therefore are omitted; the interested reader should consult such references as Doucet *et al.* (2001), Ristic *et al.* (2004, Chapter 3), and Simon (2006, Chapter 15).

In order to establish an approximate solution to the conceptual Bayesian filtering problem in equations (5) and (6), importance sampling is used. Importance sampling is a Monte Carlo integration method that facilitates the approximation of general integrals (e.g. Rubenstein, 1981, p. 122). When the true distribution, $p(\mathbf{x}_t)$, is difficult to sample, a proposal distribution, $q(\mathbf{x}_t)$, is chosen whose support covers that of the true distribution, and from which it is easier to sample. The objective in filtering is to estimate equations (5) and (6), and expectations based on these densities sequentially, by propagating the appropriate PDFs through time.

We begin with the general importance sampling approach, and then show how this approach can be made sequential for filtering. In general, we would like to approximate the following expectation

$$\mathrm{E}\left[\mathbf{g}_t(\mathbf{x}_{0:t})\right] = \int \mathbf{g}_t(\mathbf{x}_{0:t}) p(\mathbf{x}_{0:t} \mid \mathbf{y}_{1:t}) \mathrm{d}\mathbf{x}_{0:t}, \quad (7)$$

where $\mathbf{g}_t(\cdot)$ is a function, possibly non-linear, of the states. Employing importance sampling with the proposal density $q(\mathbf{x}_{0:t}|\mathbf{y}_{1:t})$, which covers the support of $p(\mathbf{x}_{0:t}|\mathbf{y}_{1:t})$, yields

$$\mathrm{E}[\mathbf{g}_t(\mathbf{x}_{0:t})] = \int \mathbf{g}_t(\mathbf{x}_{0:t}) p(\mathbf{x}_{0:t} \mid \mathbf{y}_{1:t}) \frac{q(\mathbf{x}_{0:t} \mid \mathbf{y}_{1:t})}{q(\mathbf{x}_{0:t} \mid \mathbf{y}_{1:t})} \mathrm{d}\mathbf{x}_{0:t},$$

Applying Bayes' rule to $p(\mathbf{x}_{0:t}|\mathbf{y}_{1:t})$ and some manipulation eventually yields the importance weights

$$w_t(\mathbf{x}_{0:t}) = \frac{p(\mathbf{y}_{1:t} \mid \mathbf{x}_{0:t})p(\mathbf{x}_{0:t})}{q(\mathbf{x}_{0:t} \mid \mathbf{y}_{1:t})}, \tag{8}$$

allowing the expectation to be written as

$$\mathrm{E}[\mathbf{g}_t(\mathbf{x}_{0:t})] = \frac{\mathrm{E}_q[\mathbf{g}_t(\mathbf{x}_{0:t})w_t(\mathbf{x}_{0:t})]}{\mathrm{E}_q[w_t(\mathbf{x}_{0:t})]},$$

where in each case $\mathrm{E}_q[\cdot]$ means that we are taking the expectation with respect to the proposal distribution $q(\mathbf{x}_{0:t}|\mathbf{y}_{1:t})$.

The Monte Carlo approach to the evaluation of this expectation is to draw N independent, identically distributed samples from the proposal distribution to obtain a weighted approximation to this integral; viz.

$$\widehat{\mathrm{E}}[\mathbf{g}_t(\mathbf{x}_{0:t})] = \sum_{i=1}^{N} \mathbf{g}_t(\mathbf{x}_{0:t}^i)\,\tilde{w}_t^i, \tag{9}$$

where the normalized weights are given as

$$\tilde{w}_t^i = \frac{w_t^i(\mathbf{x}_{0:t})}{\sum_{i=1}^{N} w_t^i(\mathbf{x}_{0:t})} \tag{10}$$

It should be clear from the form of equation (9) that the weights (10) form a discrete representation of the posterior $p(\mathbf{x}_{0:t}|\mathbf{y}_{1:t})$, since the entire quantity is an approximation to equation (7); this will be formalized momentarily. However, the general importance sampling procedure outlined thus far is not a sequential, or recursive, filtering solution. To make this outline of the general procedure sequential, assume that the proposal can be factored as follows

$$q(\mathbf{x}_{0:t}|\mathbf{y}_{1:t}) = q(\mathbf{x}_{0:t-1} \mid \mathbf{y}_{1:t-1})q(\mathbf{x}_t \mid \mathbf{x}_{0:t-1}, \mathbf{y}_{1:t})$$

where the rightmost density on the right-hand side represents an adjustment to the existing path. Then it is straightforward to show by iteration that

$$q(\mathbf{x}_{0:t} \mid \mathbf{y}_{1:t}) = q(\mathbf{x}_0)\prod_{k=1}^{t} q(\mathbf{x}_k \mid \mathbf{x}_{0:k-1}, \mathbf{y}_{1:k}).$$

In addition, the other PDFs in equation (8) can be factored similarly. From these results, it can be shown that the importance weights can also be written recursively (Arulampalam *et al.*, 2002) as

$$w_t^i \propto w_{t-1}^i \frac{p(\mathbf{y}_t \mid \mathbf{x}_t^i)p(\mathbf{x}_t^i \mid \mathbf{x}_{t-1}^i)}{q(\mathbf{x}_t^i \mid \mathbf{x}_{t-1}^i, \mathbf{y}_t)} \tag{11}$$

The sketch of derivation given above yields the components necessary for a particle-based approximate solution to the sequential filtering problem. In the Monte Carlo particle context, the state space is represented by N particles, or support points $\{\mathbf{x}_t^i,\ i = 1, \ldots, N\}$ at time t. The normalized weights $\{\tilde{w}_t^i, i = 1,\ldots,N\}$ represent the contribution to the discretization of the posterior of each support point at time t. The set of support points and associated weights determines a random measure $\{\mathbf{x}_t^i, \tilde{w}_t^i\}_{i=1}^N$. The posterior, therefore, is represented by (Arulampalam *et al.*, 2002; Ristic *et al.*, 2004, p. 39)

$$p(\mathbf{x}_t|\mathbf{y}_{1:t}) \approx \sum_{i=1}^{N} \tilde{w}_t^i\ \delta(\mathbf{x}_t - \mathbf{x}_t^i), \tag{12}$$

where $\delta(\cdot)$ is Dirac's delta (Maybeck, 1979, p. 84) and the weights, $\tilde{w}_t^i$, have been normalized via

$$\tilde{w}_t^i = \frac{w_t^i}{\sum_{i=1}^{N} w_t^i}, \tag{13}$$

such that $\sum_{i=1}^{N} \tilde{w}_t^i = 1$. Furthermore, it can be shown that as $N \to \infty$, the posterior approximation (12) approaches the true posterior density (Crisan and Doucet, 2002).

The general SIS algorithm consists of simply (a) drawing samples $\mathbf{x}_t^i \sim q(\mathbf{x}_t^i \mid \mathbf{x}_{t-1}^i, \mathbf{y}_t)$, (b) calculating the weights via equation (11), and (c) normalizing the weights via equation (13). Steps (a)–(c) are conducted for each particle at each time step, and the entire algorithm is applied sequentially through time. Therefore, the SIS filter algorithm can be seen to propagate the support points and associated weights (i.e. the random measure $\{\mathbf{x}_t^i, \tilde{w}_t^i\}_{i=1}^N$), and thus the discrete estimate of the posterior (12), through time.

The SIR filter

The SIS algorithm has been known for some time. Unfortunately, it has also been known that in application, the variance of the importance weights will increase over time in the SIS algorithm, leading to degeneracy. The result of degeneracy is that

one importance weight will be nearly equal to one, while all of the others will be approximately zero, yielding a poor representation of the posterior. However, a resampling step can be judiciously inserted into the SIS algorithm when certain criteria are met in order to counteract degeneracy (Ristic *et al.*, 2004, pp. 40–41). The main idea of resampling is to multiply particles with high importance weights and eliminate those with small importance weights.

Gordon *et al.* (1993) developed the SIR filter as an extension to the SIS algorithm by resampling at every time step and using the transition density as the importance proposal density. This algorithm was originally called the 'bootstrap' filter because it employed a multinomial resampling scheme, which is equivalent to drawing a weighted bootstrap resample, with weights given by the normalized weights (13). There are many different ways to resample the particles (Hol *et al.*, 2006); however, in this paper, we use the original multinomial scheme. Conceptually, in the resampling step, an integer number of particles, N_t^i, are drawn based on the magnitude of the weight at each support point: if $N_t^i = 0$, the particle is pruned, while connotes replication. More precisely, the weighted measure is replaced by the uniform measure such that ; therefore, the weights are all equal post-resampling.

Table 1 presents the SIR filter algorithm in pseudo-code. Notice in particular, that when the transition density is used as the proposal density, the importance weights (11) become simply . Note also that the weights are all uniform after resampling, so does not appear in this equation. Since the form of the likelihood is known by assumption, calculation of the weights in the importance sampling step is straightforward. In the sampling step, a sample may be drawn as follows: (a) draw a sample from the process noise distribution, then (b) apply the process model (1) as (Ristic *et al.*, 2004, p. 48). This SIR algorithm is used as the non-linear filter in the yield analysis that follows.

Finally, approximating integral quantities such as expectations based on the posterior at any given time period becomes simple under the SIR filter. For example, to estimate the general expectation at time t,

(14)

one would use the approximation based on equation (9)

. (15)

Note that such quantities are calculated based on the weighted approximation to the posterior (12) prior to resampling (Ristic *et al.*, 2004, p. 44).

Table 1: SIR particle filter algorithm

```
for i = 1:N                    Initialize the filter
        ~ p(x_0)
for t = 1:T {                  For each time period
    for i = 1: N {
        Sampling step…
        Draw
        Importance sampling step…
        Calculate
    }
    for i = 1: N
        Normalize the weights via (13)
    Resampling selection step…
    Resample from
}
```

Filtering stand yield

In this section we show how the SIR particle filter can be applied to the estimation of stand yield. In this problem, the system state is the yield at time t, and the process dynamics are stand growth. Measurements enter the system from an inventory. The SIR filter propagates the joint posterior distribution in the form of equation (12), through time. It is important to note that the system state can be multivariate, as outlined in the problem statement. For example, it could be a simple compatible growth and yield system in the spirit of Clutter (1963), or something more complicated with more state variables. However, because it is helpful to the understanding of the filter implementation, here we restrict attention to one state variable, basal area.

Frothingham (1914) presents several yield tables for fully stocked eastern white pine stands

in different quality classes. A Chapman–Richards yield model (Pienaar and Turnbull, 1973) was fitted to the basal area yields in the lowest quality class given (Frothingham, 1914, Table 5). The model is

where t_0 is the initial time and A, k and m are parameters. The fit of the Chapman–Richards yield model to the Frothingham yield data (not shown) was nearly perfect. The differential form of this yield equation is given as

where the parameters and are determined from the parameter estimates of the fitted yield curve; viz. , and .

The state space formulation of the dynamics equations given in equations (1) and (2), can be applied to this model as

where the measurement equation is a simple linear random walk and the vector state notation has been retained for consistency, even though the current problem is scalar. To complete the probabilistic formulation of the state space model as given in (3) and (4), the errors are assumed to be normally distributed, yielding

(16)

(17)

where $\mathbf{Q}_{t-1}$ and $\mathbf{R}_t$ are the associated variances (or, more generally covariance matrices when $n_x > 1$ or $n_y > 1$).

Simulated yield example

In order to apply the particle filter with the model given by equations (16) and (17), the variances in both densities as well as the initial condition require quantification. The SE from the non-linear least squares fit of the Chapman–Richards model can be used for the process noise variance in equation (16); that is $\mathbf{Q}_t = 32$, $t = 1, \ldots, T$. The measurement variance could come from the SE of a forest inventory, such as a periodic stand assessment, and need not be from permanent plots. For this example, we simply set $\mathbf{R}_t = 100$, $t = 1, \ldots, T$, yielding an approximate error of 10 $ft^2\ ac^{-1}$ (2.3 $m^2\ ha^{-1}$). In both instances, we have made the simplifying assumption that the variance is constant through time; however, this is not necessary as will be made clear later, but is only done for expedience. The initial condition was drawn from a density according to the yield equation with appropriate age, corrupted by a Gaussian disturbance with the variance equal to the process noise variance.

Figure 1 presents a single 50-year simulation generated using the state space model described above. The circles denote the true system state in terms of basal area, the quantity we are trying to estimate and is normally (outside of simulation) unknown. Also shown are the estimates from the SIR filter using $N = 1000$ particles. The particle approximation of the posterior given in (12) is used with equation (15) to calculate filter-based estimates such as the mean and variance. In Figure 1, the weighted mean estimate is shown as the solid line, while the maximum *a posteriori* (MAP) estimate is represented by the dot–dash line. The MAP estimate is seen to be almost indistinguishable from the mean largely due to the Gaussian assumptions and the mildly non-linear form of the process dynamics. The MAP estimate was calculated by fitting a weighted kernel density estimator with Gaussian kernel (Silverman, 1986, p. 43) to the posterior representation (12) at each time period. The shaded area shows the 95% posterior or credible intervals, as estimated from the approximation (12). As mentioned earlier, Ristic *et al.* (2004, p. 44) note that all such quantities should be calculated prior to resampling, as done here.

The top panel in Figure 1 shows the SIR filter results when yearly measurements are available. The simulated states are seen to wander about the model yield line due to the addition of process noise. The measurements behave similarly, with the addition of measurement (i.e. sampling) variability. The objective in filtering is to estimate the unknown states using noisy measurements. We consider the filter successful if the posterior intervals capture the unknown states. In this example, this happened 90% of the time (five were missed). The mean estimates from the filter are 'corrected'

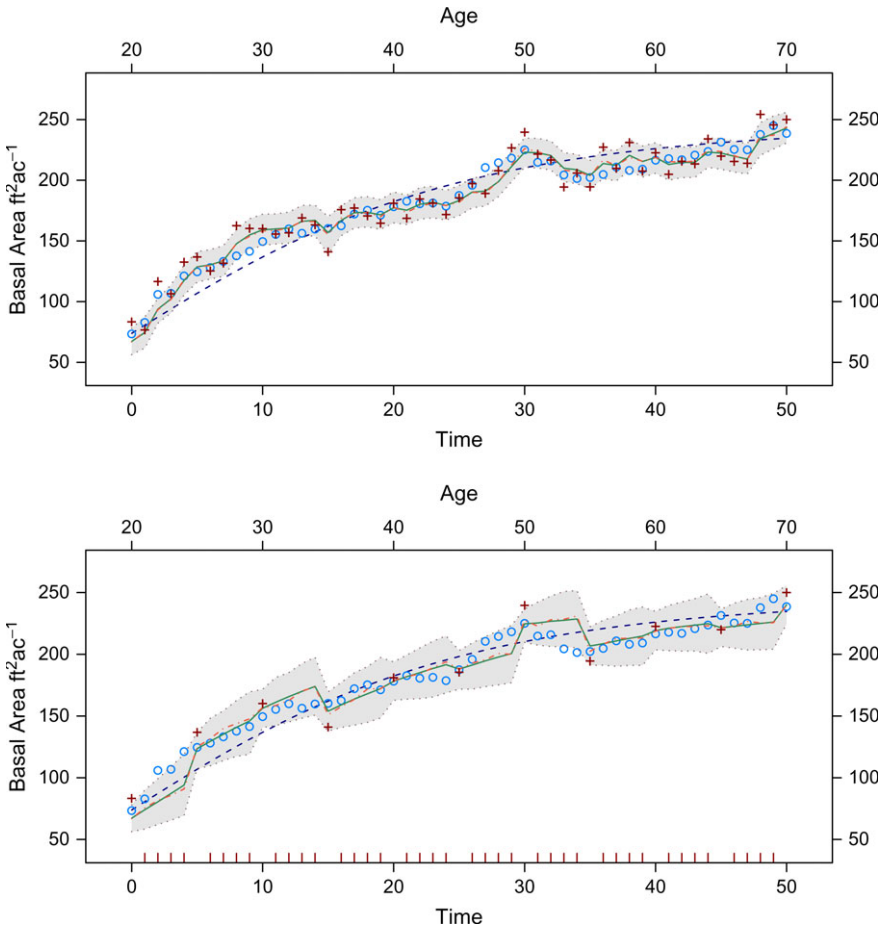


Figure 1. Trajectory of simulated white pine basal area yield. In both panels, the measurements are denoted by the plus, while the true states are circles. The shaded area marks the 95% credible intervals. The solid line is the SIR filter estimate of the mean and the dot–dash line is the MAP estimate based on $N = 1000$ particles. The deterministic yield equation B_t is shown as the long-dash line. In the top panel, measurements are yearly, while in the bottom they are periodic at 5-year intervals, with bars denoting missing measurements. (Note: 1 ft^2 ac^{-1} equals 0.2296 m^2 ha^{-1}.)

by the measurements, but only to a certain extent, which is based on the uncertainty in the likelihood. Consider, for example that what appears to be an aberrant measurement at $t = 15$ pulls the trajectory down, but does so only slightly relative to the measurement. This shows the filter's robustness to poor inventory information in such circumstances.

The correction mechanism just described can be illustrated by plotting all of the densities used in the filter at a given time step. This has been done for a more well-behaved measurement at $t = 30$ in Figure 2. The likelihood is centred about the measurement at 238 ft^2 ac^{-1}, which is higher in basal area than almost the entire transition prior would represent. Indeed, the prior is centred in an area of very low likelihood. Because the likelihood establishes the weights in the SIR filter, the vast majority of the prior gets down-weighted by the low weights, while the upper tail of the prior receives higher weight. The result is the posterior $p(\mathbf{x}_t|\mathbf{y}_{1:t})$, which has shifted towards the observation. In this particular case, the shifting produced a correction in the direction of the true state (225 ft^2 ac^{-1}),

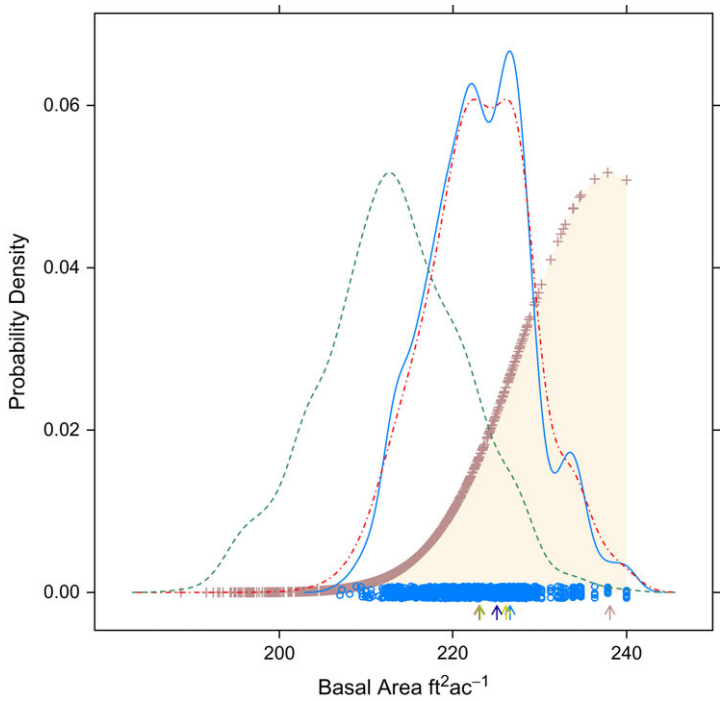


Figure 2. Illustration of the various SIR densities at time t = 30 from the top panel in Figure 1. A kernel density estimator was applied to the transition prior (dashed), posterior prior to resampling (dot–dashed) and the resampled posterior (solid), with the circles showing the resampled particles associated with this density. The likelihood is denoted as the shaded region and is plotted over the support of the prior (plus) and has been scaled to the prior for illustration. The leftmost two arrows (coincident) denote the means for both posterior representations; the next denotes the true state, with the following two denoting the weighted and resampled MAP estimates, respectively, and the rightmost arrow denoting the measurement. (Note: 1 $ft^2\ ac^{-1}$ equals 0.2296 $m^2\ ha^{-1}$.)

which falls between the mean and MAP estimates for the posterior densities. Notice that resampling tends to exaggerate the modes and humps in the weighted representation of the posterior. Both representations of the posterior are approximately normal in overall shape, but have developed slight bimodality and humps due to the weighting of the prior. In cases where the likelihood and the prior are more closely aligned, the multiple modes will disappear and the posteriors will be approximately Gaussian in shape. Additionally, increasing the number of particles may tend to smooth the densities and more closely approach Gaussian in appearance. Notice also that the posterior has smaller variance than the prior: the standard deviation for the posterior is 6.4 $ft^2\ ac^{-1}$while that for the prior is 7.4 $ft^2\ ac^{-1}$ (1.47 and 1.7 $m^2\ ha^{-1}$, respectively). This is typically the case when the inventory provides relatively precise information.

Unfortunately, it would be a rare situation for forest managers to obtain yearly measurements from an annual inventory; more generally, measurements are available periodically. The bottom panel in Figure 1 shows a more realistic estimation run of the SIR filter, where periodic remeasurements enter the record every 5 years. In the absence of measurements, the SIR filter becomes a sequential reapplication of sampling from the prior (Doucet *et al.*, 2000). This is because the rest of the algorithm depends on the weights, which cannot be calculated when measurements are missing (Table 1). There are two salient points to be noticed in this figure. First, as

the time increases since the last received measurement, the credible intervals (now based only on the transition prior) become wider. When a new measurement is recorded – a new inventory taken – the likelihood adjusts the prior density as discussed above and the posterior intervals narrow. Second, notice that in the absence of measurements, the filter mean and MAP estimates tend to track approximately parallel to the underlying yield model. An example of the value of the measurement correction occurs at t = 35. At t = 31, the true state begins a slow decline in basal area, but the filter continues to estimate on the high side, as there has been no external input signalling that any change to the system has occurred. The widening of the intervals still catches the true state until t = 33, when basal area has declined to a point where the intervals fail to cover the true state at the 95% level for the next two periods. The new measurement at t = 35 has appropriately adjusted the prediction trajectory, resulting in the true state's inclusion within the posterior interval again. The credible intervals caught the true state 92% of the time (four missed) over all time periods in this particular example.

The rates at which the posterior intervals captured the true states are slightly lower in the simulation above than the nominal coverage rate of 95%. To determine whether the number of particles used in the above simulation was adequate, a Monte Carlo experiment was performed. Series of length T = 50 were simulated 100 times for each of N = 50, 100, 250 and 500 particles. The results were coverage rates of 90.9%, 93.1%, 94.4% and 94.9%, respectively. Therefore, for this particular set of models and variance parameters, somewhere between N = 250 and 500 particles should result in approximate nominal coverage. Root mean square error analysis based on the same set of simulations supported this conclusion. The fact that the rates in the example above were slightly lower is simply due to a chance selection of the particular trajectory, since the number of particles (N = 1000) was more than adequate.

Updating permanent growth plot estimates

In this section, the SIR filter is applied to a single permanent growth plot with records dating back over 40 years, to illustrate filter estimation on real data. The plot is 0.1 acre (0.4 ha) in size, and is part of an eastern white pine growth study established in the 1950s throughout southern New Hampshire known as the Hatch Study. The Frothingham growth and yield formulation fitted previously is used again to show how the filter will correct aberrant predictions when modelled growth is somewhat optimistic. The same correction mechanism is at work here through the SIR filter as in the simulation example, but the true state of the stand basal area is now unknown.

The top panel in Figure 3 presents the trajectory. The figure shows the somewhat erratic remeasurement schedule adopted with anywhere from 1–6 years lapsing before subsequent remeasurements were taken. Note that the stand measurements plot well below the Frothingham yield curve, implying that the growth predictions will indeed be optimistic. The SIR filter was run on these data with the same assumptions as in the previous simulation; however, since there is only one plot, the measurement variance, $\mathbf{R}_t$, is assumed to be composed solely of measurement error. Regarding the filter mean and MAP estimates, the optimism of the Frothingham yield curve is seen to be especially true early in the trajectory where the slope of the yield curve is more severe. However, after t = 26, the growth of the stand and model predictions are more closely aligned. Notice that when a prediction is corrected by a new measurement, the posterior estimate is drawn towards the measurement, but they do not coincide. Ideally, the correction would be less dramatic with a more suitable growth model (e.g. in the previous simulation example). However, this example illustrates that even an optimistic growth model can be used in this setting, because an estimate of the probability distribution of the state is being propagated, providing a degree of belief in our estimate, rather than simply the stand mean from a deterministic growth projection.

Suppose, alternatively, that the perceived bias associated with the Frothingham growth equation is determined to be too extensive when applied to these data. In this case, the process noise component can be adjusted in an attempt to correct the bias. For example, the bottom panel in Figure 3 shows a run of the SIR filter with everything the same, except that the process error distribution is now assumed to be . The variance

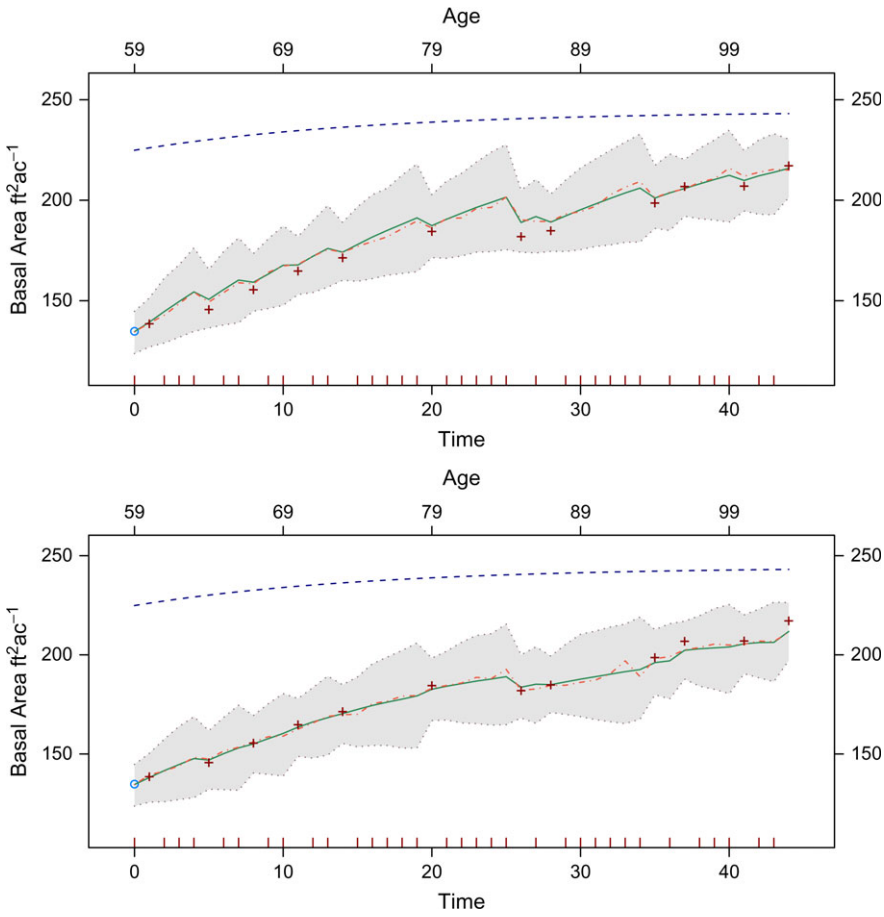


Figure 3. Single Hatch plot trajectory of white pine basal area yield from 40 years of remeasurements. Top panel trajectory has no bias correction; bottom panel trajectory has process model bias correction with . The measurements are denoted by the plus with bars at the figure base denoting missing measurements; the initial state is shown as a circle. The shaded area marks the 95% credible intervals. The solid line is the SIR filter estimate of the mean and the dot–dash line is the MAP estimate using $N = 1000$ particles. The deterministic yield equation B_t is shown as the long-dash line for comparison. (Note: 1 ft^2 ac^{-1} equals 0.2296 m^2 ha^{-1}.)

is still the same; however, a 2-ft^2 ac^{-1} (0.46 m^2 ha^{-1}) downward adjustment on average has been made through the random error component. This adjustment could be approximated by comparing the recorded growth between two periods with that proposed by the Frothingham model and, in general, need not be constant over time; however, here it was simply estimated by eye for illustration. Notice how a simple small change has led to an adjustment in the particle trajectories such that the mean and MAP estimates no longer have a 'sawtooth' form (as a result of the optimistic growth), but rather, more closely conform to the growth trajectory of the measurements. However, even though the filter trajectories align well with the measurement trajectory for a few periods, the intention was not to match the measurements, since they are assumed to be corrupted with noise, but simply to adjust the overly optimistic growth produced by the Frothingham model.

The non-uniform nature of the remeasurement periods in this example should serve to illustrate

that the filter does not require an exact periodic measurement schedule. Additionally, as in the simulation example, notice that the probability intervals widen the further out in time the prediction gets from the past measurement. One very simple method for planning a future remeasurement using such probabilistic estimates would be to schedule the inventory when the intervals get past some tolerable degree of error – this rate may be set somewhat higher with a mismatched growth model like the one used here if left uncorrected.

Figure 4 illustrates how the form of the estimated densities change over time. The density at $t = 28$ is the posterior density and is nearly normal. In the absence of measurements, the prediction densities distort, spread and are translated as they proceed through time (Ristic *et al.*, 2004, p. 4) due to compounding of the uncertainty in the predictions. When the new measurement is assimilated at $t = 35$, the posterior density is again nearly normal, more highly peaked, and has much smaller variance.

It should be clear from these examples that the value of sequential Monte Carlo methods lies not only in the probabilistic propagation of the state but also in the fact that such methods offer a general methodology for updating inventory projections on permanent remeasured growth plots with the latest measurements. In the general filtering literature, this is often referred to equivalently as data assimilation or data–model fusion. A general inventory-updating scheme can be envisioned by running a sequential filter on individual permanent growth plots where the state variables might be, for example the basal area, number of trees, or volume, at each plot. In a large-scale inventory, filters could be run independently, one per plot, since the spatial covariances between plots would probably be negligible. Alternatively, the state vector might be composed of the state variables from many plots, and the covariances estimated, either as part of an adaptive scheme or outside the filter. There is no restriction in either case on when the measurements are recorded on the individual plots; indeed, only a portion of the plots might be remeasured in any particular year.

Discussion and conclusions

Filtering methods owe their genesis to applications such as object tracking, where measurements arrive sequentially in real time, often on the order of fractions of a second. The fact that the time scale is different in forest yield estimation does

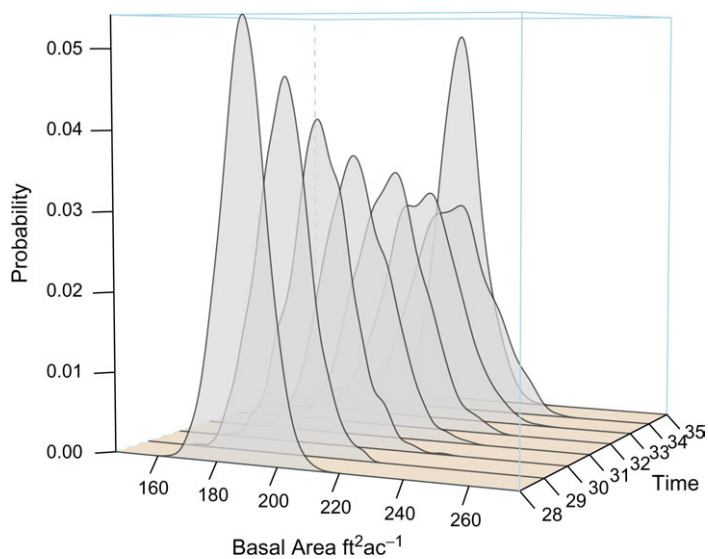


Figure 4. Estimated probability densities for time periods $t = 28$ through $t = 35$ corresponding to the trajectory in the top panel of Figure 3. (Note: 1 $ft^2\ ac^{-1}$ equals 0.2296 $m^2\ ha^{-1}$.)

not matter, it simply gives forest managers the luxury of making adaptive decisions concerning the scheduling of events, such as the next inventory, in a more relaxed manner. Because the time frame between inventory updates will normally be measured in years, it enables the manager and biometrician to consider alternate models, or even to recalibrate existing models based on new sets of measurements in the intervening period. Indeed, the luxury of time might even allow propagating future state estimates based on some new filtering algorithm that has been developed in the interim.

The SIR particle filter described here is only one in a class of general non-linear filters referred to earlier as suboptimal. While the appellation 'suboptimal' may sound less than desirable, especially when applied to a statistical estimation technique, as is explained more fully in Ristic *et al.* (2004, Chapter 2), it is used because there is no single overall solution method to the general non-linear non-Gaussian filtering problem. In this sense, suboptimality is not something to be shunned, it is the best one can do given the circumstances. In the case of linear Gaussian problems, the Kalman filter can be used and is optimal in the minimum mean square error sense. For the general non-linear filtering problem, particle filters are the most flexible because of their ability to handle various error distribution models. This flexibility does, however, come with an associated computational cost – for as the dimension of the state space increases, so too the number of particles must increase to adequately cover the support of the state space in this higher dimensional problem (Gordon *et al.*, 1993). Daum and Huang (2003) have shown that the computational cost of the SIR filter can indeed be high as the state dimension increases. However, Daum and Huang (2003) note that a well-designed filter, with conditional densities close to Gaussian and choice of a good proposal distribution (which is a component of filter design) can mitigate this cost substantially, an observation echoed by Ristic *et al.* (2004, p. 59) and elsewhere. Research into alternative proposal distributions has produced numerous versions of the particle filter and is a subject of continuing intense research that will not be discussed further here, the interested reader may consult Ristic *et al.* (2004, Chapter 3) for a recent overview.

The so-called curse of dimensionality probably will not be a major concern in most forest growth and yield applications for two reasons. First, the state space in stand-level problems is often small, made up of a few state variables characterizing the stand such as basal area, number of trees, volume and the like; and second, most problems encountered will undoubtedly have conditional densities that are nearly Gaussian, as in the examples above (assuming Gaussian noise). Moreover, if need be, algorithms exist for parallelizing particle filters, using specially designed resampling steps that can be run in parallel, to help ameliorate any burdens associated with large N (Hegyi *et al.*, 2007). Needless to say, when real-time sequential estimation is measured in years, rather than in seconds, computational time becomes somewhat of a moot point; even if millions of particles were used, estimation would still take a small fraction of the time available for forest planning.

The overall flexibility of the filtering approach owes much to its original formulation with linked state evolution and measurement equations. This allows the incorporation of two main sources of variability inherent in all coupled inventory and growth projection schemes: sampling error and model uncertainty. In theory, additive measurement errors $\mathbf{n}_t$ are the difference between the measurements and the true unknown value of the state as related through $\mathbf{h}_t$, which may be non-linear. In our examples, the measurement equation is a linear identity function, so that the measurement errors represent the difference between the measurements and the unknown state. The assumption was that these errors were zero mean, Gaussian. This assumption implies that there is no bias in the estimation of the states from the measurements. Inasmuch as the SE from a forest inventory provides information about the variability of the measurement estimate from the true unknown population mean (i.e. the unknown state), its use as the variance component seems reasonable. However, in the case where other errors (e.g. classification) might also be acting upon the measurements, possibly even in a biasing manner, the general error structure for the measurement equation can be used to incorporate these other sources and possible biases as well.

In addition, there is no consistency restriction on measurements, implying that they must always derive from an inventory with similar goals

in precision. For example, returning to the bottom panel of Figure 1 at $t = 30$, recall that mortality due to some event can be inferred, since the true state is decreasing. Perhaps this decline has been noticed by the forest manager. Then it is trivial to incorporate information from an auxiliary estimate during the interim, perhaps from a basal area walk-through count, or some other source like recently flown photography, with associated variance $\mathbf{R}_t$ applied. This variance estimate would likely be higher than that from a normal inventory, both due to the fact that the stand may be less homogeneous from the mortality and because the purpose of the inventory would be for a quick update rather than a full stand examination. The point is that measurement updates need not always come from a full inventory and can be done at any time, yielding a very flexible system.

Similar advantages apply to the process model component of the state space formulation. For example, bias in the process model can be accommodated by specifying a Gaussian or other distribution that is not assumed to have zero mean, as in the Hatch white pine example. In addition, notice in equation (1) that all components are time dependent, including both the model and the noise variance. This structure allows for different model formulations, if desired, at different time periods, with associated changes in variance. This could be as simple as external model recalibration based on the latest information or as extensive as a complete change in model formulation. Indeed, recalibration need not be external to the filter, as the application of filtering methods is not limited solely to state estimation but can include model parameters as well. In general, filters have been applied in various ways to the estimation of model parameters in a sequential manner. Depending upon the complexity of the models and their ability to match the process under consideration, filter-based parameter estimates may be approximately constant over time requiring little adjustment or be adjusted like the states, yielding time-varying parameter trajectories. An example where an unscented Kalman filter was used to estimated the parameters of a process model, in addition to the states, based on eddy covariance measurements (Baldocchi, 2003) is found in Gove and Hollinger (2006). The unscented Kalman filter is closely related to the particle filter described here in that it is a general non-linear filter. They differ primarily in the underlying sampling mechanism. In the particle filter, sampling is based on many particles generated through Monte Carlo methods, while in the unscented filter, a small deterministic sample of the state space is taken at each time period structured in such a way that it captures the mean and covariance of the state. Relatedly, in linear Kalman filter applications to tree-ring studies, the differentiation between state variables and parameters is often blurred, because the models are of simple linear regression form with process mean and slope varying through time (Van Deusen 1991; Gove and Houston 1996).

State space models are known by various names including hidden Markov models (Cappé *et al.*, 2005, p. 4). The latter derives from the fact that the value of the state may be hidden, or unknown, coupled with the Markov assumption discussed earlier. In the examples presented here, the true value of stand basal area per unit area is unknown, but is estimated from measurements on the same variable. This need not be so, the state itself can be unmeasurable (or hidden), for example in the case of stand biomass, and the measurement would then be related to this quantity at time t via the measurement equation $\mathbf{h}_t$.

The yield model used here was of simple structure for pure even-aged stands based on the Chapman–Richards function. More complicated growth and yield models could have been used. Systems of equations common in forest growth prediction on the stand level could easily be accommodated. In models where mortality is explicitly accounted for, it would also become a state variable. In inventory update situations based on temporary rather than permanent plots, which yield components of change, mortality would more naturally fit the description of a hidden state variable, to be inferred from the process model, but not necessarily measured directly in the inventory. In addition, rather than applying an *ad hoc* bias correction as was done here for illustration, a more appropriate growth model would serve to align the dynamic prior with the true posterior. Other filter-based methods that would also help accomplish this involve extensions to the basic SIR filter such as choosing different proposal distributions (Ristic *et al.*, 2004, p. 55).

As detailed in Ståhl *et al.* (1994) and Nyström and Ståhl (2001), the impact of categorizing the stand in terms of its probable state values influences how decisions might be made with respect to scheduling of treatments or inventories. Both studies present decision criteria such as expected loss, and other quantities, based on posterior integrals. The integrals involved are naturally of the form (14); therefore, they may be approximated at any given time t from the particle filter state estimates using equation (15). Again, this particle approximation yields a tractable solution to problems where the posterior is intractable, or where the resulting integrals are difficult and may also be intractable.

The SIR filter is one example of a large class of similar filters for solving non-linear non-Gaussian sequential estimation problems. Its utility in problems concerning probabilistic state estimation of forest stand variables has been demonstrated using simple examples of basal area yield. Use of such methods allows the forest manager to combine model predictions and measurement updates of various forms into an integrated system. The Markovian assumption states that it is unnecessary to keep the past sequence of states beyond $t-1$ when estimating $\mathbf{x}_t$. This assumption – which forms the basis for the majority of important filters – seems quite logical in forest dynamics as most growth models employ it implicitly; that is growth projection proceeds sequentially based on the most recent measurement (or prediction) only, and not based on estimates further back in time. In this sense, forest growth is envisioned in general as a short-memory process. Therefore, not only does the filtering approach make probabilistic sense but also it is in keeping with our current understanding and modelling of the biology of the system. Lastly, the particle approximation of the posterior yields an approximation to general expectations based on the posterior that would be difficult or intractable to solve analytically. The result is not only a general state estimation mechanism but also a methodology that provides the foundation for decision making under uncertainty as well.

Conflict of Interest Statement

None declared.

References

Arulampalam, S., Maskell, S., Gordon, N. and Clapp, T. 2002 A tutorial on particle filters for on-line non-linear/non-Gaussian Bayesian tracking. *IEEE Trans. Signal Process.* **50**, 174–188.

Baldocchi, D.D. 2003 Assessing the eddy covariance technique for evaluating carbon dioxide exchange rates of ecosystems: past, present and future. *Global Chang. Biol.* **9**, 479–492.

Burk, T.E., Hansen, M.H. and Ek, A.R. 1982 Combining sources of information for improved in-place inventory statistics. In *In-place Inventories: Principles and Practices*. T.B. Brann, L.O. House IV and H.G. Lund (eds). Society of American Foresters, Bethesda, MD, 82-02, pp. 413–420.

Cappé, O., Moulines, E. and Rydén, T. 2005 *Inference in Hidden Markov Models*. Springer, New York.

Clutter, J.L. 1963 Compatible growth and yield models for loblolly pine. *For. Sci.* **9**, 354–371.

Crisan, D. and Doucet, A. 2002 A survey of convergence results on particle filtering methods for practitioners. *IEEE Trans. Signal Process.* **50**, 736–746.

Daum, F. and Huang, J. 2003 Curse of dimensionality and particle filters. In *Aerospace Conference 2003 Proceedings*. **volume 4**, IEEE, Big Sky, MT, pp. 1979–1993. http://ieeexplore.ieee.org/xpl/tocresult.jsp?isnumber=27670&isYear=2003

Doucet, A., de Freitas, N. and Gordon, N. 2001 An introduction to sequential Monte Carlo methods. In *Sequential Monte Carlo Methods in Practice*. A. Doucet, N. de Freitas and N. Gordon (eds). Springer, New York, pp. 3–14.

Doucet, A., Godseill, S. and Andrieu, C. 2000 On sequential Monte Carlo sampling methods for Bayesian filtering. *Stat. Comput.* **10**, 197–208.

Fairweather, S.E. and Turner, B.J. 1983 The use of simulated remeasurements in double sampling for successive forest inventory. In *Proceedings, Renewable Resource Inventories for Monitoring Changes and Trends*. In J.F. Bell and T. Atterbury (eds). Oregon State University, Corvallis, OR, pp. 609–612.

Frothingham, E.H. 1914 *White Pine under Forest Management*. Bulletin 13, U.S. Department of Agriculture, U.S. Government Printing Office, Washington, DC.

Gelb, A.E. (ed). 1974 *Applied Optimal Estimation*. MIT Press, Cambridge, MA.

Gordon, N.J., Salmond, D.J. and Smith, A.F.M. 1993 Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proc. F.* **140**, 107–113.

Gove, J.H. and Fairweather, S.E. 1992 Optimizing the management of uneven-aged forest stands: a stochastic approach. *For. Sci.* **38**, 623–640.

Gove, J.H. and Hollinger, D.Y. 2006 Application of a dual unscented Kalman filter for simultaneous state and parameter estimation in problems of surface-atmosphere exchange. *J. Geophys. Res.* **111**, 1–21.

Gove, J.H. and Houston, D.R. 1996 Monitoring the growth of American beech affected by beech bark disease in Maine using the Kalman filter. *Environ. Ecol. Stat.* **3**, 167–187.

Green, E.J. and Strawderman, W.E. 1988 Combining inventory data with model predictions. In *Forest Growth Modelling and Prediction*. A.R. Ek, S.R. Shifley and T.E. Burk (eds). **volume 2**, USDA Forest Service, North Central Forest Experiment Station, GTR NE-120, pp. 676–682.

Hegyi, A., Mihaylova, L., Boel, R. and Lendek, Z. 2007 Parallelizable particle filtering for freeway traffic state estimation. In *Proceedings of the European Control Conference*, Greece. pp. 2442–2449. http://www-sigproc.eng.cam.ac.uk/smc/papers.html (accessed on 13 March, 2008).

Ho, Y.C. and Lee, R.C.K. 1964 A Bayesian approach to problems in stochastic estimation and control. *IEEE Trans. Automat. Contr.* **AC-9**, 333–339.

Hof, J.G., Robinson, K.S. and Betters, D.R. 1988 Optimization with expected values of random yield coefficients in renewable resource linear programming. *For. Sci.* **34**, 634–646.

Hol, J.D., Schön, T.B. and Gustafsson, F. 2006 On resampling algorithms for particle filters. In *Nonlinear Statistical Signal Processing Workshop*. http://www-sigproc.eng.cam.ac.uk/smc/papers.html (accessed on 18 May, 2007), Cambridge, UK.

Julier, S.J. and Uhlmann, J.K. 2004 Unscented filtering and nonlinear estimation. *Proc. IEEE.* **92**, 410–422.

Kalman, R.E. 1960 A new approach to linear filtering and prediction problems. *Trans. ASME J. Basic Eng. Ser.* **82D**, 35–45.

Kangas, A.S. and Kangas, J. 1999 Optimization bias in forest management planning solutions due to errors in forest variables. *Silva Fenn.* **33**, 303–315.

Maybeck, P.S. 1979 *Stochastic Models, Estimation, and Control*. **Volume 1**, 1st edn. Academic Press, New York.

Nyström, K. and Ståhl, G. 2001 Forecasting probability distributions of forest yield for a Bayesian approach to management planning. *Silva Fenn.* **35**, 185–201.

Pickens, J.B. and Dress, P.E. 1988 Use of stochastic production coefficients in linear programming models: Objective function distribution, feasibility, and dual activities. *For. Sci.* **34**, 574–591.

Pienaar, L.V. and Turnbull, K.J. 1973 The Chapman-Richards generalization of Von Bertalanffy's growth model for basal area growth and yield in even-aged stands. *For. Sci.* **19**, 2–22.

Ristic, B., Arulampalam, S. and Gordon, N. 2004 *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. Artech House, Boston, MA.

Rubenstein, R.Y. 1981 *Simulation and the Monte Carlo Method*. Wiley, New York.

Silverman, B.W. 1986 *Density Estimation for Statistics and Data Analysis*, 1st edn. Chapman and Hall/CRC, New York.

Simon, D. 2006 *Optimal State Estimation*. Wiley, Hoboken, NJ.

Ståhl, G., Carlsson, D. and Bondesson, L. 1994 A method to determine optimal stand data acquisition policies. *For. Sci.* **40**, 630–649.

Van Deusen, P.C. 1991 Trend monitoring with varying coefficient models. *For. Sci.* **37**, 1365–1375.

Received 19 May 2008

A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking

M. Sanjeev Arulampalam, Simon Maskell, Neil Gordon, and Tim Clapp

***Abstract*—Increasingly, for many application areas, it is becoming important to include elements of nonlinearity and non-Gaussianity in order to model accurately the underlying dynamics of a physical system. Moreover, it is typically crucial to process data on-line as it arrives, both from the point of view of storage costs as well as for rapid adaptation to changing signal characteristics. In this paper, we review both optimal and suboptimal Bayesian algorithms for nonlinear/non-Gaussian tracking problems, with a focus on particle filters. Particle filters are sequential Monte Carlo methods based on point mass (or "particle") representations of probability densities, which can be applied to any state-space model and which generalize the traditional Kalman filtering methods. Several variants of the particle filter such as SIR, ASIR, and RPF are introduced within a generic framework of the sequential importance sampling (SIS) algorithm. These are discussed and compared with the standard EKF through an illustrative example.**

***Index Terms*—Bayesian, nonlinear/non-Gaussian, particle filters, sequential Monte Carlo, tracking.**

I. Introduction

MANY problems in science require estimation of the state of a system that changes over time using a sequence of noisy measurements made on the system. In this paper, we will concentrate on the state-space approach to modeling dynamic systems, and the focus will be on the discrete-time formulation of the problem. Thus, difference equations are used to model the evolution of the system with time, and measurements are assumed to be available at discrete times. For dynamic state estimation, the discrete-time approach is widespread and convenient.

The state-space approach to time-series modeling focuses attention on the state vector of a system. The state vector contains all relevant information required to describe the system under investigation. For example, in tracking problems, this information could be related to the kinematic characteristics of the target. Alternatively, in an econometrics problem, it could be related to monetary flow, interest rates, inflation, etc. The measurement vector represents (noisy) observations that are related to the state vector. The measurement vector is generally (but not necessarily) of lower dimension than the state vector. The state-space approach is convenient for handling multivariate data and nonlinear/non-Gaussian processes, and it provides a significant advantage over traditional time-series techniques for these problems. A full description is provided in [41]. In addition, many varied examples illustrating the application of nonlinear/non-Gaussian state space models are given in [26].

In order to analyze and make inference about a dynamic system, at least two models are required: First, a model describing the evolution of the state with time (the system model) and, second, a model relating the noisy measurements to the state (the measurement model). We will assume that these models are available in a probabilistic form. The probabilistic state-space formulation and the requirement for the updating of information on receipt of new measurements are ideally suited for the Bayesian approach. This provides a rigorous general framework for dynamic state estimation problems.

In the Bayesian approach to dynamic state estimation, one attempts to construct the posterior probability density function (pdf) of the state based on all available information, including the set of received measurements. Since this pdf embodies all available statistical information, it may be said to be the complete solution to the estimation problem. In principle, an optimal (with respect to any criterion) estimate of the state may be obtained from the pdf. A measure of the accuracy of the estimate may also be obtained. For many problems, an estimate is required every time that a measurement is received. In this case, a recursive filter is a convenient solution. A recursive filtering approach means that received data can be processed sequentially rather than as a batch so that it is not necessary to store the complete data set nor to reprocess existing data if a new measurement becomes available.[1] Such a filter consists of essentially two stages: prediction and update. The prediction stage uses the system model to predict the state pdf forward from one measurement time to the next. Since the state is usually subject to unknown disturbances (modeled as random noise), prediction generally translates, deforms, and spreads the state pdf. The update operation uses the latest measurement to modify the prediction pdf. This is achieved using Bayes theorem, which is the mechanism for updating knowledge about the target state in the light of extra information from new data.

Manuscript received February 8, 2001; revised October 15, 2001. S. Arulampalam was supported by the Royal Academy of Engineering with an Anglo–Australian Post-Doctoral Research Fellowship. S. Maskell was supported by the Royal Commission for the Exhibition of 1851 with an Industrial Fellowship. The associate editor coordinating the review of this paper and approving it for publication was Dr. Petar M. Djurić.

M. S. Arulampalam is with the Defence Science and Technology Organisation, Adelaide, Australia (e-mail: sanjeev.arulampalam@dsto.defence.gov.au).

S. Maskell and N. Gordon are with the Pattern and Information Processing Group, QinetiQ, Ltd., Malvern, U.K., and Cambridge University Engineering Department, Cambridge, U.K. (e-mail: s.maskell@signal.qinetiq.com; n.gordon@signal.qinetiq.com).

T. Clapp is with Astrium Ltd., Stevenage, U.K. (e-mail: t.clapp@iee.org).

Publisher Item Identifier S 1053-587X(02)00569-X.

[1]In this paper, we assume no out-of-sequence measurements; in the presence of out-of-sequence measurements, the order of times to which the measurements relate can differ from the order in which the measurements are processed. For a particle filter solution to the problem of relaxing this assumption, see [32].

We begin in Section II with a description of the nonlinear tracking problem and its optimal Bayesian solution. When certain constraints hold, this optimal solution is tractable. The Kalman filter and grid-based filter, which is described in Section III, are two such solutions. Often, the optimal solution is intractable. The methods outlined in Section IV take several different approximation strategies to the optimal solution. These approaches include the extended Kalman filter, approximate grid-based filters, and particle filters. Finally, in Section VI, we use a simple scalar example to illustrate some points about the approaches discussed up to this point and then draw conclusions in Section VII. This paper is a tutorial; therefore, to facilitate easy implementation, the "pseudo-code" for algorithms has been included at relevant points.

II. Nonlinear Bayesian Tracking

To define the problem of tracking, consider the evolution of the state sequence $\{\mathbf{x}_k,\, k \in \mathbb{N}\}$ of a target given by

$$\mathbf{x}_k = \mathbf{f}_k(\mathbf{x}_{k-1},\, \mathbf{v}_{k-1}) \tag{1}$$

where $\mathbf{f}_k$: $\Re^{n_x} \times \Re^{n_v} \rightarrow \Re^{n_x}$ is a possibly nonlinear function of the state $\mathbf{x}_{k-1}$, $\{\mathbf{v}_{k-1},\, k \in \mathbb{N}\}$ is an i.i.d. process noise sequence, n_x, n_v are dimensions of the state and process noise vectors, respectively, and $\mathbb{N}$ is the set of natural numbers. The objective of tracking is to recursively estimate $\mathbf{x}_k$ from measurements

$$\mathbf{z}_k = \mathbf{h}_k(\mathbf{x}_k,\, \mathbf{n}_k) \tag{2}$$

where $\mathbf{h}_k$: $\Re^{n_x} \times \Re^{n_n} \rightarrow \Re^{n_z}$ is a possibly nonlinear function, $\{\mathbf{n}_k,\, k \in \mathbb{N}\}$ is an i.i.d. measurement noise sequence, and n_z, n_n are dimensions of the measurement and measurement noise vectors, respectively. In particular, we seek filtered estimates of $\mathbf{x}_k$ based on the set of all available measurements $\mathbf{z}_{1:k} = \{\mathbf{z}_i,\, i = 1,\, \ldots,\, k\}$ up to time k.

From a Bayesian perspective, the tracking problem is to recursively calculate some degree of belief in the state $\mathbf{x}_k$ at time k, taking different values, given the data $\mathbf{z}_{1:k}$ up to time k. Thus, it is required to construct the pdf $p(\mathbf{x}_k|\mathbf{z}_{1:k})$. It is assumed that the initial pdf $p(\mathbf{x}_0|\mathbf{z}_0) \equiv p(\mathbf{x}_0)$ of the state vector, which is also known as the prior, is available ($\mathbf{z}_0$ being the set of no measurements). Then, in principle, the pdf $p(\mathbf{x}_k|\mathbf{z}_{1:k})$ may be obtained, recursively, in two stages: prediction and update.

Suppose that the required pdf $p(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1})$ at time $k-1$ is available. The prediction stage involves using the system model (1) to obtain the prior pdf of the state at time k via the Chapman–Kolmogorov equation

$$p(\mathbf{x}_k|\mathbf{z}_{1:k-1}) = \int p(\mathbf{x}_k|\mathbf{x}_{k-1}) p(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1})\, d\mathbf{x}_{k-1}. \tag{3}$$

Note that in (3), use has been made of the fact that $p(\mathbf{x}_k|\mathbf{x}_{k-1},\, \mathbf{z}_{1:k-1}) = p(\mathbf{x}_k|\mathbf{x}_{k-1})$ as (1) describes a Markov process of order one. The probabilistic model of the state evolution $p(\mathbf{x}_k|\mathbf{x}_{k-1})$ is defined by the system equation (1) and the known statistics of $\mathbf{v}_{k-1}$.

At time step k, a measurement $\mathbf{z}_k$ becomes available, and this may be used to update the prior (update stage) via Bayes' rule

$$p(\mathbf{x}_k|\mathbf{z}_{1:k}) = \frac{p(\mathbf{z}_k|\mathbf{x}_k) p(\mathbf{x}_k|\mathbf{z}_{1:k-1})}{p(\mathbf{z}_k|\mathbf{z}_{1:k-1})} \tag{4}$$

where the normalizing constant

$$p(\mathbf{z}_k|\mathbf{z}_{1:k-1}) = \int p(\mathbf{z}_k|\mathbf{x}_k) p(\mathbf{x}_k|\mathbf{z}_{1:k-1})\, d\mathbf{x}_k \tag{5}$$

depends on the likelihood function $p(\mathbf{z}_k|\mathbf{x}_k)$ defined by the measurement model (2) and the known statistics of $\mathbf{n}_k$. In the update stage (4), the measurement $\mathbf{z}_k$ is used to modify the prior density to obtain the required posterior density of the current state.

The recurrence relations (3) and (4) form the basis for the optimal Bayesian solution.[2] This recursive propagation of the posterior density is only a conceptual solution in that in general, it cannot be determined analytically. Solutions do exist in a restrictive set of cases, including the Kalman filter and grid-based filters described in the next section. We also describe how, when the analytic solution is intractable, extended Kalman filters, approximate grid-based filters, and particle filters approximate the optimal Bayesian solution.

III. Optimal Algorithms

A. Kalman Filter

The Kalman filter assumes that the posterior density at every time step is Gaussian and, hence, parameterized by a mean and covariance.

If $p(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1})$ is Gaussian, it can be proved that $p(\mathbf{x}_k|\mathbf{z}_{1:k})$ is also Gaussian, provided that certain assumptions hold [21]:

- $\mathbf{v}_{k-1}$ and $\mathbf{n}_k$ are drawn from Gaussian distributions of known parameters.
- $\mathbf{f}_k(\mathbf{x}_{k-1},\, \mathbf{v}_{k-1})$ is known and is a linear function of $\mathbf{x}_{k-1}$ and $\mathbf{v}_{k-1}$.
- $\mathbf{h}_k(\mathbf{x}_k,\, \mathbf{n}_k)$ is a known linear function of $\mathbf{x}_k$ and $\mathbf{n}_k$.

That is, (1) and (2) can be rewritten as

$$\mathbf{x}_k = F_k \mathbf{x}_{k-1} + \mathbf{v}_{k-1} \tag{6}$$

$$\mathbf{z}_k = H_k \mathbf{x}_k + \mathbf{n}_k. \tag{7}$$

F_k and H_k are known matrices defining the linear functions. The covariances of $\mathbf{v}_{k-1}$ and $\mathbf{n}_k$ are, respectively, Q_{k-1} and R_k. Here, we consider the case when $\mathbf{v}_{k-1}$ and $\mathbf{n}_k$ have zero mean and are statistically independent. Note that the system and measurement matrices F_k and H_k, as well as noise parameters Q_{k-1} and R_k, are allowed to be time variant.

The Kalman filter algorithm, which was derived using (3) and (4), can then be viewed as the following recursive relationship:

$$p(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1}) = \mathcal{N}(\mathbf{x}_{k-1};\, m_{k-1|k-1},\, P_{k-1|k-1}) \tag{8}$$

$$p(\mathbf{x}_k|\mathbf{z}_{1:k-1}) = \mathcal{N}(\mathbf{x}_k;\, m_{k|k-1},\, P_{k|k-1}) \tag{9}$$

$$p(\mathbf{x}_k|\mathbf{z}_{1:k}) = \mathcal{N}(\mathbf{x}_k;\, m_{k|k},\, P_{k|k}) \tag{10}$$

[2]For clarity, the optimal Bayesian solution solves the problem of recursively calculating the exact posterior density. An optimal algorithm is a method for deducing this solution.

where

$$m_{k|k-1} = F_k m_{k-1|k-1} \tag{11}$$
$$P_{k|k-1} = Q_{k-1} + F_k P_{k-1|k-1} F_k^T \tag{12}$$
$$m_{k|k} = m_{k|k-1} + K_k(\mathbf{z}_k - H_k m_{k|k-1}) \tag{13}$$
$$P_{k|k} = P_{k|k-1} - K_k H_k P_{k|k-1} \tag{14}$$

and where $\mathcal{N}(x;\, m,\, P)$ is a Gaussian density with argument x, mean m, and covariance P, and

$$S_k = H_k P_{k|k-1} H_k^T + R_k \tag{15}$$
$$K_k = P_{k|k-1} H_k^T S_k^{-1} \tag{16}$$

are the covariance of the innovation term $\mathbf{z}_k - H_k m_{k|k-1}$, and the Kalman gain, respectively. In the above equations, the transpose of a matrix M is denoted by M^T.

This is the optimal solution to the tracking problem—if the (highly restrictive) assumptions hold. The implication is that no algorithm can ever do better than a Kalman filter in this linear Gaussian environment. It should be noted that it is possible to derive the same results using a least squares (LS) argument [22]. All the distributions are then described by their means and covariances, and the algorithm remains unaltered, but are not constrained to be Gaussian. Assuming the means and covariances to be unbiased and consistent, the filter then optimally derives the mean and covariance of the posterior. However, this posterior is not necessarily Gaussian, and therefore, if optimality is the ability of an algorithm to calculate the posterior, the filter is then not certain to be optimal.

Similarly, if smoothed estimates of the states are required, that is, estimates of $p(\mathbf{x}_k|\mathbf{z}_{1:k+\ell})$, where $\ell \geq 0$,[3] then the Kalman smoother is the optimal estimator of $p(\mathbf{x}_k|\mathbf{z}_{1:k+\ell})$. This holds if ℓ is fixed (*fixed-lag smoothing*, if a batch of data are considered and $0 \leq \ell \leq k$ (*fixed-interval smoothing*), or if the state at a particular time is of interest k is fixed (*fixed-point smoothing*). The problem of calculating smoothed densities is of interest because the densities at time k are then conditional not only on measurements up to and including time index k but also on future measurements. Since there is more information on which to base the estimation, these smoothed densities are typically tighter than the filtered densities.

Although this is true, there is an algorithmic issue that should be highlighted here. It is possible to formulate a backward-time Kalman filter that recurses through the data sequence from the final data to the first and then combines the estimates from the forward and backward passes to obtain overall smoothed estimates [20]. A different formulation implicitly calculates the backward-time state estimates and covariances, recursively estimating the smoothed quantities [38]. Both techniques are prone to having to calculate matrix inverses that do not necessarily exist. Instead, it is preferable to propagate different quantities using an information filter when carrying out the backward-time recursion [4].

[3]If $\ell = 0$, then the problem reduces to the estimation of $p(\mathbf{x}_k|\mathbf{z}_{1:k})$ considered up to this point.

B. Grid-Based Methods

Grid-based methods provide the optimal recursion of the filtered density $p(\mathbf{x}_k|\mathbf{z}_{1:k})$ if the state space is discrete and consists of a finite number of states. Suppose the state space at time $k-1$ consists of discrete states $\mathbf{x}_{k-1}^i$, $i = 1, \ldots, N_s$. For each state $\mathbf{x}_{k-1}^i$, let the conditional probability of that state, given measurements up to time $k-1$ be denoted by $w_{k-1|k-1}^i$, that is, $\Pr(\mathbf{x}_{k-1} = \mathbf{x}_{k-1}^i|\mathbf{z}_{1:k-1}) = w_{k-1|k-1}^i$. Then, the posterior pdf at $k-1$ can be written as

$$p(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1}) = \sum_{i=1}^{N_s} w_{k-1|k-1}^i \delta(\mathbf{x}_{k-1} - \mathbf{x}_{k-1}^i) \tag{17}$$

where $\delta(\cdot)$ is the Dirac delta measure. Substitution of (17) into (3) and (4) yields the prediction and update equations, respectively

$$p(\mathbf{x}_k|\mathbf{z}_{1:k-1}) = \sum_{i=1}^{N_s} w_{k|k-1}^i \delta(\mathbf{x}_k - \mathbf{x}_k^i) \tag{18}$$

$$p(\mathbf{x}_k|\mathbf{z}_{1:k}) = \sum_{i=1}^{N_s} w_{k|k}^i \delta(\mathbf{x}_k - \mathbf{x}_k^i) \tag{19}$$

where

$$w_{k|k-1}^i \triangleq \sum_{j=1}^{N_s} w_{k-1|k-1}^j p(\mathbf{x}_k^i|\mathbf{x}_{k-1}^j) \tag{20}$$

$$w_{k|k}^i \triangleq \frac{w_{k|k-1}^i p(\mathbf{z}_k|\mathbf{x}_k^i)}{\sum_{j=1}^{N_s} w_{k|k-1}^j p(\mathbf{z}_k|\mathbf{x}_k^j)}. \tag{21}$$

The above assumes that $p(\mathbf{x}_k^i|\mathbf{x}_{k-1}^j)$ and $p(\mathbf{z}_k|\mathbf{x}_k^i)$ are known but does not constrain the particular form of these discrete densities. Again, this is the optimal solution if the assumptions made hold.

IV. Suboptimal Algorithms

In many situations of interest, the assumptions made above do not hold. The Kalman filter and grid-based methods cannot, therefore, be used as described—approximations are necessary. In this section, we consider three approximate nonlinear Bayesian filters:

a) extended Kalman filter (EKF);
b) approximate grid-based methods;
c) particle filters.

A. Extended Kalman Filter

If (1) and (2) cannot be rewritten in the form of (6) and (7) because the functions are nonlinear, then a local linearization of the equations may be a sufficient description of the nonlinearity. The EKF is based on this approximation. $p(\mathbf{x}_k|\mathbf{z}_{1:k})$ is approximated by a Gaussian

$$p(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1}) \approx \mathcal{N}(\mathbf{x}_{k-1};\, m_{k-1|k-1},\, P_{k-1|k-1}) \tag{22}$$
$$p(\mathbf{x}_k|\mathbf{z}_{1:k-1}) \approx \mathcal{N}(\mathbf{x}_k;\, m_{k|k-1},\, P_{k|k-1}) \tag{23}$$
$$p(\mathbf{x}_k|\mathbf{z}_{1:k}) \approx \mathcal{N}(\mathbf{x}_k;\, m_{k|k},\, P_{k|k}) \tag{24}$$

where

$$m_{k|k-1} = \mathbf{f}_k(m_{k-1|k-1}) \tag{25}$$
$$P_{k|k-1} = Q_{k-1} + \hat{F}_k P_{k-1|k-1} \hat{F}_k^T \tag{26}$$
$$m_{k|k} = m_{k|k-1} + K_k(\mathbf{z}_k - \mathbf{h}_k(m_{k|k-1})) \tag{27}$$
$$P_{k|k} = P_{k|k-1} - K_k \hat{H}_k P_{k|k-1} \tag{28}$$

and where now, $\mathbf{f}_k(.)$ and $\mathbf{h}_k(.)$ are nonlinear functions, and $\hat{F}_k$ and $\hat{H}_k$ are local linearizations of these nonlinear functions (i.e., matrices)

$$\hat{F}_k = \left.\frac{d\mathbf{f}_k(x)}{dx}\right|_{x=m_{k-1|k-1}} \tag{29}$$
$$\hat{H}_k = \left.\frac{d\mathbf{h}_k(x)}{dx}\right|_{x=m_{k|k-1}} \tag{30}$$
$$S_k = \hat{H}_k P_{k|k-1} \hat{H}_k^T + R_k \tag{31}$$
$$K_k = P_{k|k-1} \hat{H}_k^T S_k^{-1}. \tag{32}$$

The EKF as described above utilizes the first term in a Taylor expansion of the nonlinear function. A higher order EKF that retains further terms in the Taylor expansion exists, but the additional complexity has prohibited its widespread use.

Recently, the unscented transform has been used in an EKF framework [23], [42], [43]. The resulting filter, which is known as the "unscented Kalman filter," considers a set of points that are deterministically selected from the Gaussian approximation to $p(\mathbf{x}_k|\mathbf{z}_{1:k})$. These points are all propagated through the true nonlinearity, and the parameters of the Gaussian approximation are then re-estimated. For some problems, this filter has been shown to give better performance than a standard EKF since it better approximates the nonlinearity; the parameters of the Gaussian approximation are improved.

However, the EKF always approximates $p(\mathbf{x}_k|\mathbf{z}_{1:k})$ to be Gaussian. If the true density is non-Gaussian (e.g., if it is bimodal or heavily skewed), then a Gaussian can never describe it well. In such cases, approximate grid-based filters and particle filters will yield an improvement in performance in comparison to that of an EKF [1].

B. Approximate Grid-Based Methods

If the state space is continuous but can be decomposed into N_s "cells," $\{\mathbf{x}_k^i: i = 1, \ldots, N_s\}$, then a grid-based method can be used to approximate the posterior density. Specifically, suppose the approximation to the posterior pdf at $k-1$ is given by

$$p(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1}) \approx \sum_{i=1}^{N_s} w_{k-1|k-1}^i \delta(\mathbf{x}_{k-1} - \mathbf{x}_{k-1}^i). \tag{33}$$

Then, the prediction and update equations can be written as

$$p(\mathbf{x}_k|\mathbf{z}_{1:k-1}) \approx \sum_{i=1}^{N_s} w_{k|k-1}^i \delta(\mathbf{x}_k - \mathbf{x}_k^i) \tag{34}$$
$$p(\mathbf{x}_k|\mathbf{z}_{1:k}) \approx \sum_{i=1}^{N_s} w_{k|k}^i \delta(\mathbf{x}_k - \mathbf{x}_k^i) \tag{35}$$

where

$$w_{k|k-1}^i \triangleq \sum_{j=1}^{N_s} w_{k-1|k-1}^j \int_{\mathbf{x}\in\mathbf{x}_k^i} p(\mathbf{x}|\overline{\mathbf{x}}_{k-1}^j)\, d\mathbf{x} \tag{36}$$

$$w_{k|k}^i \triangleq \frac{w_{k|k-1}^i \int_{\mathbf{x}\in\mathbf{x}_k^i} p(\mathbf{z}_k|\mathbf{x})\, d\mathbf{x}}{\sum_{j=1}^{N_s} w_{k|k-1}^j \int_{\mathbf{x}\in\mathbf{x}_k^j} p(\mathbf{z}_k|\mathbf{x})\, d\mathbf{x}}. \tag{37}$$

Here, $\overline{\mathbf{x}}_{k-1}^j$ denotes the center of the jth cell at time index $k-1$. The integrals in (36) and (37) arise due to the fact that the grid points $\mathbf{x}_k^i$, $i = 1, \ldots, N_s$, represent regions of continuous state space, and thus, the probabilities must be integrated over these regions. In practice, to simplify computation, a further approximation is made in the evaluation of $w_{k|k}^i$. Specifically, these weights are computed at the center of the "cell" corresponding to $\mathbf{x}_k^i$

$$w_{k|k-1}^i \triangleq \sum_{j=1}^{N_s} w_{k-1|k-1}^j p(\overline{\mathbf{x}}_k^i|\overline{\mathbf{x}}_{k-1}^j) \tag{38}$$

$$w_{k|k}^i \approx \frac{w_{k|k-1}^i p(\mathbf{z}_k|\overline{\mathbf{x}}_k^i)}{\sum_{j=1}^{N_s} w_{k|k-1}^j p(\mathbf{z}_k|\overline{\mathbf{x}}_k^j)}. \tag{39}$$

The grid must be sufficiently dense to get a good approximation to the continuous state space. As the dimensionality of the state space increases, the computational cost of the approach therefore increases dramatically. If the state space is not finite in extent, then using a grid-based approach necessitates some truncation of the state space. Another disadvantage of grid-based methods is that the state space must be predefined and, therefore, cannot be partitioned unevenly to give greater resolution in high probability density regions, unless prior knowledge is used.

Hidden Markov model (HMM) filters [30], [35], [36], [39] are an application of such approximate grid-based methods in a fixed-interval smoothing context and have been used extensively in speech processing. In HMM-based tracking, a common approach is to use the Viterbi algorithm [18] to calculate the maximum *a posteriori* estimate of the path through the trellis, that is, the sequence of discrete states that maximizes the probability of the state sequence given the data. Another approach, due to Baum–Welch [35], is to calculate the probability of each discrete state at each time epoch given the entire data sequence.[4]

V. Particle Filtering Methods

A. Sequential Importance Sampling (SIS) Algorithm

The sequential importance sampling (SIS) algorithm is a Monte Carlo (MC) method that forms the basis for most sequential MC filters developed over the past decades; see [13],

[4]The Viterbi and Baum–Welch algorithms are frequently applied when the state space is approximated to be discrete. The algorithms are optimal if and only if the underlying state space is truly discrete in nature.

[14]. This sequential MC (SMC) approach is known variously as bootstrap filtering [17], the condensation algorithm [29], particle filtering [6], interacting particle approximations [10], [11], and survival of the fittest [24]. It is a technique for implementing a recursive Bayesian filter by MC simulations. The key idea is to represent the required posterior density function by a set of random samples with associated weights and to compute estimates based on these samples and weights. As the number of samples becomes very large, this MC characterization becomes an equivalent representation to the usual functional description of the posterior pdf, and the SIS filter approaches the optimal Bayesian estimate.

In order to develop the details of the algorithm, let $\{\mathbf{x}_{0:k}^i, w_k^i\}_{i=1}^{N_s}$ denote a *random measure* that characterizes the posterior pdf $p(\mathbf{x}_{0:k}|\mathbf{z}_{1:k})$, where $\{\mathbf{x}_{0:k}^i, i = 0, \ldots, N_s\}$ is a set of support points with associated weights $\{w_k^i, i = 1, \ldots, N_s\}$ and $\mathbf{x}_{0:k} = \{\mathbf{x}_j, j = 0, \ldots, k\}$ is the set of all states up to time k. The weights are normalized such that $\sum_i w_k^i = 1$. Then, the posterior density at k can be approximated as

$$p(\mathbf{x}_{0:k}|\mathbf{z}_{1:k}) \approx \sum_{i=1}^{N_s} w_k^i \delta(\mathbf{x}_{0:k} - \mathbf{x}_{0:k}^i). \tag{40}$$

We therefore have a discrete weighted approximation to the true posterior, $p(\mathbf{x}_{0:k}|\mathbf{z}_{1:k})$. The weights are chosen using the principle of *importance sampling* [3], [12]. This principle relies on the following. Suppose $p(x) \propto \pi(x)$ is a probability density from which it is difficult to draw samples but for which $\pi(x)$ can be evaluated [as well as $p(x)$ up to proportionality]. In addition, let $x^i \sim q(x)$, $i = 1, \ldots, N_s$ be samples that are easily generated from a proposal $q(\cdot)$ called an *importance density*. Then, a weighted approximation to the density $p(\cdot)$ is given by

$$p(x) \approx \sum_{i=1}^{N_s} w^i \delta(x - x^i) \tag{41}$$

where

$$w^i \propto \frac{\pi(x^i)}{q(x^i)} \tag{42}$$

is the normalized weight of the ith particle.

Therefore, if the samples $\mathbf{x}_{0:k}^i$ were drawn from an importance density $q(\mathbf{x}_{0:k}|\mathbf{z}_{1:k})$, then the weights in (40) are defined by (42) to be

$$w_k^i \propto \frac{p(\mathbf{x}_{0:k}^i|\mathbf{z}_{1:k})}{q(\mathbf{x}_{0:k}^i|\mathbf{z}_{1:k})}. \tag{43}$$

Returning to the sequential case, at each iteration, one could have samples constituting an approximation to $p(\mathbf{x}_{0:k-1}|\mathbf{z}_{1:k-1})$ and want to approximate $p(\mathbf{x}_{0:k}|\mathbf{z}_{1:k})$ with a new set of samples. If the importance density is chosen to factorize such that

$$q(\mathbf{x}_{0:k}|\mathbf{z}_{1:k}) = q(\mathbf{x}_k|\mathbf{x}_{0:k-1}, \mathbf{z}_{1:k}) q(\mathbf{x}_{0:k-1}|\mathbf{z}_{1:k-1}) \tag{44}$$

then one can obtain samples $\mathbf{x}_{0:k}^i \sim q(\mathbf{x}_{0:k}|\mathbf{z}_{1:k})$ by augmenting each of the existing samples $\mathbf{x}_{0:k-1}^i \sim q(\mathbf{x}_{0:k-1}|\mathbf{z}_{1:k-1})$ with the new state $\mathbf{x}_k^i \sim q(\mathbf{x}_k|\mathbf{x}_{0:k-1}, \mathbf{z}_{1:k})$. To derive the weight update equation, $p(\mathbf{x}_{0:k}|\mathbf{z}_{1:k})$ is first expressed in terms of $p(\mathbf{x}_{0:k-1}|\mathbf{z}_{1:k-1})$, $p(\mathbf{z}_k|\mathbf{x}_k)$, and $p(\mathbf{x}_k|\mathbf{x}_{k-1})$. Note that (4) can be derived by integrating (45)

$$\begin{aligned} p(\mathbf{x}_{0:k}|\mathbf{z}_{1:k}) &= \frac{p(\mathbf{z}_k|\mathbf{x}_{0:k}|\mathbf{z}_{1:k-1}) p(\mathbf{x}_{0:k}|\mathbf{z}_{1:k-1})}{p(\mathbf{z}_k|\mathbf{z}_{1:k-1})} \\ &= \frac{p(\mathbf{z}_k|\mathbf{x}_{0:k}|\mathbf{z}_{1:k-1}) p(\mathbf{x}_k|\mathbf{x}_{0:k-1}|\mathbf{z}_{1:k-1})}{p(\mathbf{z}_k|\mathbf{z}_{1:k-1})} \\ &\quad \times p(\mathbf{x}_{0:k-1}|\mathbf{z}_{1:k-1}) \qquad (45) \\ &= \frac{p(\mathbf{z}_k|\mathbf{x}_k) p(\mathbf{x}_k(\mathbf{x}_{k-1})}{p(\mathbf{z}_k|\mathbf{z}_{1:k-1})} p(\mathbf{x}_{0:k-1}|\mathbf{z}_{1:k-1}) \\ &\propto p(\mathbf{z}_k|\mathbf{x}_k) p(\mathbf{x}_k(\mathbf{x}_{k-1}) p(\mathbf{x}_{0:k-1}|\mathbf{z}_{1:k-1}). \qquad (46) \end{aligned}$$

By substituting (44) and (46) into (43), the weight update equation can then be shown to be

$$\begin{aligned} w_k^i &\propto \frac{p(\mathbf{z}_k|\mathbf{x}_k^i) p(\mathbf{x}_k^i|\mathbf{x}_{k-1}^i) p(\mathbf{x}_{0:k-1}^i|\mathbf{z}_{1:k-1})}{q(\mathbf{x}_k^i|\mathbf{x}_{0:k-1}^i, \mathbf{z}_{1:k}) q(\mathbf{x}_{0:k-1}^i|\mathbf{z}_{1:k-1})} \\ &= w_{k-1}^i \frac{p(\mathbf{z}_k|\mathbf{x}_k^i) p(\mathbf{x}_k^i|\mathbf{x}_{k-1}^i)}{q(\mathbf{x}_k^i|\mathbf{x}_{0:k-1}^i, \mathbf{z}_{1:k})}. \end{aligned} \tag{47}$$

Furthermore, if $q(\mathbf{x}_k|\mathbf{x}_{0:k-1}, \mathbf{z}_{1:k}) = q(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{z}_k)$, then the importance density becomes only dependent on $\mathbf{x}_{k-1}$ and $\mathbf{z}_k$. This is particularly useful in the common case when only a filtered estimate of $p(\mathbf{x}_k|\mathbf{z}_{1:k})$ is required at each time step. From this point on, we will assume such a case, except when explicitly stated otherwise. In such scenarios, only $\mathbf{x}_k^i$ need be stored; therefore, one can discard the path $\mathbf{x}_{0:k-1}^i$ and history of observations $\mathbf{z}_{1:k-1}$. The modified weight is then

$$w_k^i \propto w_{k-1}^i \frac{p(\mathbf{z}_k|\mathbf{x}_k^i) p(\mathbf{x}_k^i|\mathbf{x}_{k-1}^i)}{q(\mathbf{x}_k^i|\mathbf{x}_{k-1}^i, \mathbf{z}_k)} \tag{48}$$

and the posterior filtered density $p(\mathbf{x}_k|\mathbf{z}_{1:k})$ can be approximated as

$$p(\mathbf{x}_k|\mathbf{z}_{1:k}) \approx \sum_{i=1}^{N_s} w_k^i \delta(\mathbf{x}_k - \mathbf{x}_k^i) \tag{49}$$

where the weights are defined in (48). It can be shown that as $N_s \to \infty$, the approximation (49) approaches the true posterior density $p(\mathbf{x}_k|\mathbf{z}_{1:k})$.

The SIS algorithm thus consists of recursive propagation of the weights and support points as each measurement is received sequentially. A pseudo-code description of this algorithm is given by algorithm 1.

`Algorithm 1: SIS Particle Filter`

$[\{\mathbf{x}_k^i, w_k^i\}_{i=1}^{N_s}] = \mathrm{SIS}[\{\mathbf{x}_{k-1}^i, w_{k-1}^i\}_{i=1}^{N_s}, \mathbf{z}_k]$

- `FOR` $i = 1$: N_s
  - `Draw` $\mathbf{x}_k^i \sim q(\mathbf{x}_k|\mathbf{x}_{k-1}^i, \mathbf{z}_k)$
  - `Assign the particle a weight,` w_k^i, `according to (48)`
- `END FOR`

---

1) Degeneracy Problem: A common problem with the SIS particle filter is the degeneracy phenomenon, where after a few iterations, all but one particle will have negligible weight. It has

been shown [12] that the variance of the importance weights can only increase over time, and thus, it is impossible to avoid the degeneracy phenomenon. This degeneracy implies that a large computational effort is devoted to updating particles whose contribution to the approximation to $p(\mathbf{x}_k|\mathbf{z}_{1:k})$ is almost zero. A suitable measure of degeneracy of the algorithm is the effective sample size N_{eff} introduced in [3] and [28] and defined as

$$N_{eff} = \frac{N_s}{1 + \text{Var}(w_k^{*i})} \tag{50}$$

where $w_k^{*i} = p(\mathbf{x}_k^i|\mathbf{z}_{1:k})/q(\mathbf{x}_k^i|\mathbf{x}_{k-1}^i, \mathbf{z}_k)$ is referred to as the "true weight." This cannot be evaluated exactly, but an estimate $\widehat{N_{eff}}$ of N_{eff} can be obtained by

$$\widehat{N_{eff}} = \frac{1}{\sum_{i=1}^{N_s} (w_k^i)^2} \tag{51}$$

where w_k^i is the normalized weight obtained using (47). Notice that $N_{eff} \leq N_s$, and small N_{eff} indicates severe degeneracy. Clearly, the degeneracy problem is an undesirable effect in particle filters. The brute force approach to reducing its effect is to use a very large N_s. This is often impractical; therefore, we rely on two other methods: a) good choice of importance density and b) use of resampling. These are described next.

2) Good Choice of Importance Density: The first method involves choosing the importance density $q(\mathbf{x}_k|\mathbf{x}_{k-1}^i, \mathbf{z}_k)$ to minimize $\text{Var}(w_k^{*i})$ so that N_{eff} is maximized. The optimal importance density function that minimizes the variance of the true weights w_k^{*i} conditioned on $\mathbf{x}_{k-1}^i$ and $\mathbf{z}_k$ has been shown [12] to be

$$\begin{aligned} q(\mathbf{x}_k(\mathbf{x}_{k-1}^i, \mathbf{z}_k)_{opt} &= p(\mathbf{x}_k(\mathbf{x}_{k-1}^i, \mathbf{z}_k) \\ &= \frac{p(\mathbf{z}_k|\mathbf{x}_k(\mathbf{x}_{k-1}^i) p(\mathbf{x}_k(\mathbf{x}_{k-1}^i)}{p(\mathbf{z}_k(\mathbf{x}_{k-1}^i)}. \end{aligned} \tag{52}$$

Substitution of (52) into (48) yields

$$\begin{aligned} w_k^i &\propto w_{k-1}^i p(\mathbf{z}_k|\mathbf{x}_{k-1}^i) \\ &= w_{k-1}^i \int p(\mathbf{z}_k|\mathbf{x}_k') p(\mathbf{x}_k'|\mathbf{x}_{k-1}^i)\, d\mathbf{x}_k'. \end{aligned} \tag{53}$$

This choice of importance density is optimal since for a given $\mathbf{x}_{k-1}^i$, w_k^i takes the same value, whatever sample is drawn from $q(\mathbf{x}_k|\mathbf{x}_{k-1}^i, \mathbf{z}_k)_{opt}$. Hence, conditional on $\mathbf{x}_{k-1}^i$, $\text{Var}(w_k^{*i}) = 0$. This is the variance of the different w_k^i resulting from different sampled $\mathbf{x}_k^i$.

This optimal importance density suffers from two major drawbacks. It requires the ability to sample from $p(\mathbf{x}_k|\mathbf{x}_{k-1}^i, \mathbf{z}_k)$ and to evaluate the integral over the new state. In the general case, it may not be straightforward to do either of these things. There are two cases when use of the optimal importance density is possible.

The first case is when $\mathbf{x}_k$ is a member of a finite set. In such cases, the integral in (53) becomes a sum, and sampling from $p(\mathbf{x}_k|\mathbf{x}_{k-1}^i, \mathbf{z}_k)$ is possible. An example of an application when $\mathbf{x}_k$ is a member of a finite set is a Jump–Markov linear system for tracking maneuvering targets [15], whereby the discrete modal state (defining the maneuver index) is tracked using a particle filter, and (conditioned on the maneuver index) the continuous base state is tracked using a Kalman filter.

Analytic evaluation is possible for a second class of models for which $p(\mathbf{x}_k|\mathbf{x}_{k-1}^i, \mathbf{z}_k)$ is Gaussian [12], [9]. This can occur if the dynamics are nonlinear and the measurements linear. Such a system is given by

$$\mathbf{x}_k = \mathbf{f}_k(\mathbf{x}_{k-1}) + \mathbf{v}_{k-1} \tag{54}$$

$$\mathbf{z}_k = H_k \mathbf{x}_k + \mathbf{n}_k \tag{55}$$

where

$$\mathbf{v}_{k-1} \sim \mathcal{N}(\mathbf{v}_{k-1}; \mathbf{0}_{n_v \times 1}, Q_{k-1}) \tag{56}$$

$$\mathbf{n}_k \sim \mathcal{N}(\mathbf{n}_k; \mathbf{0}_{n_v \times 1}, R_k) \tag{57}$$

and $\mathbf{f}_k$: $\Re^{n_x} \rightarrow \Re^{n_x}$ is a nonlinear function, $H_k \in \Re^{n_z \times n_x}$ is an observation matrix, and $\mathbf{v}_{k-1}$ and $\mathbf{n}_k$ are mutually independent i.i.d. Gaussian sequences with $Q_{k-1} > \mathbf{0}$ and $R_k > \mathbf{0}$. Defining

$$\mathbf{\Sigma}_k^{-1} = Q_{k-1}^{-1} + H_k^T R_k^{-1} H_k \tag{58}$$

$$\mathbf{m}_k = \mathbf{\Sigma}_k (Q_{k-1}^{-1} \mathbf{f}_k(\mathbf{x}_{k-1}) + H_k^T R_k^{-1} \mathbf{z}_k) \tag{59}$$

one obtains

$$p(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{z}_k) = \mathcal{N}(\mathbf{x}_k; \mathbf{m}_k, \mathbf{\Sigma}_k) \tag{60}$$

and

$$p(\mathbf{z}_k|\mathbf{x}_{k-1}) = \mathcal{N}(\mathbf{z}_k; H_k \mathbf{f}_k(\mathbf{x}_{k-1}), Q_{k-1} + H_k R_k H_k^T). \tag{61}$$

For many other models, such analytic evaluations are not possible. However, it is possible to construct suboptimal approximations to the optimal importance density by using local linearization techniques [12]. Such linearizations use an importance density that is a Gaussian approximation to $p(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{z}_k)$. Another approach is to estimate a Gaussian approximation to $p(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{z}_k)$ using the unscented transform [40]. The authors' opinion is that the additional computational cost of using such an importance density is often more than offset by a reduction in the number of samples required to achieve a certain level of performance.

Finally, it is often convenient to choose the importance density to be the prior

$$q(\mathbf{x}_k|\mathbf{x}_{k-1}^i, \mathbf{z}_k) = p(\mathbf{x}_k|\mathbf{x}_{k-1}^i). \tag{62}$$

Substitution of (62) into (48) then yields

$$w_k^i \propto w_{k-1}^i p(\mathbf{z}_k|\mathbf{x}_k^i). \tag{63}$$

This would seem to be the most common choice of importance density since it is intuitive and simple to implement. However, there are a plethora of other densities that can be used, and as illustrated by Section VI, the choice is the crucial design step in the design of a particle filter.

3) Resampling: The second method by which the effects of degeneracy can be reduced is to use resampling whenever a significant degeneracy is observed (i.e., when N_{eff} falls below some threshold N_T). The basic idea of resampling is to eliminate particles that have small weights and to concentrate on particles with large weights. The resampling step involves generating a new set $\{\mathbf{x}_k^{i*}\}_{i=1}^{N_s}$ by resampling (with replacement) N_s times from an approximate discrete representation of $p(\mathbf{x}_k|\mathbf{z}_{1:k})$ given by

$$p(\mathbf{x}_k|\mathbf{z}_{1:k}) \approx \sum_{i=1}^{N_s} w_k^i \delta(\mathbf{x}_k - \mathbf{x}_k^i) \tag{64}$$

so that $\Pr(\mathbf{x}_k^{i*} = \mathbf{x}_k^j) = w_k^j$. The resulting sample is in fact an i.i.d. sample from the discrete density (64); therefore, the

weights are now reset to $w_k^i = 1/N_s$. It is possible to implement this resampling procedure in $O(N_s)$ operations by sampling N_s ordered uniforms using an algorithm based on order statistics [6], [37]. Note that other efficient (in terms of reduced MC variation) resampling schemes, such as stratified sampling and residual sampling [28], may be applied as alternatives to this algorithm. Systematic resampling [25] is the scheme preferred by the authors [since it is simple to implement, takes $O(N_s)$ time, and minimizes the MC variation], and its operation is described in Algorithm 2, where $\mathbb{U}[a, b]$ is the uniform distribution on the interval $[a, b]$ (inclusive of the limits). For each resampled particle $\mathbf{x}_k^{j*}$, this resampling algorithm also stores the index of its parent, which is denoted by i^j. This may appear unnecessary here (and is), but it proves useful in Section V-B2.

A generic particle filter is then as described by Algorithm 3.

Although the resampling step reduces the effects of the degeneracy problem, it introduces other practical problems. First, it limits the opportunity to parallelize since all the particles must be combined. Second, the particles that have high weights w_k^i are statistically selected many times. This leads to a loss of diversity among the particles as the resultant sample will contain many repeated points. This problem, which is known as *sample impoverishment*, is severe in the case of small process noise. In fact, for the case of very small process noise, all particles will collapse to a single point within a few iterations.[5] Third, since the diversity of the paths of the particles is reduced, any smoothed estimates based on the particles' paths degenerate.[6] Schemes exist to counteract this effect. One approach considers the states for the particles to be predetermined by the forward filter and then obtains the smoothed estimates by recalculating the particles' weights via a recursion from the final to the first time step [16]. Another approach is to use MCMC [5].

Algorithm 2: Resampling Algorithm
$[\{\mathbf{x}_k^{j*}, w_k^j, \ i^j\}_{j=1}^{N_s}] = \text{RESAMPLE}\ [\{\mathbf{x}_k^i, w_k^i\}_{i=1}^{N_s}]$

- Initialize the CDF: $c_1 = 0$
- FOR $i = 2$: N_s
  - Construct CDF: $c_i = c_{i-1} + w_k^i$
- END FOR
- Start at the bottom of the CDF: $i = 1$
- Draw a starting point: $u_1 \sim \mathbb{U}[0, N_s^{-1}]$
- FOR $j = 1$: N_s
  - Move along the CDF: $u_j = u_1 + N_s^{-1}(j-1)$
  - WHILE $u_j > c_i$
    - $i = i + 1$
  - END WHILE
  - Assign sample: $\mathbf{x}_k^{j*} = \mathbf{x}_k^i$
  - Assign weight: $w_k^j = N_s^{-1}$
  - Assign parent: $i^j = i$
- END FOR

---

[5]If the process noise is zero, then using a particle filter is not entirely appropriate. Particle filtering is a method well suited to the estimation of dynamic states. If static states, which can be regarded as parameters, need to be estimated then alternative approaches are necessary [7], [27].

[6]Since the particles actually represent paths through the state space, by storing the trajectory taken by each particle, fixed-lag and fixed-point smoothed estimates of the state can be obtained [4].

Algorithm 3: Generic Particle Filter
$[\{\mathbf{x}_k^i, w_k^i\}_{i=1}^{N_s}] = \text{PF}[\{\mathbf{x}_{k-1}^i, \ w_{k-1}^i\}_{i=1}^{N_s}, \mathbf{z}_k]$

- FOR $i = 1$: N_s
  - Draw $\mathbf{x}_k^i \sim q(\mathbf{x}_k|\mathbf{x}_{k-1}^i, \mathbf{z}_k)$
  - Assign the particle a weight, w_k^i, according to (48)
- END FOR
- Calculate total weight: $t = \text{SUM}[\{w_k^i\}_{i=1}^{N_s}]$
- FOR $i = 1$: N_s
  - Normalize: $w_k^i = t^{-1} w_k^i$
- END FOR
- Calculate $\widehat{N_{eff}}$ using (51)
- IF $\widehat{N_{eff}} < N_T$
  - Resample using algorithm 2:
    - $[\{\mathbf{x}_k^i, w_k^i, -\}_{i=1}^{N_s}] = \text{RESAMPLE}[\{\mathbf{x}_k^i, w_k^i\}_{i=1}^{N_s}]$
- END IF

There have been some systematic techniques proposed recently to solve the problem of sample impoverishment. One such technique is the *resample-move* algorithm [19], which is not be described in detail in this paper. Although this technique draws conceptually on the same technologies of importance sampling-resampling and MCMC sampling, it avoids sample impoverishment. It does this in a rigorous manner that ensures the particles asymtotically approximate samples from the posterior and, therefore, is the method of choice of the authors. An alternative solution to the same problem is *regularization* [31], which is discussed in Section V-B3. This approach is frequently found to improve performance, despite a less rigorous derivation and is included here in preference to the resample-move algorithm since its use is so widespread.

4) Techniques for Circumventing the Use of a Suboptimal Importance Density: It is often the case that a good importance density is not available. For example, if the prior $p(\mathbf{x}_k|\mathbf{x}_{k-1})$ is used as the importance density and is a much broader distribution than the likelihood $p(\mathbf{z}_k|\mathbf{x}_k)$, then only a few particles will have a high weight. Methods exist for encouraging the particles to be in the right place; the use of bridging densities [8] and progressive correction [33] both introduce intermediate distributions between the prior and likelihood. The particles are then reweighted according to these intermediate distributions and resampled. This "herds" the particles into the right part of the state space.

Another approach known as partitioned sampling [29] is useful if the likelihood is very peaked but can be factorized into a number of broader distributions. Typically, this occurs because each of the partitioned distributions are functions of some (not all) of the states. By treating each of these partitioned distributions in turn and resampling on the basis of each such partitioned distribution, the particles are again herded toward the peaked likelihood.

B. Other Related Particle Filters

The sequential importance sampling algorithm presented in Section V-A forms the basis for most particle filters that have

been developed so far. The various versions of particle filters proposed in the literature can be regarded as special cases of this general SIS algorithm. These special cases can be derived from the SIS algorithm by an appropriate choice of importance sampling density and/or modification of the resampling step. Below, we present three particle filters proposed in the literature and show how these may be derived from the SIS algorithm. The particle filters considered are

i) sampling importance resampling (SIR) filter;
ii) auxiliary sampling importance resampling (ASIR) filter;
iii) regularized particle filter (RPF).

1) Sampling Importance Resampling Filter: The SIR filter proposed in [17] is an MC method that can be applied to recursive Bayesian filtering problems. The assumptions required to use the SIR filter are very weak. The state dynamics and measurement functions $\mathbf{f}_k(\cdot, \cdot)$ and $\mathbf{h}_k(\cdot, \cdot)$ in (1) and (2), respectively, need to be known, and it is required to be able to sample realizations from the process noise distribution of $\mathbf{v}_{k-1}$ and from the prior. Finally, the likelihood function $p(\mathbf{z}_k|\mathbf{x}_k)$ needs to be available for pointwise evaluation (at least up to proportionality). The SIR algorithm can be easily derived from the SIS algorithm by an appropriate choice of i) the importance density, where $q(\mathbf{x}_k|\mathbf{x}_{k-1}^i, \mathbf{z}_{1:k})$ is chosen to be the prior density $p(\mathbf{x}_k|\mathbf{x}_{k-1}^i)$, and ii) the resampling step, which is to be applied at every time index.

The above choice of importance density implies that we need samples from $p(\mathbf{x}_k|\mathbf{x}_{k-1}^i)$. A sample $\mathbf{x}_k^i \sim p(\mathbf{x}_k|\mathbf{x}_{k-1}^i)$ can be generated by first generating a process noise sample $\mathbf{v}_{k-1}^i \sim p_v(\mathbf{v}_{k-1})$ and setting $\mathbf{x}_k^i = \mathbf{f}_k(\mathbf{x}_{k-1}^i, \mathbf{v}_{k-1}^i)$, where $p_v(\cdot)$ is the pdf of $\mathbf{v}_{k-1}$. For this particular choice of importance density, it is evident that the weights are given by

$$w_k^i \propto w_{k-1}^i p(\mathbf{z}_k|\mathbf{x}_k^i). \tag{65}$$

However, noting that resampling is applied at every time index, we have $w_{k-1}^i = 1/N \; \forall i$; therefore

$$w_k^i \propto p(\mathbf{z}_k|\mathbf{x}_k^i). \tag{66}$$

The weights given by the proportionality in (66) are normalized before the resampling stage. An iteration of the algorithm is then described by Algorithm 4.

As the importance sampling density for the SIR filter is independent of measurement $\mathbf{z}_k$, the state space is explored without any knowledge of the observations. Therefore, this filter can be inefficient and is sensitive to outliers. Furthermore, as resampling is applied at every iteration, this can result in rapid loss of diversity in particles. However, the SIR method does have the advantage that the importance weights are easily evaluated and that the importance density can be easily sampled.

Algorithm 4: SIR Particle Filter
$[\{\mathbf{x}_k^i, w_k^i\}_{i=1}^{N_s}] = \mathrm{SIR}[\{\mathbf{x}_{k-1}^i, w_{k-1}^i\}_{i=1}^{N_s}, \mathbf{z}_k]$
- FOR $i = 1$: N_s
  - Draw $\mathbf{x}_k^i \sim p(\mathbf{x}_k|\mathbf{x}_{k-1}^i)$
  - Calculate $w_k^i = p(\mathbf{z}_k|\mathbf{x}_k^i)$
- END FOR
- Calculate total weight: $t = \mathrm{SUM}[\{w_k^i\}_{i=1}^{N_s}]$
- FOR $i = 1$: N_s
  - Normalize: $w_k^i = t^{-1} w_k^i$
- END FOR
- Resample using algorithm 2:
  - $[\{\mathbf{x}_k^i, w_k^i, -\}_{i=1}^{N_s}] = \mathrm{RESAMPLE}[\{\mathbf{x}_k^i, \; w_k^i\}_{i=1}^{N_s}]$

---

2) Auxiliary Sampling Importance Resampling Filter: The ASIR filter was introduced by Pitt and Shephard [34] as a variant of the standard SIR filter. This filter can be derived from the SIS framework by introducing an importance density $q(\mathbf{x}_k, i|\mathbf{z}_{1:k})$, which samples the pair $\{\mathbf{x}_k^j, i^j\}_{j=1}^{M_s}$, where i^j refers to the index of the particle at $k-1$.

By applying Bayes' rule, a proportionality can be derived for $p(\mathbf{x}_k, i|\mathbf{z}_{1:k})$ as

$$\begin{aligned} p(\mathbf{x}_k, i|\mathbf{z}_{1:k}) &\propto p(\mathbf{z}_k|\mathbf{x}_k) p(\mathbf{x}_k, i|\mathbf{z}_{1:k-1}) \\ &= p(\mathbf{z}_k|\mathbf{x}_k) p(\mathbf{x}_k|i, \mathbf{z}_{1:k-1}) p(i|\mathbf{z}_{1:k-1}) \\ &= p(\mathbf{z}_k|\mathbf{x}_k) p(\mathbf{x}_k|\mathbf{x}_{k-1}^i) w_{k-1}^i. \end{aligned} \tag{67}$$

The ASIR filter operates by obtaining a sample from the joint density $p(\mathbf{x}_k, i|\mathbf{z}_{1:k})$ and then omitting the indices i in the pair $(\mathbf{x}_k, i)$ to produce a sample $\{\mathbf{x}_k^j\}_{j=1}^{N_s}$ from the marginalized density $p(\mathbf{x}_k|\mathbf{z}_{1:k})$. The importance density used to draw the sample $\{\mathbf{x}_k^j, i^j\}_{j=1}^{M_s}$ is defined to satisfy the proportionality

$$q(\mathbf{x}_k, i|\mathbf{z}_{1:k}) \propto p(\mathbf{z}_k|\mu_k^i) p(\mathbf{x}_k|\mathbf{x}_{k-1}^i) w_{k-1}^i \tag{68}$$

where μ_k^i is some characterization of $\mathbf{x}_k$, given $\mathbf{x}_{k-1}^i$. This could be the mean, in which case, $\mu_k^i = \mathbb{E}[\mathbf{x}_k|\mathbf{x}_{k-1}^i]$ or a sample $\mu_k^i \sim p(\mathbf{x}_k|\mathbf{x}_{k-1}^i)$. By writing

$$q(\mathbf{x}_k, i|\mathbf{z}_{1:k}) = q(i|\mathbf{z}_{1:k}) q(\mathbf{x}_k|i, \mathbf{z}_{1:k}) \tag{69}$$

and defining

$$q(\mathbf{x}_k|i, \mathbf{z}_{1:k}) \triangleq p(\mathbf{x}_k|\mathbf{x}_{k-1}^i) \tag{70}$$

it follows from (68) that

$$q(i|\mathbf{z}_{1:k}) \propto p(\mathbf{z}_k|\mu_k^i) w_{k-1}^i. \tag{71}$$

The sample $\{\mathbf{x}_k^j, i^j\}_{j=1}^{M_s}$ is then assigned a weight proportional to the ratio of the right-hand side of (67) to (68)

$$w_k^j \propto w_{k-1}^{i^j} \frac{p(\mathbf{z}_k|\mathbf{x}_k^j) p\left(\mathbf{x}_k^j|\mathbf{x}_{k-1}^{i^j}\right)}{q(\mathbf{x}_k^j, i^j|\mathbf{z}_{1:k})} = \frac{p(\mathbf{z}_k|\mathbf{x}_k^j)}{p\left(\mathbf{z}_k|\mu_k^{i^j}\right)}. \tag{72}$$

The algorithm then becomes that described by Algorithm 5.

Algorithm 5: Auxiliary Particle Filter
$[\{\mathbf{x}_k^i, w_k^i\}_{i=1}^{N_s}] = \mathrm{APF}[\{\mathbf{x}_{k-1}^i, w_{k-1}^i\}_{i=1}^{N_s}, \mathbf{z}_k]$
- FOR $i = 1$: N_s
  - Calculate μ_k^i
  - Calculate $w_k^i = q(i|\mathbf{z}_{1:k}) \propto p(\mathbf{z}_k|\mu_k^i) w_{k-1}^i$.
- END FOR
- Calculate total weight: $t = \mathrm{SUM}[\{w_k^i\}_{i=1}^{N_s}]$
- FOR $i = 1$: N_s
  - Normalize: $w_k^i = t^{-1} w_k^i$
- END FOR
- Resample using algorithm 2:
  - $[\{-, -, i^j\}_{j=1}^{N_s}] = \mathrm{RESAMPLE}[\{\mathbf{x}_k^i, w_k^i\}_{i=1}^{N_s}]$
- FOR $j = 1$: N_s
  - Draw $\mathbf{x}_k^j \sim q(\mathbf{x}_k|i^j, \mathbf{z}_{1:k}) = p(\mathbf{x}_k|\mathbf{x}_{k-1}^{i^j})$, as in the SIR filter.
  - Assign weight w_k^j using (72)

- END FOR
- Calculate total weight: $t = \text{SUM}[\{w_k^i\}_{i=1}^{N_s}]$
- FOR $i = 1$: N_s
  - Normalize: $w_k^i = t^{-1} w_k^i$
- END FOR

Although unnecessary, the original ASIR filter as proposed in [34] consisted of one more step, namely, a resampling stage, to produce an i.i.d. sample $\{\mathbf{x}_k^j, i^j\}_{j=1}^{M_s}$ with equal weights.

Compared with the SIR filter, the advantage of the ASIR filter is that it naturally generates points from the sample at $k-1$, which, conditioned on the current measurement, are most likely to be close to the true state. ASIR can be viewed as resampling at the previous time step, based on some point estimates μ_k^i that characterize $p(\mathbf{x}_k|\mathbf{x}_{k-1}^i)$. If the process noise is small so that $p(\mathbf{x}_k|\mathbf{x}_{k-1}^i)$ is well characterized by μ_k^i, then ASIR is often not so sensitive to outliers as SIR, and the weights w_k^i are more even. However, if the process noise is large, a single point does not characterize $p(\mathbf{x}_k|\mathbf{x}_{k-1}^i)$ well, and ASIR resamples based on a poor approximation of $p(\mathbf{x}_k|\mathbf{x}_{k-1}^i)$. In such scenarios, the use of ASIR then degrades performance.

3) Regularized Particle Filter: Recall that resampling was suggested in Section V-B1 as a method to reduce the degeneracy problem, which is prevalent in particle filters. However, it was pointed out that resampling in turn introduced other problems and, in particular, the problem of loss of diversity among the particles. This arises due to the fact that in the resampling stage, samples are drawn from a discrete distribution rather than a continuous one. If this problem is not addressed properly, it may lead to "particle collapse," which is a severe case of sample impoverishment where all N_s particles occupy the same point in the state space, giving a poor representation of the posterior density. A modified particle filter known as the regularized particle filter (RPF) was proposed [31] as a potential solution to the above problem.

The RPF is identical to the SIR filter, except for the resampling stage. The RPF resamples from a continuous approximation of the posterior density $p(\mathbf{x}_k|\mathbf{z}_{1:k})$, whereas the SIR resamples from the discrete approximation (64). Specifically, in the RPF, samples are drawn from the approximation

$$p(\mathbf{x}_k|\mathbf{z}_{1:k}) \approx \sum_{i=1}^{N_s} w_k^i K_h(\mathbf{x}_k - \mathbf{x}_k^i) \tag{73}$$

where

$$K_h(\mathbf{x}) = \frac{1}{h^{n_x}} K\left(\frac{\mathbf{x}}{h}\right) \tag{74}$$

is the rescaled Kernel density $K(\cdot)$, $h > 0$ is the Kernel bandwidth (a scalar parameter), n_x is the dimension of the state vector $\mathbf{x}$, and w_k^i, $i = 1, \ldots, N_s$ are normalized weights. The Kernel density is a symmetric probability density function such that

$$\int \mathbf{x} K(\mathbf{x})\, d\mathbf{x} = \mathbf{0}, \qquad \int \|\mathbf{x}\|^2 K(\mathbf{x})\, d\mathbf{x} < \infty.$$

The Kernel $K(\cdot)$ and bandwidth h are chosen to minimize the mean integrated square error (MISE) between the true posterior density and the corresponding regularized empirical representation in (73), which is defined as

$$\text{MISE}(\hat{p}) = \mathbb{E}\left[\int \left[\hat{p}(\mathbf{x}_k|\mathbf{z}_{1:k}) - p(\mathbf{x}_k|\mathbf{z}_{1:k})\right]^2 d\mathbf{x}_k\right] \tag{75}$$

where $\hat{p}(\cdot|\cdot)$ denotes the approximation to $p(\mathbf{x}_k|\mathbf{z}_{1:k})$ given by the right-hand side of (73).[7] In the special case of all the samples having the same weight, the optimal choice of the kernel is the Epanechnikov kernel [31]

$$K_{opt} = \begin{cases} \dfrac{n_x+2}{2c_{n_x}}(1 - \|\mathbf{x}\|^2), & \text{if } \|\mathbf{x}\| < 1 \\ 0, & \text{otherwise} \end{cases} \tag{76}$$

where c_{n_x} is the volume of the unit hypersphere in $\Re^{n_x}$. Furthermore, when the underlying density is Gaussian with a unit covariance matrix, the optimal choice for the bandwidth is [31]

$$h_{opt} = A N_s^{1/(n_x+4)} \tag{77}$$

$$A = \left[8 c_{n_x}^{-1}(n_x+4)\left(2\sqrt{\pi}\right)^{n_x}\right]^{1/(n_x+4)}. \tag{78}$$

Algorithm 6: Regularized Particle Filter

$[\{\mathbf{x}_k^{i*}, w_k^i\}_{i=1}^{N_s}] = \text{RPF}[\{\mathbf{x}_{k-1}^i, w_{k-1}^i\}_{i=1}^{N_s}, \mathbf{z}_k]$

- FOR $i = 1$: N_s
  - Draw $\mathbf{x}_k^i \sim q(\mathbf{x}_k|\mathbf{x}_{k-1}^i, \mathbf{z}_k)$
  - Assign the particle a weight, w_k^i, according to (48)
- END FOR
- Calculate total weight: $t = \text{SUM}[\{w_k^i\}_{i=1}^{N_s}]$
- FOR $i = 1$: N_s
  - Normalize: $w_k^i = t^{-1} w_k^i$
- END FOR
- Calculate $\widehat{N_{eff}}$ using (51)
- IF $\widehat{N_{eff}} < N_T$
  - Calculate the empirical covariance matrix S_k of $\{\mathbf{x}_k^i, w_k^i\}_{i=1}^{N_s}$
  - Compute $\mathbf{D}_k$ such that $\mathbf{D}_k \mathbf{D}_k^T = S_k$.
  - Resample using algorithm 2:
  - $[\{\mathbf{x}_k^i, w_k^i, -\}_{i=1}^{N_s}] = \text{RESAMPLE}[\{\mathbf{x}_k^i, w_k^i\}_{i=1}^{N_s}]$
  - FOR $i = 1$: N_s
  - Draw $\epsilon^i \sim K$ from the Epanechnikov Kernel
  - $\mathbf{x}_k^{i*} = \mathbf{x}_k^i + h_{opt} \mathbf{D}_k \epsilon^i$
  - END FOR
- END IF

Alhough the results of (76) and (77) and (78) are optimal only in the special case of equally weighted particles and underlying Gaussian density, these results can still be used in the general case to obtain a suboptimal filter. One iteration of the RPF is described by Algorithm 6. The RPF only differs from the generic particle filter described by Algorithm 3 as a result of the addition of the regularization steps when conducting the resampling. Note also that the calculation of the empirical covariance matrix

[7] As observed by one of the anonymous reviewers, it is worth noting that the use of the Kernel approximation become increasingly less appropriate as the dimensionality of the state increases.

TABLE I
TABLE OF THE ALGORITHMS USED, THE SECTIONS OF THE ARTICLE, AND FIGURES THAT RELATE TO THE ALGORITHMS, AND RMSE VALUES (AVERAGED OVER 100 MC RUNS)

Algorithm	Proposal	Section	Figures	RMSE
Extended Kalman filter	N/A	IV-A	3,4	23.19
Approximate grid-Based Filter	N/A	IV-B	5	6.09
SIR Particle filter	$p(\mathbf{x}_k\|\mathbf{x}_{k-1}^i)$	V-B.1	6	5.54
Auxiliary Particle filter	$p(\mathbf{x}_k\|\mathbf{x}_{k-1}^i)$	V-B.2	7	5.35
Regularised Particle filter	$p(\mathbf{x}_k\|\mathbf{x}_{k-1}^i)$	V-B.3	8	5.55
'Likelihood' Particle filter	$p(\mathbf{x}_k\|\mathbf{s}_k)p(\mathbf{s}_k\|\mathbf{z}_k)$	A	9	5.30

S_k is carried out prior to the resampling and is therefore a function of both the $\mathbf{x}_k^i$ and w_k^i. This is done since the accuracy of any estimate of a function of the distribution can only decrease as a result of the resampling. If quantities such as the mean and covariance of the samples are to be output, then these should be calculated prior to resampling.

By following the above procedure, we generate an i.i.d. random sample $\{\mathbf{x}_k^{i*}\}_{i=1}^{N_s}$ drawn from (73).

In terms of complexity, the RPF is comparable with SIR since it only requires N_s additional generations from the kernel $K(\cdot)$ at each time step. The RPF has the theoretic disadvantage that the samples are no longer guaranteed to asymtotically approximate those from the posterior. In practical scenarios, the RPF performance is better than the SIR in cases where sample impoverishment is severe, for example, when the process noise is small.

VI. EXAMPLE

Here, we consider the following set of equations as an illustrative example:

$$p(\mathbf{x}_k(\mathbf{x}_{k-1}) = \mathcal{N}(\mathbf{x}_k;\, \mathbf{f}_k(\mathbf{x}_{k-1},\, k),\, Q_{k-1}) \tag{79}$$

$$p(\mathbf{z}_k|\mathbf{x}_k) = \mathcal{N}\left(\mathbf{z}_k;\, \frac{\mathbf{x}_k^2}{20},\, R_k\right) \tag{80}$$

or equivalently

$$\mathbf{x}_k = \mathbf{f}_k(\mathbf{x}_{k-1},\, k) + \mathbf{v}_{k-1} \tag{81}$$

$$\mathbf{z}_k = \frac{\mathbf{x}_k^2}{20} + \mathbf{n}_k \tag{82}$$

where

$$\mathbf{f}_k(\mathbf{x}_{k-1},\, k) = \frac{\mathbf{x}_{k-1}}{2} + \frac{25\mathbf{x}_{k-1}}{1+\mathbf{x}_{k-1}^2} + 8\cos(1.2k) \tag{83}$$

and where $\mathbf{v}_{k-1}$ and $\mathbf{n}_k$ are zero mean Gaussian random variables with variances Q_{k-1} and R_k, respectively. We use $Q_{k-1} = 10$ and $R_k = 1$. This example has been analyzed before in many publications [5], [17], [25].

We consider the performance of the algorithms detailed in Table I. In order to qualitatively gauge performance and discuss resulting issues, we consider one exemplar run. In order to quantify performance, we use the traditional measure of performance: the Root Mean Squared Error (RMSE). It should be noted that this measure of performance is not exceptionally meaningful for this multimodal problem. However, it has been used extensively in the literature and is included here for that reason and because it facilitates quantitative comparison.

For reference, the true states for the exemplar run are shown in Fig. 1 and the measurements in Fig. 2.

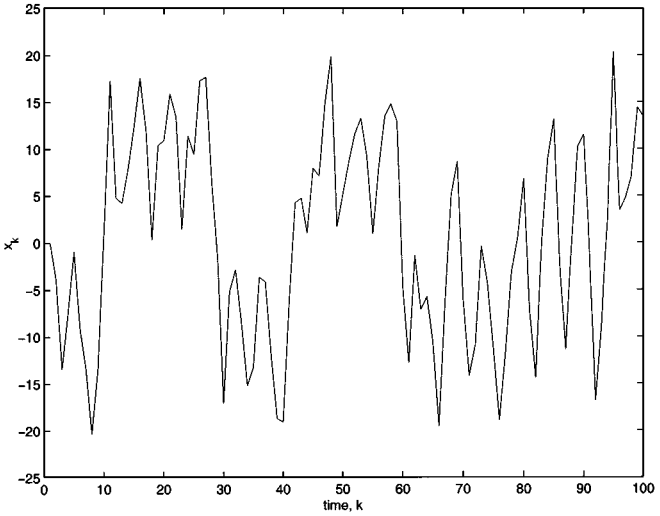


Fig. 1. Figure of the true values of the state $\mathbf{x}_k$ as a function of k for the exemplar run.

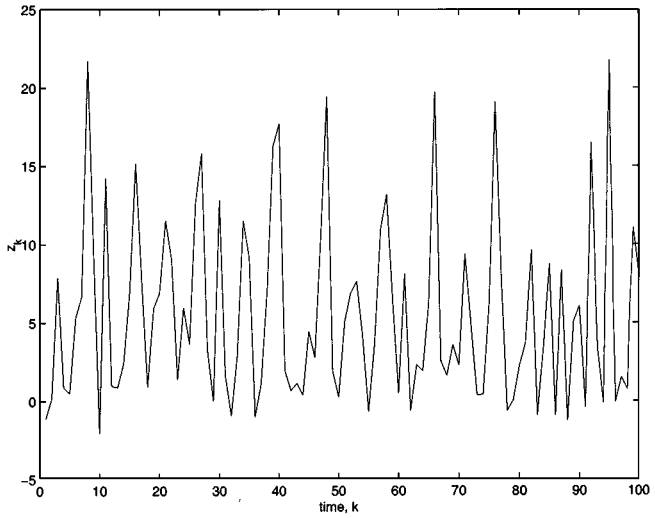


Fig. 2. Figure of the measurements $\mathbf{z}_k$ of the states $\mathbf{x}_k$ shown in Fig. 1 for the same exemplar run.

The approximate grid-based method uses 50 states with centers equally spaced on $[-25,\, 25]$. All the particle filters have 50 particles and employ resampling at every time step ($N_T = N_s$). The auxiliary particle filter uses $\mu_k^i \sim p(\mathbf{x}_k|\mathbf{x}_{k-1}^i)$. The regularized particle filter uses the kernel and bandwidth described in Section V-B3.

To visualize the densities inferred by the approximate grid-based and particle filters, the total probability mass at any time in each of 50 equally spaced regions on $[-25,\, 25]$ is shown as images in Figs. 5–9. At any given time (and in any vertical slice through the image), darker regions represent higher probability than lighter regions. A graduated scale relating intensity to probability mass in a pixel is shown next to each image.

A. EKF

The EKFs local linearization and Gaussian approximation are not a sufficient description of the nonlinear and non-Gaussian nature of the example. Once the EKF cannot adequately approximate the bimodal nature of the underlying

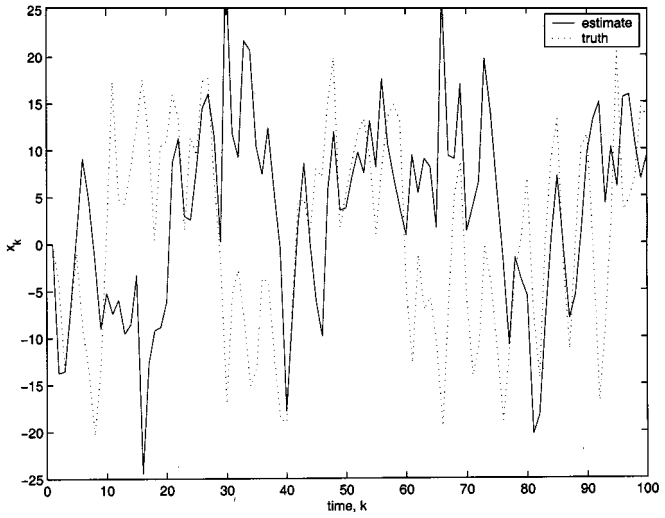


Fig. 3. Evolution of the EKFs mean estimate of the state.

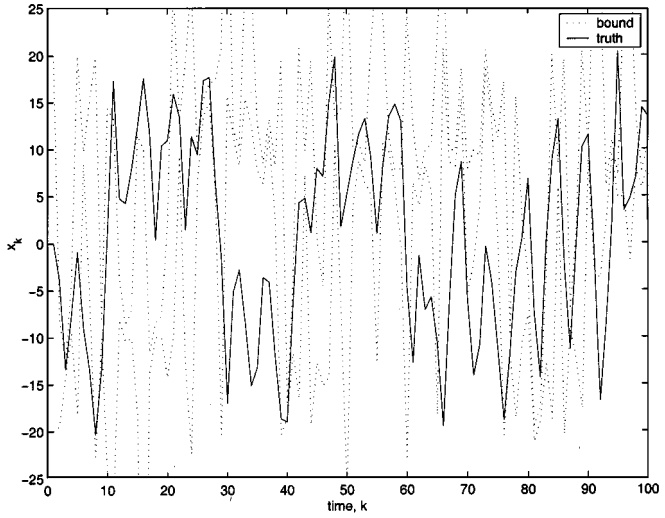


Fig. 4. Evolution of the upper and lower 2σ positions of the state as estimated by the EKF (dotted) with the true state also shown (solid).

posterior, the Gaussian approximation fails—the EKF is prone to either choosing the "wrong" mode or just sitting on the average between the modes. As a result of this inability to adequately approximate the density, the linearization approximation becomes poor.

This can be seen from Fig. 3. The mean of the filter is rarely close to the true state. Were the density to be Gaussian, one would expect the state to be within two standard deviations of the mean approximately 95% of the time. From Fig. 4, it is evident that there are times when the distribution is sufficiently broad to capture the true state in this region but that there are also times when the filter becomes highly overconfident of a biased estimate of the state. The implication of this is that it is very difficult to detect inconsistent EKF errors automatically online.

The RMSE measure indicates that the EKF is the least accurate of the algorithms at approximating the posterior. The approximations made by the EKF are inappropriate in this example.

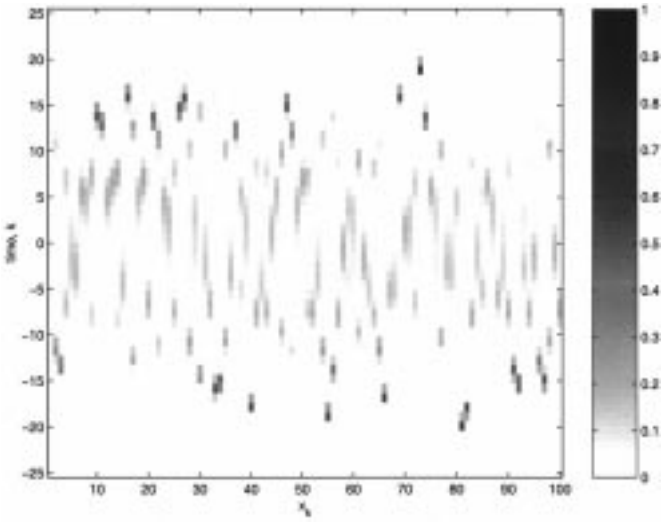


Fig. 5. Image representing evolution of probability density for approximate grid-based filter.

B. Approximate Grid-Based Filter

This example is low dimensional, and therefore, one would expect that an approximate grid-based approach would perform well. Fig. 5 shows this is indeed the case. The grid-based approximation is able to model the multimodality of the problem.

Using the approximate grid-based filter rather than an EKF yields a marked reduction in RMS errors. A particle filter with N_s particles conducts $O(N_s)$ operations per iteration, whereas an approximate grid-based filter carries out $O(N_s^2)$ operations with N_s cells. It is therefore surprising that the RMS errors for the approximate grid are larger than those of the particle filter. The authors suspect that this is an artifact of the grid being fixed; the resolution of the algorithm is predefined, and the fixed position of the grid points means that the grid points near ± 25 contribute significantly to the error when the true state is far from these values.

C. SIR Particle Filter

Using the prior distribution as the importance density is in some sense regarded as a standard SIR particle filter and, therefore, is an appropriate particle filter algorithm with which to begin. As can be seen from Fig. 6, the SIR particle filter gives disappointing results with the low number of particles used here. The speckled appearance of the figure is a result of sampling a low number of particles from the (broad) prior. It is an artifact resulting from the inadequate amount of sampling.

The RMSE metric shows a marginal improvement over the approximate grid-based filter. To achieve smaller errors, one could simply increase the number of particles, but here, we will now investigate the effect of using the alternative particle filter algorithms described up to this point.

D. Auxiliary Particle Filter

One way to reduce errors might be that the proposed particle positions are chosen badly. One might therefore think that choosing the proposed particles in a more intelligent manner would yield better results. An auxiliary particle filter would then

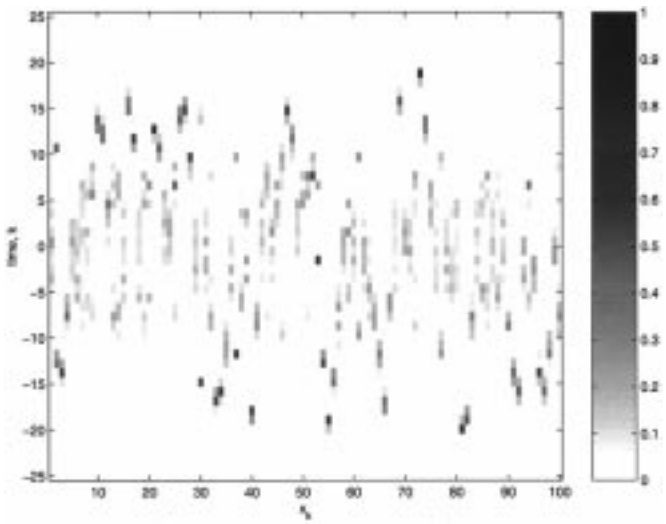


Fig. 6. Image representing evolution of probability density for SIR particle filter.

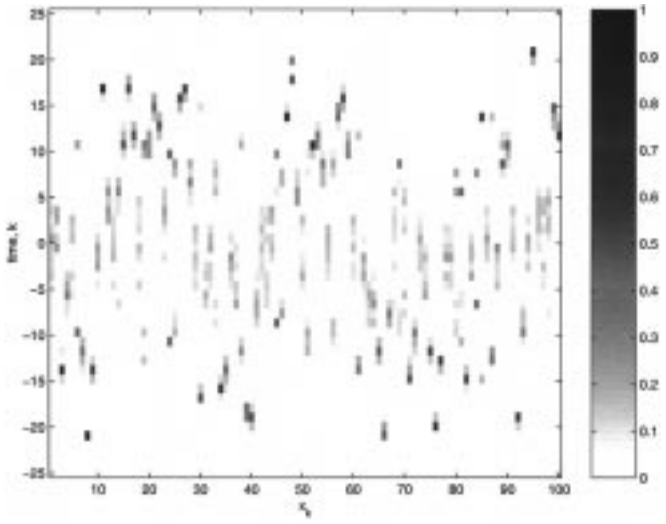


Fig. 7. Image representing evolution of probability density for auxiliary particle filter.

seem to be an appropriate candidate replacement algorithm for SIR. Here, we have μ_k^i as a sample from $p(\mathbf{x}_k|\mathbf{x}_{k-1}^i)$.

As shown by Fig. 7, for this example, the auxiliary particle filter performs well. There is arguably less speckle in Fig. 7 than in Fig. 6, and the probability mass appears to be better concentrated around the true state. However, one might think this problem is not very well suited to an auxiliary particle filter since the prior is often much broader than the likelihood. When the prior is broad, those particles with a noise realization that happens to have a high likelihood are resampled many times. There is no guarantee that other samples from the prior will also lie in the same region of the state space since only a single point is being used to characterize the filtered density for each particle.

The RMS errors are slightly reduced from those for SIR.

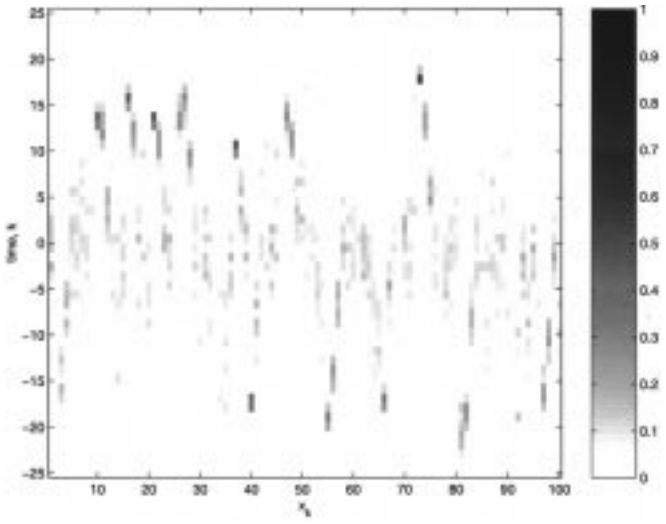


Fig. 8. Image representing evolution of probability density for regularized particle filter.

E. Regularized Particle Filter

Using the regularized particle filter results in a smoothing of the approximation to the posterior. This is apparent from Fig. 8. The speckle is reduced and the peaks broadened when compared with the previous particle filters' images.

The regularized particle filter gives very similar RMS errors to the SIR particle filter. The regularization does not result in a significant reduction in errors for this data set.

F. "Likelihood" Particle Filter

All the aforementioned particle filters share the prior as a proposal density. For this example, much of the time, the likelihood is far tighter than the prior. As a result, the posterior is closer in similarity to the likelihood than to the prior. The importance density is an approximation to the posterior. Therefore, using a better approximation based on the likelihood, rather than the prior, can be expected to improve performance.

Fig. 9 shows that the use of such an importance density (see the Appendix for details) yields a reduction in speckle and that the peaks of the density are closer on average to the true state than for any of the other particle filters.

The RMS errors are similar to those for the Auxiliary particle filter.

G. Crucial Step in the Application of a Particle Filter

The RMS errors indicate that in highly nonlinear environments, a nonlinear filter such as an approximate grid-based filter or particle filter offers an improvement in performance over an EKF. This improvement results from approximating the density rather than the models.

When using a particle filter, one can often expect and frequently achieve an improvement in performance by using far more particles or alternatively by employing regularization or using an auxiliary particle filter. For this example, a slight improvement in RMS errors is possible by using an importance density other than $p(\mathbf{x}_k|\mathbf{x}_{k-1}^i)$. The authors assert that an importance density tuned to a particular problem will yield an appropriate trade off between the number of particles and the com-

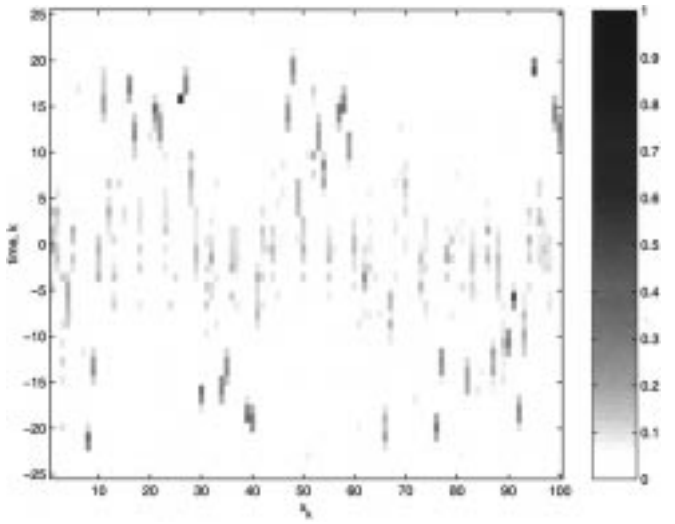


Fig. 9. Image representing evolution of probability density for "likelihood" particle filter.

putational expense necessary for each particle, giving the best qualitative performance with affordable computational effort.

The crucial point to convey is that all the refinements of the particle filter assume that the choice of importance density has already been made. Choosing the importance density to be well suited to a given application requires careful thought. The choice made is crucial.

VII. Conclusions

For a particular problem, if the assumptions of the Kalman filter or grid-based filters hold, then no other algorithm can outperform them. However, in a variety of real scenarios, the assumptions do not hold, and approximate techniques must be employed.

The EKF approximates the models used for the dynamics and measurement process in order to be able to approximate the probability density by a Gaussian. Approximate grid-based filters approximate the continuous state space as a set of discrete regions. This necessitates the predefinition of these regions and becomes prohibitively computationally expensive when dealing with high-dimensional state spaces [3]. Particle filtering approximates the density directly as a finite number of samples. A number of different types of particle filter exist, and some have been shown to outperform others when used for particular applications. However, when designing a particle filter for a particular application, it is the choice of importance density that is critical.

Appendix
Importance Density for "Likelihood" Particle Filter

This Appendix describes the importance density for the "likelihood" particle filter, which is intended to illustrate the crucial nature of the choice of importance density in a particle filter. This importance density is not intended to be generically applicable but to be one chosen to work well for the specific problem and parameters described in Section VI.

To keep the notation simple, throughout this Appendix, $\mathbf{s}_k = (\mathbf{x}_k)^2$. For a uniform prior on $\mathbf{s}_k$, the density $p(\mathbf{s}_k|\mathbf{z}_k)$ can be written by Bayes' rule as

$$p(\mathbf{s}_k|\mathbf{z}_k) \propto \begin{cases} p(\mathbf{z}_k|\mathbf{s}_k), & \mathbf{s}_k \geq 0 \\ 0, & \text{otherwise.} \end{cases} \tag{84}$$

We can then sample $\mathbf{s}_k^i \sim p(\mathbf{s}_k|\mathbf{z}_k)$ [samples $\mathbf{s}_k^i$ are repeatedly drawn from $\hat{p}(\mathbf{s}_k|\mathbf{z}_k) \propto p(\mathbf{z}_k|\mathbf{s}_k)$ until one is drawn such that $p(\mathbf{s}_k|\mathbf{z}_k) > 0$, i.e., one such that $\mathbf{s}_k \geq 0$]. Then, $p(\mathbf{x}_k|\mathbf{s}_k^i)$ can be chosen to be a pair of delta functions

$$p(\mathbf{x}_k|\mathbf{s}_k^i) = \frac{\delta\left(\mathbf{x}_k - \sqrt{\mathbf{s}_k^i}\right) + \delta\left(\mathbf{x}_k + \sqrt{\mathbf{s}_k^i}\right)}{2}. \tag{85}$$

This can then be used to form a "Likelihood" based importance density that samples $\mathbf{x}_k^i$ conditional on $\mathbf{z}_k$ and independently from $\mathbf{x}_{k-1}^i$

$$q(\mathbf{x}_k|\mathbf{x}_{k-1}^i, \mathbf{z}_{1:k}) \propto p(\mathbf{x}_k|\mathbf{s}_k)p(\mathbf{s}_k|\mathbf{z}_k). \tag{86}$$

The weight of the sample can be calculated according to (47)

$$w_k^i \propto w_{k-1}^i \frac{p(\mathbf{z}_k|\mathbf{x}_k^i)p(\mathbf{x}_k^i(\mathbf{x}_{k-1}^i)}{q(\mathbf{x}_k^i(\mathbf{x}_{k-1}^i, |\mathbf{z}_{1:k})} \tag{87}$$

$$= w_{k-1}^i \frac{p(\mathbf{z}_k|\mathbf{x}_k^i)p(\mathbf{x}_k^i(\mathbf{x}_{k-1}^i)}{p(\mathbf{x}_k^i|\mathbf{s}_k^i)p(\mathbf{s}_k^i|\mathbf{z}_k)} \tag{88}$$

$$= w_{k-1}^i \frac{p(\mathbf{x}_k^i|\mathbf{z}_k)p(\mathbf{z}_k)p(\mathbf{x}_k^i(\mathbf{x}_{k-1}^i)}{p(\mathbf{x}_k^i)p(\mathbf{x}_k^i|\mathbf{s}_k^i)p(\mathbf{s}_k^i|\mathbf{z}_k)}. \tag{89}$$

Now, $p(\mathbf{x}_k^i|\mathbf{s}_k^i) = 1/2$, $p(\mathbf{z}_k)$ and $p(\mathbf{x}_k^i)$ are constant; therefore, they disappear, leaving

$$w_k^i \propto w_{k-1}^i p(\mathbf{x}_k^i|\mathbf{x}_{k-1}^i) \frac{p(\mathbf{x}_k^i|\mathbf{z}_k)}{p(\mathbf{s}_k^i|\mathbf{z}_k)}. \tag{90}$$

Now, the ratio of $p(\mathbf{x}_k^i|\mathbf{z}_k)$ to $p(\mathbf{s}_k^i|\mathbf{z}_k)$ needs careful consideration. Although the values of $p(\mathbf{x}_k^i|\mathbf{z}_k)$ and $p(\mathbf{s}_k^i|\mathbf{z}_k)$ might be initially thought to be proportional, they are probability densities defined with respect to a different measure (i.e., a different parameterization of the space). Since $p(\mathbf{x}_k|\mathbf{z}_k)$ integrates to unity over $d\mathbf{x}_k$ while $p(\mathbf{s}_k|\mathbf{z}_k)$ integrates to unity over $d\mathbf{s}_k$, the ratio of the probability densities is then proportional to the inverse of the ratio of the lengths, $d\mathbf{x}_k$ and $d\mathbf{s}_k$. The ratio of $p(\mathbf{x}_k^i|\mathbf{z}_k)$ to $p(\mathbf{s}_k^i|\mathbf{z}_k)$ is the determinant of the Jacobian of the transformation from $\mathbf{s}_k$ to $\mathbf{x}_k$

$$\frac{p(\mathbf{x}_k^i|\mathbf{z}_k)}{p(\mathbf{s}_k^i|\mathbf{z}_k)} \propto \left|\frac{d\mathbf{s}_k}{d\mathbf{x}_k}\right| = 2\mathbf{x}_k. \tag{91}$$

An expression for the weight is then forthcoming:

$$w_k^i \propto w_{k-1}^i p(\mathbf{x}_k^i|\mathbf{x}_{k-1}^i) \frac{p(\mathbf{x}_k^i|\mathbf{z}_k)}{p(\mathbf{s}_k^i|\mathbf{z}_k)} \propto w_{k-1}^i p(\mathbf{x}_k^i|\mathbf{x}_{k-1}^i)\mathbf{x}_k^i. \tag{92}$$

The particle filter that results from this sampling procedure is given in Algorithm 7.

Therefore, rather than draw samples from the state evolution distribution and then weight them according to their likelihood, samples are drawn from the likelihood and then assigned weights on the basis of the state evolution distribution.

Algorithm 7: "Likelihood" Particle Filter

$[\{\mathbf{x}_k^i, w_k^i\}_{i=1}^{N_s}] = \text{LPF}[\{\mathbf{x}_{k-1}^i, w_{k-1}^i\}_{i=1}^{N_s}, \mathbf{z}_k]$

- FOR $i = 1$: N_s
  - REPEAT
  - Draw $\mathbf{s}_k^i \sim \hat{p}(\mathbf{s}_k|\mathbf{z}_k) \propto p(\mathbf{z}_k|\mathbf{s}_k)$
  - UNTIL $\mathbf{s}_k^i \geq 0$
  - IF $\cup[0, 1] > 1/2$
  - $\mathbf{x}_k^i = \sqrt{\mathbf{s}_k^i}$
  - ELSE
  - $\mathbf{x}_k^i = -\sqrt{\mathbf{s}_k^i}$
  - END IF
  - $w_k^i = w_{k-1}^i p(\mathbf{x}_k^i|\mathbf{x}_{k-1}^i)\mathbf{x}_k^i$
- END FOR
- Calculate total weight: $t = \text{SUM}[\{w_k^i\}_{i=1}^{N_s}]$
- FOR $i = 1$: N_s
  - Normalize: $w_k^i = t^{-1} w_k^i$
- END FOR
- Calculate $\widehat{N_{eff}}$ using (51)
- IF $\widehat{N_{eff}} < N_T$
  - Resample using algorithm 2:
  - $[\{\mathbf{x}_k^i, w_k^i, -\}_{i=1}^{N_s}] = \text{RESAMPLE}[\{\mathbf{x}_k^i, w_k^i\}_{i=1}^{N_s}]$
- END IF

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers and the editors of this Special Issue for their many helpful suggestions, which have greatly improved the presentation of this paper. The authors would also like to thank various funding sources who have contributed to this research. N. Gordon would like to thank QinetiQ Ltd.

REFERENCES

[1] S. Arulampalam and B. Ristic, "Comparison of the particle filter with range parameterized and modified polar EKF's for angle-only tracking," *Proc. SPIE*, vol. 4048, pp. 288–299, 2000.

[2] Y. Bar-Shalom and X. R. Li, *Multitarget–Multisensor Tracking: Principles and Techniques*. Urbana, IL: YBS, 1995.

[3] N. Bergman, "Recursive Bayesian estimation: Navigation and tracking applications," Ph.D. dissertation, Linköping Univ., Linköping, Sweden, 1999.

[4] N. Bergman, A. Doucet, and N. Gordon, "Optimal estimation and Cramer–Rao bounds for partial non-Gaussian state space models," *Ann. Inst. Statist. Math.*, vol. 53, no. 1, pp. 97–112, 2001.

[5] B. P. Carlin, N. G. Polson, and D. S. Stoffer, "A Monte Carlo approach to nonnormal and nonlinear state-space modeling," *J. Amer. Statist. Assoc.*, vol. 87, no. 418, pp. 493–500, 1992.

[6] J. Carpenter, P. Clifford, and P. Fearnhead, "Improved particle filter for nonlinear problems," *Proc. Inst. Elect. Eng., Radar, Sonar, Navig.*, 1999.

[7] T. Clapp, "Statistical methods for the processing of communications data," Ph.D. dissertation, Dept. Eng., Univ. Cambridge, Cambridge, U.K., 2000.

[8] T. Clapp and S. Godsill, "Improvement strategies for Monte Carlo particle filters," in *Sequential Monte Carlo Methods in Practice*, A. Doucet, J. F. G. de Freitas, and N. J. Gordon, Eds. New York: Springer-Verlag, 2001.

[9] P. Del Moral, "Measure valued processes and interacting particle systems. Application to nonlinear filtering problems," *Ann. Appl. Probab.*, vol. 8, no. 2, pp. 438–495, 1998.

[10] D. Crisan, P. Del Moral, and T. J. Lyons, "Non-linear filtering using branching and interacting particle systems," *Markov Processes Related Fields*, vol. 5, no. 3, pp. 293–319, 1999.

[11] P. Del Moral, "Non-linear filtering: Interacting particle solution," *Markov Processes Related Fields*, vol. 2, no. 4, pp. 555–580.

[12] A. Doucet, "On sequential Monte Carlo methods for Bayesian filtering," Dept. Eng., Univ. Cambridge, UK, Tech. Rep., 1998.

[13] A. Doucet, J. F. G. de Freitas, and N. J. Gordon, "An introduction to sequential Monte Carlo methods," in *Sequential Monte Carlo Methods in Practice*, A. Doucet, J. F. G. de Freitas, and N. J. Gordon, Eds. New York: Springer-Verlag, 2001.

[14] A. Doucet, S. Godsill, and C. Andrieu, "On sequential Monte Carlo sampling methods for Bayesian filtering," *Statist. Comput.*, vol. 10, no. 3, pp. 197–208.

[15] A. Doucet, N. Gordon, and V. Krishnamurthy, "Particle filters for state estimation of jump Markov linear systems," *IEEE Trans. Signal Processing*, vol. 49, pp. 613–624, Mar. 2001.

[16] S. Godsill, A. Doucet, and M. West, "Methodology for Monte Carlo smoothing with application to time-varying autoregressions," in *Proc. Int. Symp. Frontiers Time Series Modeling*, 2000.

[17] N. Gordon, D. Salmond, and A. F. M. Smith, "Novel approach to nonlinear and non-Gaussian Bayesian state estimation," *Proc. Inst. Elect. Eng., F*, vol. 140, pp. 107–113, 1993.

[18] G. D. Forney, "The Viterbi algorithm," *Proc. IEEE*, vol. 61, pp. 268–278, Mar. 1973.

[19] W. R. Gilks and C. Berzuini, "Following a moving target—Monte Carlo inference for dynamic Bayesian models," *J. R. Statist. Soc. B*, vol. 63, pp. 127–146, 2001.

[20] R. E. Helmick, D. Blair, and S. A. Hoffman, "Fixed-interval smoothing for Markovian switching systems," *IEEE Trans. Inform. Theory*, vol. 41, pp. 1845–1855, Nov. 1995.

[21] Y. C. Ho and R. C. K. Lee, "A Bayesian approach to problems in stochastic estimation and control," *IEEE Trans. Automat. Contr.*, vol. AC-9, pp. 333–339, 1964.

[22] A. H. Jazwinski, *Stochastic Processes and Filtering Theory*. New York: Academic, 1970.

[23] S. Julier, "A skewed approach to filtering," *Proc. SPIE*, vol. 3373, pp. 271–282, 1998.

[24] K. Kanazawa, D. Koller, and S. J. Russell, "Stochastic simulation algorithms for dynamic probabilistic networks," in *Proc. Eleventh Annu. Conf. Uncertainty AI*, 1995, pp. 346–351.

[25] G. Kitagawa, "Monte Carlo filter and smoother for non-Gaussian nonlinear state space models," *J. Comput. Graph. Statist.*, vol. 5, no. 1, pp. 1–25, 1996.

[26] G. Kitagawa and W. Gersch, *Smoothness Priors Analysis of Time Series*. New York: Springer-Verlag, 1996.

[27] J. Liu and M. West, "Combined parameter and state estimation in simulation-based filtering," in *Sequential Monte Carlo Methods in Practice*, A. Doucet, J. F. G. de Freitas, and N. J. Gordon, Eds. New York: Springer-Verlag, 2001.

[28] J. S. Liu and R. Chen, "Sequential Monte Carlo methods for dynamical systems," *J. Amer. Statist. Assoc.*, vol. 93, pp. 1032–1044, 1998.

[29] J. MacCormick and A. Blake, "A probabilistic exclusion principle for tracking multiple objects," in *Proc. Int. Conf. Comput. Vision*, 1999, pp. 572–578.

[30] F. Martinerie and P. Forster, "Data association and tracking using hidden Markov models and dynamic programming," in *Proc. Conf. ICASSP*, 1992.

[31] C. Musso, N. Oudjane, and F. LeGland, "Improving regularised particle filters," in *Sequential Monte Carlo Methods in Practice*, A. Doucet, J. F. G. de Freitas, and N. J. Gordon, Eds. New York: Springer-Verlag, 2001.

[32] M. Orton and A. Marrs, "Particle filters for tracking with out-of-sequence measurements," *IEEE Trans. Aerosp. Electron. Syst.*, submitted for publication.

[33] N. Oudjane and C. Musso, "Progressive correction for regularized particle filters," in *Proc. 3rd Int. Conf. Inform. Fusion*, 2000.

[34] M. Pitt and N. Shephard, "Filtering via simulation: Auxiliary particle filters," *J. Amer. Statist. Assoc.*, vol. 94, no. 446, pp. 590–599, 1999.

[35] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, no. 2, pp. 257–285, Feb. 1989.

[36] L. R. Rabiner and B. H. Juang, "An introduction to hidden Markov models," *IEEE Acoust., Speech, Signal Processing Mag.*, pp. 4–16, Jan. 1986.

[37] B. Ripley, *Stochastic Simulation*. New York: Wiley, 1987.

[38] R. H. Shumway and D. S. Stoffer, "An approach to time series smoothing and forecasting using the EM algorithm," *J. Time Series Anal.*, vol. 3, no. 4, pp. 253–264, 1982.

[39] R. L. Streit and R. F. Barrett, "Frequency line tracking using hidden Markov models," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 38, pp. 586–598, Apr. 1990.

[40] R. van der Merwe, A. Doucet, J. F. G. de Freitas, and E. Wan, "The unscented particle filter," *Adv. Neural Inform. Process. Syst.*, Dec. 2000.
[41] M. West and J. Harrison, "Bayesian forecasting and dynamic models," in *Springer Series in Statistics*, 2nd ed. New York: Springer-Verlag, 1997.
[42] E. A. Wan and R. Van der Merwe, "The unscented Kalman filter for nonlinear estimation," in *Proc. Symp. Adaptive Syst. Signal Process., Commun. Contr.*, Lake Louise, AB, Canada, Oct. 2000.
[43] ——, "The unscented Kalman filter," in Kalman Filtering and Neural Networks. New York: Wiley, 2001, ch. 7, to be published.

M. Sanjeev Arulampalam received the B.Sc. degree in mathematical sciences and the B.E. degree with first-class honors in electrical and electronic engineering from the University of Adelaide, Adelaide, Australia, in 1991 and 1992, respectively. In 1993, he won a Telstra Postgraduate Fellowship award and received the Ph.D. degree in electrical and electronic engineering at the University of Melbourne, Parkville, Australia, in 1997. His doctoral dissertation was "Performance analysis of hidden Markov model based tracking algorithms."

In 1992, he joined the staff of Computer Sciences of Australia (CSA), where he worked as a Software Engineer in the Safety Critical Software Systems Group. In 1998, he joined the Defence Science and Technology Organization (DSTO), Canberra, Australia, as a Research Scientist in the Surveillance Systems Division, where he carried out research in many aspects of airborne target tracking with a particular emphasis on tracking in the presence of deception jamming. His research interests include estimation theory, target tracking, and sequential Monte Carlo methods.

Dr. Arulampalam won the Anglo–Australian postdoctoral fellowship, awarded by the Royal Academy of Engineering, London, in 1998.

Simon Maskell received the B.A. degree in engineering and the M.Eng. degree in electronic and information sciences from Cambridge University Engineering Department (CUED), Cambridge, U.K., both in 1999. He currently pursuing the Ph.D. degree at CUED.

He is with the Pattern and Information Processing Group, QinetiQ Ltd., Malvern, U.K. His research interested include Bayesian inference, signal processing, tracking, and data fusion, with particular emphasis on the application of particle filters.

Mr. Maskell was awarded a Royal Commission for the Exhibition of 1851 Industrial Fellowship in 2001.

Neil Gordon received the B.Sc. degree in mathematics and physics from Nottingham University, Nottingham, U.K., in 1988 and the Ph.D. degree in statistics from Imperial College, University of London, London, U.K., in 1993.

He is currently with the Pattern and Information Processing Group, QinetiQ Ltd., Malvern, U.K. His research interests include Bayesian estimation and sequential Monte Carlo methods (a.k.a. particle filters) with a particular emphasis on target tracking and missile guidance. He has co-edited, with A. Doucet and J. F. G. de Freitas, *Sequential Monte Carlo Methods in Practice* (New York: Springer-Verlag).

Tim Clapp received the B.A., M.Eng., and Ph.D. degrees from the Signal Processing and Communications Group, Cambridge University Engineering Department, Cambridge, U.K.

His research interests include blind equalization, Markov chain Monte Carlo techniques, and particle filters. He is currently involved with telecommunications satellite system design for the Payload Processor Group, Astrium Ltd., Stevenage, U.K.

The Ensemble Kalman Filter for Combined State and Parameter Estimation

MONTE CARLO TECHNIQUES FOR DATA ASSIMILATION IN LARGE SYSTEMS

GEIR EVENSEN

The ensemble Kalman filter (EnKF) [1] is a sequential Monte Carlo method that provides an alternative to the traditional Kalman filter (KF) [2], [3] and adjoint or four-dimensional variational (4DVAR) methods [4]–[6] to better handle large state spaces and nonlinear error evolution. EnKF provides a simple conceptual formulation and ease of implementation, since

PHOTO COMPOSITE COURTESY OF DAVID STENSRUD

there is no need to derive a tangent linear operator or adjoint equations, and there are no integrations backward in time. EnKF is used extensively in a large community, including ocean and atmospheric sciences, oil reservoir simulations, and hydrological modeling.

To a large extent EnKF overcomes two problems associated with the traditional KF. First, in KF an error covariance matrix for the model state needs to be stored and propagated in time, making the method computationally infeasible for models with high-dimensional state vectors. Second, when the model dynamics are nonlinear, the extended KF (EKF) uses a linearized equation for the error covariance evolution, and this linearization can result in unbounded linear instabilities for the error evolution [7].

Digital Object Identifier 10.1109/MCS.2009.932223

1066-033X/09/$25.00©2009IEEE

In contrast with EKF, EnKF represents the error covariance matrix by a large stochastic ensemble of model realizations. For large systems, the dimensionality problem is managed by using a low-rank approximation of the error covariance matrix, where the number of independent model realizations is less than the number of unknowns in the model. Thus, the uncertainty is represented by a set of model realizations rather than an explicit expression for the error covariance matrix. The ensemble of model states is integrated forward in time to predict error statistics. For linear models the ensemble integration is consistent with the exact integration of an error covariance equation in the limit of an infinite ensemble size. Furthermore, for nonlinear dynamical models, the use of an ensemble integration leads to full nonlinear evolution of the error statistics, which in EnKF can be computed with a much lower computational cost than in EKF [8].

Whenever measurements are available, each individual realization is updated to incorporate the new information provided by the measurements. Implementations of the update schemes can be formulated as either a stochastic [9] or a deterministic scheme [8], [10]–[13]. Both kinds of schemes solve for a variance-minimizing solution and implicitly assume that the forecast error statistics are Gaussian by using only the ensemble covariance in the update equation.

The assumption of Gaussian distributions in EnKF allows for a linear and efficient update equation to be used. A more sophisticated update scheme needs to be derived to take into account higher order statistics, which leads to particle filtering theory [14], where the Bayes formula is solved at each update step, although normally at a huge computational cost. While the particle filter accounts for non-Gaussian distributions by representing the full pdf in the parameter space, its applicability is normally limited to estimation of a few unknowns at the cost of integrating a very large ensemble consisting of typically more than $\mathcal{O}(10^4)$ realizations.

In [8], EnKF is rederived as a sequential Monte Carlo method starting from a Bayesian formulation. The EnKF can then be characterized as a special case of the particle filter, where the Bayesian update step in the particle filter is approximated with a linear update step in the EnKF using only the two first moments of the predicted probability density function (pdf). With linear dynamics, EnKF is equivalent to a particle filter, since this case is fully described by Gaussian pdfs. However, with nonlinear dynamics, non-Gaussian contributions may develop, and the EnKF only approximates the particle filter. Unlike the particle filters [14], EnKF does not need to re-sample the ensemble from the posterior pdf during the analysis step, since each prior model realization is individually updated to create the correct posterior ensemble.

In EnKF, the solution is solved for in the affine space spanned by the ensemble of realizations. The ensemble, which evolves in time according to the nonlinear dynamical model, provides a representation of the subspace where the update is computed at each analysis time. It is possible to formulate analysis schemes in terms of the ensemble, leading to efficient algorithms where the state error covariance matrix is not computed and is only implicitly used.

A major approximation introduced in EnKF is related to the use of a limited number of ensemble realizations. The ensemble size limits the space where the solution is searched for and in addition introduces spurious correlations that lead to excessive decrease of the ensemble variance and possibly filter divergence. The spurious correlations can be handled by localization methods that attempt to reduce the impact of measurements that are located far from the gridpoint to be updated. Localization methods either filter away distant measurements or attempt to reduce the amplitude of the long-range spurious correlations. The use of a local analysis scheme effectively increases the ensemble solution space while reducing the impact of spurious correlations. The use of a local analysis scheme allows for a relatively small ensemble size to be used with a high-dimensional dynamical model.

A chronological list of applications of EnKF is given in [8]. This list includes both low-dimensional systems of highly nonlinear dynamical models as well as high-dimensional ocean and atmospheric circulation models with $\mathcal{O}(10^6)$ or more unknowns. Applications include state estimation in operational circulation models for the ocean and atmosphere as well as parameter estimation or history matching in reservoir simulation models. For example, [15]–[17] present an implementation of an EnKF with an isopycnal ocean general circulation model, while [18] examines an implementation of a local EnKF with a state-of-the-art operational numerical weather prediction model using simulated measurements. It is shown that a modest-sized ensemble of 40 members can track the evolution of the atmospheric state with high accuracy.

An implementation of the EnKF at the Canadian Meteorological Centre in [19] demonstrates EnKF for operational atmospheric data assimilation and reviews EnKF with focus on localization and sampling errors. A review in [20] of a variant of EnKF called the local ensemble transform Kalman filter includes a derivation of the analysis equations and the numerical implementation, which differ somewhat from what is normally used in the Kalman filtering literature. An implementation of the local ensemble transform Kalman filter with the National Centers for Environmental Prediction (NCEP) global model, given in [21], concludes that the accuracy of the method is competitive with operational algorithms and that this technique can efficiently handle large number of measurements.

An implementation of EnKF with the NCEP model in [22] is compared with the operational NCEP global data assimilation system. The ensemble data assimilation system outperforms a reduced-resolution version of the operational three-dimensional variational (3DVAR) data assimilation

system and shows improvement in data sparse regions. An observation-thinning algorithm is presented in [22], where observations with little information content leading to low variance reduction are filtered out. The thinning algorithm improves the analysis when unmodeled error correlations are present between nearby observations. The need for the thinning is eliminated if the error correlations are properly specified in the measurement error covariance matrix.

EnKF is currently used in several research fields in addition to the ocean and atmosphere applications cited throughout this article. In [23], the EnKF is used to update a model of tropospheric ozone concentrations and to compute short-term air quality forecasts. It is found that the EnKF updated estimates provide improved initial conditions and lead to better forecasts of the next day's ozone concentration maxima. In [24], EnKF is applied to a magnetohydrodynamic model for space weather prediction. The performance of EnKF in a land surface data assimilation experiment is examined in [25]. These results are compared with a sequential importance re-sampling (SIR) filter, and it is found that EnKF performs almost as well as the SIR filter. Furthermore, it is emphasized that EnKF leads to skewed and even multimodal distributions despite the normality assumption imposed when computing the analysis updates.

In this article, we outline the theory behind the EnKF and demonstrate its use in various high-dimensional and nonlinear applications in mathematical physics while also considering the combined parameter and state estimation problem in some detail. The goal of this article is to serve as an introduction and tutorial for new users of EnKF. We thus present examples that illustrate particular properties of the EnKF, such as its capability to handle high-dimensional state spaces as well as highly nonlinear dynamics.

DATA ASSIMILATION AND PARAMETER ESTIMATION

Given a dynamical model with initial and boundary conditions and a set of measurements that can be related to the model state, the state estimation problem is defined as finding the estimate of the model state that in some weighted measure best fits the model equations, the initial and boundary conditions, and the observed data. Unless we relax the equations and allow some or all of the dynamical model, the conditions, and the measurements to contain errors, the problem may become overdetermined and no general solution exists.

We often use a prior assumption of Gaussian distributions for the error terms. It is also common to assume that errors in the measurements are uncorrelated with errors in the dynamical model. The problem can then be formulated by using a quadratic cost function whose minimum defines the best estimate of the state.

The parameter estimation problem is different from the state estimation problem. Traditionally, in parameter estimation we want to improve estimates of a set of poorly known model parameters leading to an exact model solution that is close to the measurements. Thus, in this case we assume that all errors in the model equations are associated with uncertainties in the selected model parameters. The model initial conditions, boundary conditions, and the model structure are all exactly known. Thus, for any set of model parameters the corresponding solution is found from a single forward integration of the model. The way forward is then to define a cost function that measures the distance between the model prediction and the observations plus a term measuring the deviation of the parameter values from a prior estimate of the parameter values. The relative weight between these two terms is determined by the prior error statistics for the measurements and the prior parameter estimate. Unfortunately, these problems are often hard to solve [8] since the inverse problem is highly nonlinear, and multiple local minima may be present in the cost function.

In [8] the combined parameter and state estimation problem is considered. An improved state estimate and a set of improved model parameters are then searched for simultaneously. In [26] and [27] this problem is formulated using a variational cost function that is minimized using the representer method [28], [29]. Both [26] and [27] report convergence problems due to the nonlinearity of the problem and the possible presence of multiple local minima in the cost function. In [8] it is shown that the combined parameter and state estimation problem can be formulated, and in many cases solved efficiently, using ensemble methods. An illustrative application of the EnKF for combined state and parameter estimation includes estimation of the permeability fields together with dynamic state variables in reservoir simulation models [30]. These problems have huge parameter and state spaces with $\mathcal{O}(10^6)$ unknowns. The formulation and solution of the combined parameter and state estimation problem using ensemble methods are further discussed below.

REVIEW OF THE KALMAN FILTER

Variance Minimizing Analysis Scheme

The KF is a variance-minimizing algorithm that updates the state estimate whenever measurements are available. The update equations in the KF are normally derived by minimizing the trace of the posterior error covariance matrix. The algorithm refers only to first- and second-order statistical moments. With the assumption of Gaussian priors for the model prediction and the data, the update equation can also be derived as the minimizing solution of a quadratic cost function. We start with a vector of variables stored in $\psi(\boldsymbol{x}, t)$, which is defined on some spatial domain $\partial\mathcal{D}$ with spatial coordinate $\boldsymbol{x}$. When $\psi(\boldsymbol{x}, t)$ is discretized on a numerical grid representing the spatial

model domain, it can be represented by the state vector ψ_k at each time instant t_k. The cost function can then be written as

$$\mathcal{J}[\psi_k^{\mathrm{a}}] = (\psi_k^{\mathrm{f}} - \psi_k^{\mathrm{a}})^{\mathrm{T}} (\boldsymbol{C}_{\psi\psi}^{\mathrm{f}})_k^{-1} (\psi_k^{\mathrm{f}} - \psi_k^{\mathrm{a}}) + (\boldsymbol{d}_k - \boldsymbol{M}_k \psi_k^{\mathrm{a}})^{\mathrm{T}} (\boldsymbol{C}_{\epsilon\epsilon})_k^{-1} (\boldsymbol{d}_k - \boldsymbol{M}_k \psi_k^{\mathrm{a}}), \quad (1)$$

where ψ_k^{a} and ψ_k^{f} are the analyzed and forecast estimates respectively, $\boldsymbol{d}_k$ is the vector of measurements, $\boldsymbol{M}_k$ is the measurement operator that maps the model state ψ_k to the measurements $\boldsymbol{d}_k$, $(\boldsymbol{C}_{\psi\psi}^{\mathrm{f}})_k$ is the error covariance of the predicted model state, and $(\boldsymbol{C}_{\epsilon\epsilon})_k$ is the measurement error covariance matrix. Minimizing with respect to ψ_k^{a} yields the classical KF update equations

$$\psi_k^{\mathrm{a}} = \psi_k^{\mathrm{f}} + \boldsymbol{K}_k (\boldsymbol{d}_k - \boldsymbol{M}_k \psi_k^{\mathrm{f}}), \quad (2)$$

$$(\boldsymbol{C}_{\psi\psi})_k^{\mathrm{a}} = (\boldsymbol{I} - \boldsymbol{K}_k \boldsymbol{M}_k)(\boldsymbol{C}_{\psi\psi})_k^{\mathrm{f}}, \quad (3)$$

$$\boldsymbol{K}_k = (\boldsymbol{C}_{\psi\psi})_k^{\mathrm{f}} \boldsymbol{M}_k^{\mathrm{T}} (\boldsymbol{M}_k (\boldsymbol{C}_{\psi\psi})_k^{\mathrm{f}} \boldsymbol{M}_k^{\mathrm{T}} + (\boldsymbol{C}_{\epsilon\epsilon})_k)^{-1}, \quad (4)$$

where the matrix $\boldsymbol{K}_k$ is the Kalman gain. Thus, both the model state and its error covariance are updated.

Kalman Filter

It is assumed that the true state ψ^{t} evolves in time according to the dynamical model

$$\psi_k^{\mathrm{t}} = \boldsymbol{F} \psi_{k-1}^{\mathrm{t}} + \boldsymbol{q}_{k-1}, \quad (5)$$

where $\boldsymbol{F}$ is a linear model operator and $\boldsymbol{q}_{k-1}$ is the unknown model error over one time step from $k-1$ to k. In this case a numerical model evolves according to

$$\psi_k^{\mathrm{f}} = \boldsymbol{F} \psi_{k-1}^{\mathrm{a}}, \quad (6)$$

where the superscripts a and f denote analysis and forecast. That is, given the best possible estimate (traditionally named analysis) for ψ at time t_{k-1}, a forecast is calculated at time t_k, using the approximate equation (6).

The error covariance equation is derived by subtracting (6) from (5), squaring the result, and taking the expectation, which yields

$$\boldsymbol{C}_{\psi\psi}^{\mathrm{f}}(t_k) = \boldsymbol{F} \boldsymbol{C}_{\psi\psi}^{\mathrm{a}}(t_{k-1}) \boldsymbol{F}^{\mathrm{T}} + \boldsymbol{C}_{qq}(t_{k-1}), \quad (7)$$

where we define the error covariance matrices for the predicted and analyzed estimates as

$$\boldsymbol{C}_{\psi\psi}^{\mathrm{f}} = \overline{(\psi^{\mathrm{f}} - \psi^{\mathrm{t}})(\psi^{\mathrm{f}} - \psi^{\mathrm{t}})^{\mathrm{T}}}, \quad (8)$$

$$\boldsymbol{C}_{\psi\psi}^{\mathrm{a}} = \overline{(\psi^{\mathrm{a}} - \psi^{\mathrm{t}})(\psi^{\mathrm{a}} - \psi^{\mathrm{t}})^{\mathrm{T}}}. \quad (9)$$

The overline denotes an expectation operator, which is equivalent to averaging over an ensemble of infinite size.

Extended Kalman Filter

We now assume a nonlinear model, where the true state vector ψ_k^{t} at time t_k is calculated from

$$\psi_k^{\mathrm{t}} = \boldsymbol{G}(\psi_{k-1}^{\mathrm{t}}) + \boldsymbol{q}_{k-1}, \quad (10)$$

and a forecast is calculated from the approximate equation

$$\psi_k^{\mathrm{f}} = \boldsymbol{G}(\psi_{k-1}^{\mathrm{a}}). \quad (11)$$

The error statistics then evolve according to the equation

$$\boldsymbol{C}_{\psi\psi}^{\mathrm{f}}(t_k) = \boldsymbol{G}_{k-1}' \boldsymbol{C}_{\psi\psi}^{\mathrm{a}}(t_{k-1}) \boldsymbol{G}_{k-1}'^{\mathrm{T}} + \boldsymbol{C}_{qq}(t_{k-1}) + \cdots, \quad (12)$$

where $\boldsymbol{C}_{qq}(t_{k-1})$ is the model error covariance matrix and $\boldsymbol{G}_{k-1}'$ is the Jacobian or tangent linear operator given by

$$\boldsymbol{G}_{k-1}' = \left. \frac{\partial \boldsymbol{G}(\psi)}{\partial \psi} \right|_{\psi_{k-1}}. \quad (13)$$

Note that in (12) we neglect an infinite number of terms containing higher order statistical moments and higher order derivatives of the model operator. EKF is based on the assumption that the contributions from all of the higher order terms are negligible. By discarding these terms we are left with the approximate error covariance expression

$$\boldsymbol{C}_{\psi\psi}^{\mathrm{f}}(t_k) \simeq \boldsymbol{G}_{k-1}' \boldsymbol{C}_{\psi\psi}^{\mathrm{a}}(t_{k-1}) \boldsymbol{G}_{k-1}'^{\mathrm{T}} + \boldsymbol{C}_{qq}(t_{k-1}). \quad (14)$$

Higher order approximations for the error covariance evolution are discussed in [31].

EKF with a Nonlinear Ocean Circulation Model

As an application of EKF we consider a nonlinear ocean circulation model [7]. The model in Figure 1 is a multilayer quasi-geostrophic model of the mesoscale ocean currents. The quasi-geostrophic model solves simplified fluid equations for the slow motions in the ocean and are formulated in terms of potential vorticity advection in a background velocity field represented by a stream function. Given a change in the vorticity field, at each time step we can solve for the corresponding stream function.

It is found that the linear evolution equation for the error covariance matrix leads to a linear instability. This instability is demonstrated in an experiment using a steady background flow defined by an eddy standing on a flat bathymetry [see Figure 1(a)]. This particular stream function results in a velocity shear and thus supports a sheared flow instability. Thus, if we add a perturbation and advect it using the linearized equations, then the perturbation grows exponentially. This growth is exactly what is observed in Figure 1(b) and (c). By choosing an initial variance equal to one throughout the model domain, we observe strong

error-variance growth at locations of large velocity and velocity shear in the eddy. The estimated mean square errors, which equal the trace of $C_{\psi\psi}$ divided by the number of gridpoints, indicate exponential error-variance growth.

This linear instability is not realistic. In the real world we expect the instability to saturate at a certain climatological amplitude. As an example, in the atmosphere it is always possible to define a maximum and minimum pressure, which is never exceeded, and the same applies for the eddy field in the ocean. An unstable variance growth cannot be accepted but is in fact what the EKF provides in some cases.

Thus, an apparent closure problem is present in the error-covariance evolution equation, caused by discarding third- and higher order moments in the error covariance equation, leading to a linear instability. If a correct equation could be used to predict the time evolution of the errors, then linear instabilities would saturate due to nonlinear effects. This saturation is missing in EKF, as confirmed by [32]–[34].

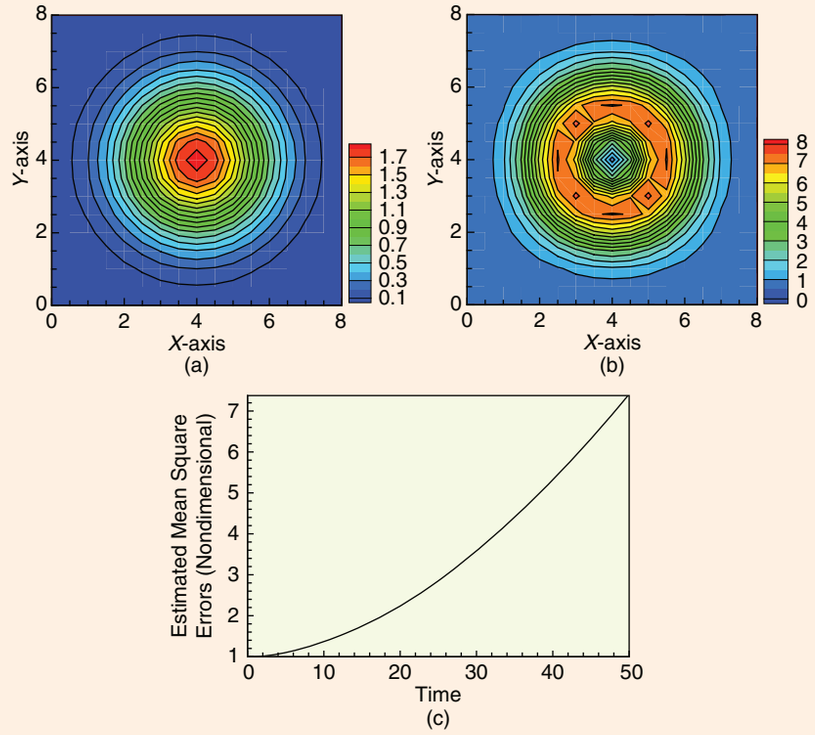


FIGURE 1 Example of an extended Kalman filter experiment from [7]. (a) shows the stream function defining the velocity field of a stationary eddy, while (b) shows the resulting error variance in the model domain after integration from $t = 0$ to $t = 25$. Note the large errors at locations where velocities are high. (c) shows the exponential time evolution of the estimated variance averaged over the model domain.

Extended Kalman Filter for the Mean

Equations (11), (13), and (14) are the most commonly used for EKF. A weakness of the formulation is that the central forecast is used as the estimate. The central forecast is the single model realization initialized with the expected value of the initial state and then integrated by the dynamical model and updated at the measurement steps. For nonlinear dynamics the central forecast may not be equal to the expected value, and thus it is just one realization from an infinite ensemble of possible realizations.

An alternative approach is to derive a model for the evolution of the first moment or mean. First $G(\psi)$ is expanded around $\overline{\psi}$ to obtain

$$G(\psi) = G(\overline{\psi}) + G'(\overline{\psi})(\psi - \overline{\psi}) + \frac{1}{2}G''(\overline{\psi})(\psi - \overline{\psi})^2 + \cdots. \tag{15}$$

Inserting (15) in (11) and taking the expectation or ensemble average yields

$$\overline{\psi_k} = G(\overline{\psi_{k-1}}) + \frac{1}{2}G''_{k-1}C_{\psi\psi}(t_{k-1}) + \cdots. \tag{16}$$

It can be argued that for a statistical estimator it makes more sense to work with the mean than a central forecast. After all, the central forecast does not have any statistical interpretation as illustrated by running an atmospheric model without assimilation updates. The central forecast then becomes just one realization out of infinitely many possible realizations, and it is not clear how we can relate the central forecast to the climatological error covariance estimate. On the other hand the equation for the mean provides an estimate that converges to the climatological mean, and the covariance estimate thus describes the error variance of the climatological mean. All applications of the EKF for data assimilation in ocean and atmospheric models use an equation for the central forecast. However, the interpretation using the equation for the mean supports the formulation used in EnKF.

ENSEMBLE KALMAN FILTER

We begin by representing the error statistics using an ensemble of model states. Next, we present an alternative to the traditional error covariance equation for predicting error statistics. Finally, we derive the traditional EnKF analysis scheme.

Representation of Error Statistics

The error covariance matrices $\boldsymbol{C}^{\mathrm{f}}_{\psi\psi}$ and $\boldsymbol{C}^{\mathrm{a}}_{\psi\psi}$ for the predicted and analyzed estimate in the Kalman filter are defined in terms of the true state in (8) and (9). However, since the true state is not known, we define the ensemble covariance matrices around the ensemble mean $\overline{\psi}$ according to

$$(\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{f}} = \overline{(\psi^{\mathrm{f}} - \overline{\psi^{\mathrm{f}}})(\psi^{\mathrm{f}} - \overline{\psi^{\mathrm{f}}})^{\mathrm{T}}}, \tag{17}$$

$$(\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{a}} = \overline{(\psi^{\mathrm{a}} - \overline{\psi^{\mathrm{a}}})(\psi^{\mathrm{a}} - \overline{\psi^{\mathrm{a}}})^{\mathrm{T}}}, \tag{18}$$

where now the overline denotes an average over the ensemble. Thus, we can use an interpretation where the ensemble mean is the best estimate and the spreading of the ensemble around the mean is a natural definition of the error in the ensemble mean.

Thus, instead of storing a full covariance matrix, we can represent the same error statistics using an appropriate ensemble of model states. Given an error covariance matrix, an ensemble of finite size provides an approximation to the error covariance matrix, and, as the size N of the ensemble increases, the errors in the Monte Carlo sampling decrease proportionally to $1/\sqrt{N}$.

Suppose now that we have N model states or realizations in the ensemble, each of dimension n. Each realization can be represented as a single point in an n-dimensional state space, while together the realizations constitute a cloud of such points. In the limit as N goes to infinity, the cloud of points can be described using the pdf

$$f(\psi) = \frac{dN}{N}, \tag{19}$$

where dN is the number of points in a small unit volume and N is the total number of points. Statistical moments can then be calculated from either $f(\psi)$ or the ensemble representing $f(\psi)$.

Prediction of Error Statistics

A nonlinear model that contains stochastic errors can be written as the stochastic differential equation

$$d\psi = \boldsymbol{G}(\psi)dt + \boldsymbol{h}(\psi)d\boldsymbol{q}. \tag{20}$$

Equation (20) states that an increment in time yields an increment in ψ, which, in addition, is influenced by a random contribution from the stochastic forcing term $\boldsymbol{h}(\psi)d\boldsymbol{q}$, representing the model errors. The term $d\boldsymbol{q}$ describes a vector Brownian motion process with covariance $\boldsymbol{C}_{qq}dt$. Since the model operator $\boldsymbol{G}$ in (20) is not an explicit function of the random variable $d\boldsymbol{q}$, the Ito interpretation is used rather than the Stratonovich interpretation [35].

When additive Gaussian model errors forming a Markov process are used, it is possible to derive the Fokker-Planck equation (also called Kolmogorov's equation), which describes the time evolution of the pdf $f(\psi)$ of the model state. This equation has the form

$$\frac{\partial f(\psi)}{\partial t} + \sum_i \frac{\partial (g_i f(\psi))}{\partial \psi_i} = \frac{1}{2}\sum_{i,j} \frac{\partial^2 f(\psi)(\boldsymbol{h}\boldsymbol{C}_{qq}\boldsymbol{h}^{\mathrm{T}})_{ij}}{\partial \psi_i \partial \psi_j}, \tag{21}$$

where g_i is the component number i of the model operator $\boldsymbol{G}$ and $\boldsymbol{h}\boldsymbol{C}_{qq}\boldsymbol{h}^{\mathrm{T}}$ is the covariance matrix for the model errors.

The Fokker-Planck equation (21) does not entail any approximations and can be considered as the fundamental equation for the time evolution of the error statistics. A detailed derivation is given in [35]. Equation (21) describes the change of the probability density in a local "volume," which depends on the divergence term describing a probability flux into the local "volume" (impact of the dynamical equation) and the diffusion term, which tends to flatten the probability density due to the effect of stochastic model errors. If (21) could be solved for the pdf, it would be possible to calculate statistical moments such as the mean and the error covariance for the model forecast to be used in the analysis scheme.

A linear model for a Gauss-Markov process, in which the initial condition is assumed to be taken from a normal distribution, has a probability density that is completely characterized by its mean and covariance for all times. We can then derive exact equations for the evolution of the mean and the covariance as a simpler alternative than solving the full Fokker-Planck equation. These moments of (21), including the error covariance (7), are easy to derive, and several methods are illustrated in [35]. The KF uses the first two moments of (21).

For a nonlinear model, the mean and covariance matrix do not in general characterize the time evolution of $f(\psi)$. These quantities do, however, determine the mean path and the width of the pdf about that path, and it is possible to solve approximate equations for the moments, which is the procedure characterizing the EKF.

The EnKF applies a Markov chain Monte Carlo (MCMC) method to solve (21). The probability density is then represented by a large ensemble of model states. By integrating these model states forward in time according to the model dynamics, as described by the stochastic differential equation (20), this ensemble prediction is equivalent to using a MCMC method to solve the Fokker-Planck equation.

Dynamical models can have stochastic terms embedded within the nonlinear model operator, and the derivation of the associated Fokker-Planck equation can become complex. Fortunately, the explicit form of the Fokker-Planck equation is not needed, since, to solve this equation using MCMC methods, it is sufficient to know that the equation and a solution exist.

Analysis Scheme

We now derive the update scheme in the KF using the ensemble covariances as defined by (17) and (18). For

convenience the time index k is omitted in the equations to follow. As shown by [9] it is essential that the observations be treated as random variables having a distribution with mean equal to the observed value and covariance equal to $\boldsymbol{C}_{\epsilon\epsilon}$. Thus, we start by defining an ensemble of observations

$$\boldsymbol{d}_j = \boldsymbol{d} + \boldsymbol{\epsilon}_j, \tag{22}$$

where j counts from one to the number N of ensemble members. By subtracting any nonzero mean from the N samples $\boldsymbol{\epsilon}_j$, it is ensured that the simulated random measurement errors have mean equal to zero and thus the random perturbations do not introduce any bias in the update. Next we define the ensemble covariance matrix of the measurement errors as

$$\boldsymbol{C}^{\mathrm{e}}_{\epsilon\epsilon} = \overline{\boldsymbol{\epsilon}\boldsymbol{\epsilon}^{\mathrm{T}}}, \tag{23}$$

while, in the limit of infinite ensemble size this matrix converges to the prescribed error covariance matrix $\boldsymbol{C}_{\epsilon\epsilon}$ used in the Kalman filter. The following discussion is valid using both an exactly prescribed $\boldsymbol{C}_{\epsilon\epsilon}$ and an ensemble representation $\boldsymbol{C}^{\mathrm{e}}_{\epsilon\epsilon}$ of $\boldsymbol{C}_{\epsilon\epsilon}$, which can be useful in some implementations of the analysis scheme.

The analysis step in EnKF consists of updates performed on each of the ensemble members, as given by

$$\boldsymbol{\psi}^{\mathrm{a}}_j = \boldsymbol{\psi}^{\mathrm{f}}_j + (\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{f}}\boldsymbol{M}^{\mathrm{T}}(\boldsymbol{M}(\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{f}}\boldsymbol{M}^{\mathrm{T}} + \boldsymbol{C}^{\mathrm{e}}_{\epsilon\epsilon})^{-1}(\boldsymbol{d}_j - \boldsymbol{M}\boldsymbol{\psi}^{\mathrm{f}}_j). \tag{24}$$

With a finite ensemble size, the use of the ensemble covariances introduces an approximation of the true covariances. Furthermore, if the number of measurements is larger than the number of ensemble members, then the matrices $\boldsymbol{M}(\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{f}}\boldsymbol{M}^{\mathrm{T}}$ and $\boldsymbol{C}^{\mathrm{e}}_{\epsilon\epsilon}$ are singular, and pseudo inversion must be used.

Equation (24) implies that

$$\overline{\boldsymbol{\psi}^{\mathrm{a}}} = \overline{\boldsymbol{\psi}^{\mathrm{f}}} + (\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{f}}\boldsymbol{M}^{\mathrm{T}}(\boldsymbol{M}(\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{f}}\boldsymbol{M}^{\mathrm{T}} + \boldsymbol{C}^{\mathrm{e}}_{\epsilon\epsilon})^{-1}(\overline{\boldsymbol{d}} - \boldsymbol{M}\overline{\boldsymbol{\psi}^{\mathrm{f}}}), \tag{25}$$

where $\overline{\boldsymbol{d}} = \boldsymbol{d}$ since the measurement perturbations have ensemble mean equal to zero. Thus, the relation between the analyzed and predicted ensemble mean is identical to the relation between the analyzed and predicted state in the standard Kalman filter, apart from the use of $(\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{f,a}}$ and $\boldsymbol{C}^{\mathrm{e}}_{\epsilon\epsilon}$ instead of $\boldsymbol{C}^{\mathrm{f,a}}_{\psi\psi}$ and $\boldsymbol{C}_{\epsilon\epsilon}$. Note that the introduction of an ensemble of observations does not affect the update of the ensemble mean.

It is now shown that, by updating each of the ensemble members using the perturbed observations, we can create a new ensemble with the correct error statistics. We derive the analyzed error covariance estimate resulting from the analysis scheme given above, although we retain the standard Kalman filter form for the analysis equations. First, (24) and (25) are used to obtain

$$\boldsymbol{\psi}^{\mathrm{a}}_j - \overline{\boldsymbol{\psi}^{\mathrm{a}}} = (\boldsymbol{I} - \boldsymbol{K}_{\mathrm{e}}\boldsymbol{M})(\boldsymbol{\psi}^{\mathrm{f}}_j - \overline{\boldsymbol{\psi}^{\mathrm{f}}}) + \boldsymbol{K}_{\mathrm{e}}(\boldsymbol{d}_j - \overline{\boldsymbol{d}}), \tag{26}$$

where we use the Kalman gain

$$\boldsymbol{K}_{\mathrm{e}} = (\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{f}}\boldsymbol{M}^{\mathrm{T}}(\boldsymbol{M}(\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{f}}\boldsymbol{M}^{\mathrm{T}} + \boldsymbol{C}^{\mathrm{e}}_{\epsilon\epsilon})^{-1}. \tag{27}$$

The error covariance update is then derived as

$$\begin{aligned}
(\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{a}} &= \overline{(\boldsymbol{\psi}^{\mathrm{a}} - \overline{\boldsymbol{\psi}^{\mathrm{a}}})(\boldsymbol{\psi}^{\mathrm{a}} - \overline{\boldsymbol{\psi}^{\mathrm{a}}})^{\mathrm{T}}} \\
&= \overline{((\boldsymbol{I} - \boldsymbol{K}_{\mathrm{e}}\boldsymbol{M})(\boldsymbol{\psi}^{\mathrm{f}} - \overline{\boldsymbol{\psi}^{\mathrm{f}}}) + \boldsymbol{K}_{\mathrm{e}}(\boldsymbol{d} - \overline{\boldsymbol{d}}))} \\
&\quad \times \overline{((\boldsymbol{I} - \boldsymbol{K}_{\mathrm{e}}\boldsymbol{M})(\boldsymbol{\psi}^{\mathrm{f}} - \overline{\boldsymbol{\psi}^{\mathrm{f}}}) + \boldsymbol{K}_{\mathrm{e}}(\boldsymbol{d} - \overline{\boldsymbol{d}}))^{\mathrm{T}}} \\
&= (\boldsymbol{I} - \boldsymbol{K}_{\mathrm{e}}\boldsymbol{M})\overline{(\boldsymbol{\psi}^{\mathrm{f}} - \overline{\boldsymbol{\psi}^{\mathrm{f}}})(\boldsymbol{\psi}^{\mathrm{f}} - \overline{\boldsymbol{\psi}^{\mathrm{f}}})^{\mathrm{T}}}(\boldsymbol{I} - \boldsymbol{K}_{\mathrm{e}}\boldsymbol{M})^{\mathrm{T}} \\
&\quad + \boldsymbol{K}_{\mathrm{e}}\overline{(\boldsymbol{d} - \overline{\boldsymbol{d}})(\boldsymbol{d} - \overline{\boldsymbol{d}})^{\mathrm{T}}}\boldsymbol{K}^{\mathrm{T}}_{\mathrm{e}} \\
&= (\boldsymbol{I} - \boldsymbol{K}_{\mathrm{e}}\boldsymbol{M})(\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{f}}(\boldsymbol{I} - \boldsymbol{M}^{\mathrm{T}}\boldsymbol{K}^{\mathrm{T}}_{\mathrm{e}}) + \boldsymbol{K}_{\mathrm{e}}\boldsymbol{C}^{\mathrm{e}}_{\epsilon\epsilon}\boldsymbol{K}^{\mathrm{T}}_{\mathrm{e}} \\
&= (\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{f}} - \boldsymbol{K}_{\mathrm{e}}\boldsymbol{M}(\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{f}} - (\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{f}}\boldsymbol{M}^{\mathrm{T}}\boldsymbol{K}^{\mathrm{T}}_{\mathrm{e}} \\
&\quad + \boldsymbol{K}_{\mathrm{e}}(\boldsymbol{M}(\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{f}}\boldsymbol{M}^{\mathrm{T}} + \boldsymbol{C}^{\mathrm{e}}_{\epsilon\epsilon})\boldsymbol{K}^{\mathrm{T}}_{\mathrm{e}} \\
&= (\boldsymbol{I} - \boldsymbol{K}_{\mathrm{e}}\boldsymbol{M})(\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{f}}.
\end{aligned} \tag{28}$$

The last expression in (28) is the traditional result for the minimum error covariance found in the KF analysis scheme. Thus, (28) implies that EnKF in the limit of an infinite ensemble size gives the same result as KF. It is assumed that the distributions used to generate the model-state ensemble and the observation ensemble are independent. Using a finite ensemble size, neglecting the cross-term introduces sampling errors. Note that the derivation (28) shows that the observations $\boldsymbol{d}$ must be treated as random variables to introduce the measurement error covariance matrix $\boldsymbol{C}^{\mathrm{e}}_{\epsilon\epsilon}$ into the expression. That is,

$$\boldsymbol{C}^{\mathrm{e}}_{\epsilon\epsilon} = \overline{\boldsymbol{\epsilon}\boldsymbol{\epsilon}^{\mathrm{T}}} = \overline{(\boldsymbol{d} - \overline{\boldsymbol{d}})(\boldsymbol{d} - \overline{\boldsymbol{d}})^{\mathrm{T}}}. \tag{29}$$

A full-rank measurement error covariance matrix can be used in (27), but the use of an ensemble representation of the measurement error covariance matrix leads to an exact cancellation in the second last line in (27), which becomes

$$\begin{aligned}
\boldsymbol{K}_{\mathrm{e}}(\boldsymbol{M}(\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{f}}\boldsymbol{M}^{\mathrm{T}} + \boldsymbol{C}^{\mathrm{e}}_{\epsilon\epsilon})\boldsymbol{K}^{\mathrm{T}}_{\mathrm{e}} &= \boldsymbol{K}_{\mathrm{e}}(\boldsymbol{M}(\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{f}}\boldsymbol{M}^{\mathrm{T}} + \boldsymbol{C}^{\mathrm{e}}_{\epsilon\epsilon}) \\
&\quad \times (\boldsymbol{M}(\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{f}}\boldsymbol{M}^{\mathrm{T}} + \boldsymbol{C}^{\mathrm{e}}_{\epsilon\epsilon})^{-1}\boldsymbol{M}(\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{f}} \\
&= \boldsymbol{K}_{\mathrm{e}}\boldsymbol{M}(\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{f}}.
\end{aligned} \tag{30}$$

Thus, we conclude that the use of a low-rank measurement error covariance matrix, represented by the measurement perturbations, when computing the Kalman gain, reduces the sampling errors in EnKF. The remaining sampling errors come from neglecting the cross-correlation term between the measurements and the forecast ensemble, which is nonzero with a final ensemble size, and from the approximation of the state error covariance matrix using a finite ensemble size.

This article provides a fundamental theoretical basis for understanding EnKF and serves as a useful text for future users.

The above derivation assumes that the inverse in the Kalman gain (27) exists. However, the derivation also holds when the matrix in the inversion is of low rank, for example, when the number of measurements is larger than the number of realizations and the low-rank $\boldsymbol{C}^{\mathrm{e}}_{\epsilon\epsilon}$ is used. The inverse in (27) can then be replaced with the pseudoinverse, and we can write the Kalman gain as

$$\boldsymbol{K}_{\mathrm{e}} = (\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{f}}\boldsymbol{M}^{\mathrm{T}}(\boldsymbol{M}(\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{f}}\boldsymbol{M}^{\mathrm{T}} + \boldsymbol{C}^{\mathrm{e}}_{\epsilon\epsilon})^{+}. \tag{31}$$

When the matrix in the inversion is of full rank, (31) becomes identical to (27). Using (31) the expression (30) becomes

$$\begin{aligned}\boldsymbol{K}_{\mathrm{e}}(\boldsymbol{M}(\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{f}}\boldsymbol{M}^{\mathrm{T}} + \boldsymbol{C}^{\mathrm{e}}_{\epsilon\epsilon})\ \mathrm{K}_{\mathrm{e}}^{\mathrm{T}} &= (\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{f}}\boldsymbol{M}^{\mathrm{T}}(\boldsymbol{M}(\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{f}}\boldsymbol{M}^{\mathrm{T}} + \boldsymbol{C}^{\mathrm{e}}_{\epsilon\epsilon})^{+} \\ &\quad \times (\boldsymbol{M}(\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{f}}\boldsymbol{M}^{\mathrm{T}} + \boldsymbol{C}^{\mathrm{e}}_{\epsilon\epsilon}) \\ &\quad \times (\boldsymbol{M}(\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{f}}\boldsymbol{M}^{\mathrm{T}} + \boldsymbol{C}^{\mathrm{e}}_{\epsilon\epsilon})^{+}\boldsymbol{M}(\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{f}} \\ &= (\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{f}}\boldsymbol{M}^{\mathrm{T}}(\boldsymbol{M}(\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{f}}\boldsymbol{M}^{\mathrm{T}} \\ &\quad + \boldsymbol{C}^{\mathrm{e}}_{\epsilon\epsilon})^{+}\boldsymbol{M}(\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{f}} \\ &= \boldsymbol{K}_{\mathrm{e}}\boldsymbol{M}(\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{f}},\end{aligned} \tag{32}$$

where we have used the property $\boldsymbol{Y}^{+} = \boldsymbol{Y}^{+}\boldsymbol{Y}\boldsymbol{Y}^{+}$ of the pseudoinverse.

It should be noted that the EnKF analysis scheme is approximate in the sense that non-Gaussian contributions in the predicted ensemble are not properly taken into account. In other words, the EnKF analysis scheme does not solve the Bayesian update equation for non-Gaussian pdfs. On the other hand, the EnKF analysis scheme is not just a resampling of a Gaussian posterior distribution. Only the updates defined by the right-hand side of (24), which are added to the prior non-Gaussian ensemble, are linear. Thus, the updated ensemble inherits many of the non-Gaussian properties from the forecast ensemble. In summary, we have a computationally efficient analysis scheme where we avoid resampling of the posterior.

Ensemble Kalman Filter with a Linear Advection Equation

The properties of EnKF are now illustrated in a simple example when used with a one-dimensional linear advection model. The model describes general transport in a prescribed background flow on a periodic domain of length 1000 m. The model has the constant advection speed $u = 1$ m/s, the grid spacing $\Delta x = 1$ m, and the time step $\Delta t = 1$ s. Given an initial condition, the solution of this model is exactly known, which facilitates realistic experiments with zero model error to examine the impact of the dynamical evolution of the error covariance.

The true initial state is sampled from a normal distribution $\mathcal{N}$, with mean equal to zero, variance equal to one, and a spatial decorrelation length of 20 m. The first guess solution is generated by drawing another sample from $\mathcal{N}$ and adding this sample to the true state. The initial ensemble of 1000 realizations is generated by adding samples drawn from $\mathcal{N}$ to the first guess solution. Thus, the initial state is assumed to have an error variance equal to one. Four measurements of the true solution, distributed regularly in the model domain, are assimilated every fifth time step. The measurements of the wave amplitude are contaminated by errors of variance equal to 0.01, in nondimensional units, and we assume uncorrelated measurement errors. The length of the integration is 300 s, which is 50 s longer than the time of 250 s needed for the solution to advect from one measurement to the next.

The example in Figure 2 illustrates the convergence of the estimated solution at various times during the experiment. In particular, Figure 2 shows how information from measurements is propagated with the advection speed and how the error variance is reduced each time measurements are assimilated. The first plot shows the result of the first update with the four measurements. Near the measurement locations, the estimated solution is consistent with both the true solution and the measurements, and the error variance is reduced accordingly. The second plot is taken at $t = 150$ s, that is, after 30 updates with measurements. Now the information from the measurements has propagated to the right with the advection speed, as seen both from direct comparison of the estimate with the true solution, as well as from the estimated variance. The final plot, which is taken at $t = 300$ s, shows that the estimate is now in good agreement with the true solution throughout the model domain. Note also the linear increase in error variance to the right of the measurements, which is caused by the addition of model errors at each time step. It is also clear that the estimated solution deteriorates far from the measurements in the advection direction. For linear models with regular measurements at fixed locations and stationary error statistics, the increase of error variance from model errors balances the reduction from the updates with measurements.

Discussion

We now have a complete system of equations that constitute the EnKF, and the similarity with the standard KF is maintained both for the prediction of error covariances

and in the analysis scheme. For linear dynamics the EnKF solution converges exactly to the KF solution with increasing ensemble size.

One of the advantages of EnKF is that, for nonlinear models, the equation for the mean is solved and no closure assumption is used since each ensemble member is integrated by the full nonlinear model. This nonlinear error evolution is contrary to the approximate equation for the mean (16), which is used in EKF.

Thus, it is possible to interpret EnKF as a purely statistical Monte Carlo method where the ensemble of model states evolves in state space with the mean as the best estimate and the spreading of the ensemble as the error variance. At measurement times each observation is represented by another ensemble, where the mean is the actual measurement and the variance of the ensemble represents the measurement errors. Thus, we combine a stochastic prediction step with a stochastic analysis step.

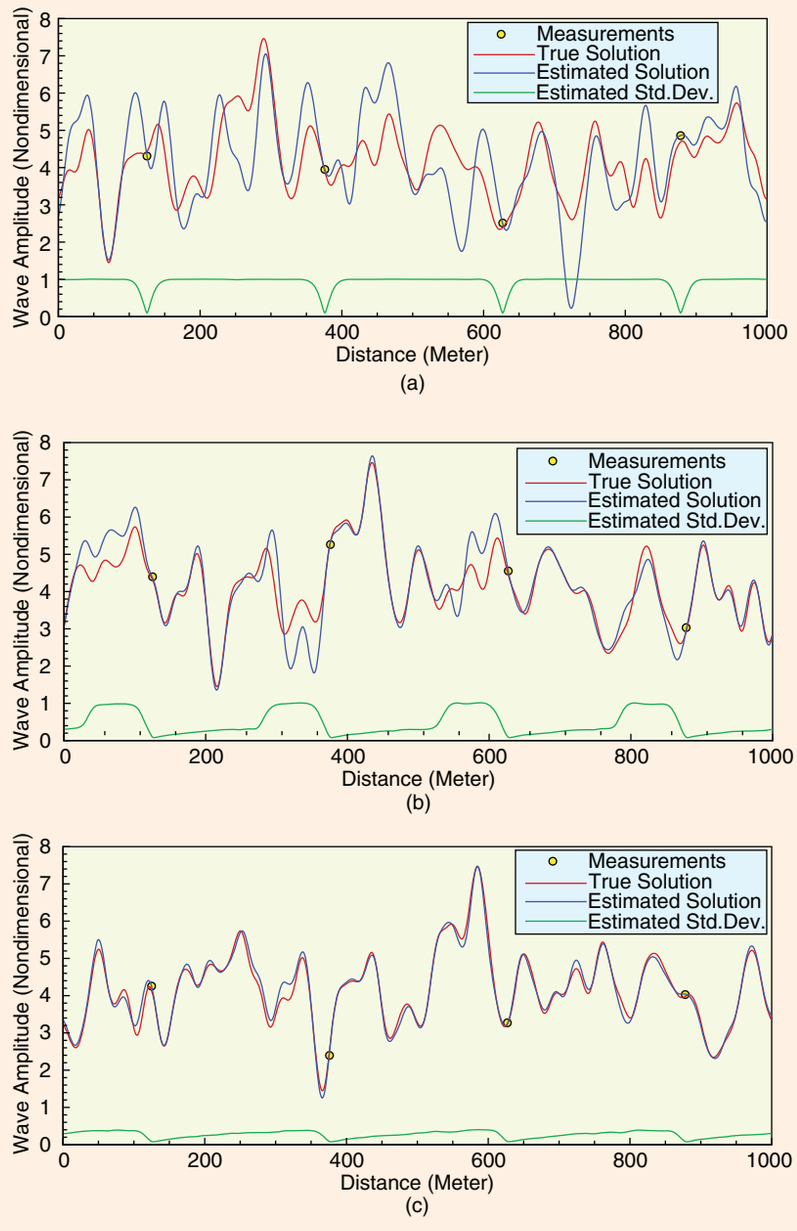


FIGURE 2 An ensemble Kalman filter experiment. For this experiment a linear advection equation illustrates how a limited ensemble size of 100 realizations facilitates estimation in a high-dimensional system whose state vector contains 1000 entries. The plots show the reference solution, measurements, estimate, and standard deviation at three different times, (a) $t = 5$ s, (b) $t = 150$ s, and (c) $t = 300$ s.

PROBABILISTIC FORMULATION

For the ensemble Kalman smoother (EnKS) [8], the estimate at a particular time is updated based on past, present, and future measurements. In contrast, a filter estimate is influenced only by the past and present measurements. Thus, EnKF becomes a special case of EnKS, where information from measurements is not projected backward in time. The assumptions of measurement errors being independent in time and the dynamical model being a Markov process are sufficient to derive the EnKF and the EnKS. These assumptions are normally not critical and are already used in the original KF. It is also possible to include the estimation of static model parameters in a consistent manner. The combined parameter and state estimation problem for a dynamical model can be formulated as finding the joint pdf of the parameters and model state, given a set of measurements and a dynamical model with known uncertainties.

Model Equations and Measurements

We consider a model with associated initial and boundary conditions on the spatial domain $\mathcal{D}$ with boundary $\partial\mathcal{D}$, and with observations

$$\frac{\partial \psi(\boldsymbol{x},t)}{\partial t} = \boldsymbol{G}(\psi(\boldsymbol{x},t), \alpha(\boldsymbol{x})) + \boldsymbol{q}(\boldsymbol{x},t), \tag{33}$$

$$\psi(\boldsymbol{x},t_0) = \Psi_0(\boldsymbol{x}) + \boldsymbol{a}(\boldsymbol{x}), \tag{34}$$

$$\psi(\xi,t) = \Psi_b(\xi,t) + \boldsymbol{b}(\xi,t), \text{ for all } \xi \in \delta\mathcal{D}, \tag{35}$$

$$\alpha(\boldsymbol{x}) = \alpha_0(\boldsymbol{x}) + \alpha'(\boldsymbol{x}), \tag{36}$$

$$\mathcal{M}[\psi,\alpha] = \boldsymbol{d} + \boldsymbol{\epsilon}. \tag{37}$$

The model state $\psi(\boldsymbol{x},t) \in \mathcal{R}^{n_\psi}$ is a vector consisting of the n_ψ model variables, where each variable is a function of

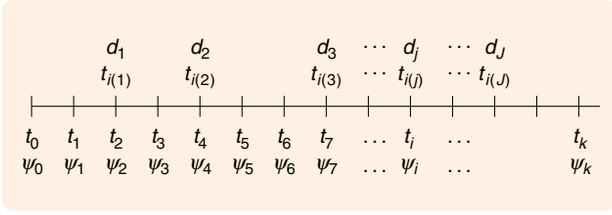


FIGURE 3 Discretization in time. The time interval is discretized into $k+1$ nodes, t_0 to t_k, where the model state vector $\psi_i = \psi(t_i)$ is defined. The measurement vectors d_j are available at the discrete subset of times $t_{i(j)}$, where $j = 1, \ldots, j$.

space and time. The nonlinear model is defined by (33), where $G(\psi, \alpha) \in \mathcal{R}^{n_\psi}$ is the nonlinear model operator. More general forms can be used for the nonlinear model operator, although (33) suffices to demonstrate the methods considered here.

The model state is assumed to evolve in time from the initial state $\Psi_0(\boldsymbol{x}) \in \mathcal{R}^{n_\psi}$ defined in (34), under the constraints of the boundary conditions $\Psi_b(\xi, t) \in \mathcal{R}^{n_\psi}$ defined in (35). The coordinate ξ runs over the surface $\partial\mathcal{D}$, where the boundary conditions are defined. The variable $\boldsymbol{b}$ is used to represent errors in the boundary conditions.

We define $\alpha(\boldsymbol{x}) \in \mathcal{R}^{n_\alpha}$ as the set of n_α poorly known parameters of the model. The parameters can be a vector of spatial fields in the form written here, or, alternatively, a vector of scalars, and are assumed to be constant in time. A prior estimate $\alpha_0(\boldsymbol{x}) \in \mathcal{R}^{n_\alpha}$ of the vector of parameters $\alpha(\boldsymbol{x}) \in \mathcal{R}^{n_\alpha}$ is introduced through (36), and possible errors in the prior are represented by $\alpha'(\boldsymbol{x})$.

Additional conditions are present in the form of the measurements $\boldsymbol{d} \in \mathcal{R}^M$. Both direct point measurements of the model solution and more complex parameters that are nonlinearly related to the model state can be used. For the time being we restrict ourselves to the case of linear measurements. An example of a direct measurement functional is then

$$\mathcal{M}_i[\psi] = \int\int \psi^{\mathrm{T}}(\boldsymbol{x}, t)\delta_{\psi_i}\delta(t - t_i)\delta(\boldsymbol{x} - \boldsymbol{x}_i)dt\, d\boldsymbol{x} = \psi(\boldsymbol{x}_i, t_i)\delta_{\psi_i}, \tag{38}$$

where the integration is over the space and time domain of the model. The measurement d_i is related to the model-state variable as selected by the vector $\delta_{\psi_i} \in \mathcal{R}^{n_\psi}$ and evaluated at the space and time location $(\boldsymbol{x}_i, t_i)$. If a model with three state variables is used and the second variable is measured, then δ_{ψ_i} becomes the vector $(0, 1, 0)^{\mathrm{T}}$, while $\delta(t - t_i)$ and $\delta(\boldsymbol{x} - \boldsymbol{x}_i)$ are Dirac delta functions.

In (33)–(37) we include unknown error terms, $\boldsymbol{q}$, $\boldsymbol{a}$, $\boldsymbol{b}$, α', and ϵ, which represent errors in the model equations, the initial and boundary conditions, the first guess for the model parameters, and the measurements, respectively. Without these error terms the system as given above is overdetermined and has no solution. On the other hand, when we introduce these error terms without additional conditions, the system has infinitely many solutions. The way to proceed is to introduce a statistical hypothesis about the errors, for example, assuming that the errors are normally distributed with means equal to zero and known error covariances.

Bayes Theorem

We now consider the model variables, the poorly known parameters, the initial and boundary conditions, and the measurements as random variables, which can be described by pdfs. The joint pdf for the model state, as a function of space, time, and the parameters, is $f(\psi, \alpha)$. Furthermore, for the measurements we can define the likelihood function $f(\boldsymbol{d} \mid \psi, \alpha)$. Thus, we may have measurements of both the model state and the parameters. Using Bayes theorem, the parameter and state estimation problem is now written in the simplified form

$$f(\psi, \alpha \mid \boldsymbol{d}) = \gamma f(\psi, \alpha) f(\boldsymbol{d} \mid \psi, \alpha), \tag{39}$$

where γ is a constant of proportionality whose computation requires the evaluation of the integral of (39) over the high-dimensional solution and parameter space.

Parameter estimation problems, in particular, for applications involving high-dimensional models, such as oceanic, atmospheric, marine ecosystem, hydrology, and petroleum applications, often do not include the model state as a variable to be estimated. It is more common to first solve for the poorly known parameters by minimizing an appropriate cost function where the model equations act as a strong constraint and then rerun the model to find the model solution. It is then implicitly assumed that the model does not contain errors, an assumption that generally is invalid.

In the dynamical model, we specify initial and boundary conditions as random variables, and we include prior information about the parameters. Thus, we define the pdfs $f(\psi_0)$, $f(\psi_b)$, and $f(\alpha)$ for the estimates ψ_0, ψ_b, and α of the initial and boundary conditions, and the parameters, respectively. Instead of $f(\psi,\alpha)$, we write

$$f(\psi, \alpha, \psi_0, \psi_b) = f(\psi \mid \alpha, \psi_0, \psi_b) f(\psi_0) f(\psi_b) f(\alpha). \tag{40}$$

Equation (39) should accordingly be written as

$$f(\psi, \alpha, \psi_0, \psi_b \mid \boldsymbol{d}) = \gamma f(\psi \mid \alpha, \psi_0, \psi_b) f(\psi_0) f(\psi_b) f(\alpha) f(\boldsymbol{d} \mid \psi, \alpha), \tag{41}$$

where it is also assumed that the boundary conditions and initial conditions are independent, although this assumption may not be true for the locations where initial and boundary conditions intersect at t_0. Here the pdf $f(\psi \mid \alpha, \psi_0, \psi_b)$ is the prior density for the model solution given the parameters and initial and boundary conditions.

Discrete Formulation

In the following discussion we work with a model state that is discretized in time, that is, $\psi(\boldsymbol{x}, t)$ is represented at fixed time intervals as $\psi_i(\boldsymbol{x}) = \psi(\boldsymbol{x}, t_i)$ with $i = 0, 1, \ldots, k$; see Figure 3. Furthermore, we define the pdf for the model integration from time t_{i-1} to t_i as $f(\psi_i \,|\, \psi_{i-1}, \alpha, \psi_b(t_i))$, which assumes that the model is a first-order Markov process. The joint pdf for the model solution and the parameters in (40) can now be written as

$$f(\psi_1, \ldots, \psi_k, \alpha, \psi_0, \psi_b) = f(\alpha) f(\psi_b) f(\psi_0) \prod_{i=1}^{k} f(\psi_i \,|\, \psi_{i-1}, \alpha, \psi_b). \tag{42}$$

Independent Measurements

We now assume that the measurements $\boldsymbol{d} \in \mathcal{R}^M$ can be divided into subsets of measurement vectors $\boldsymbol{d}_j \in \mathcal{R}^{m_j}$, collected at times $t_{i(j)}$, with $j = 1, \ldots, J$ and $0 < i(1) < i(2) < \cdots < i(J) < k$. The subset $\boldsymbol{d}_j$ depends only on $\psi(t_{i(j)}) = \psi_{i(j)}$ and α. Furthermore, it is assumed that the measurement errors are uncorrelated in time. We can then write

$$f(\boldsymbol{d} \,|\, \psi, \alpha) = \prod_{j=1}^{J} f(\boldsymbol{d}_j \,|\, \psi_{i(j)}, \alpha), \tag{43}$$

and from Bayes theorem we obtain

$$f(\psi_1, \ldots, \psi_k, \alpha, \psi_0, \psi_b \,|\, \boldsymbol{d}) = \gamma f(\alpha) f(\psi_0) f(\psi_b) \prod_{i=1}^{k} f(\psi_i \,|\, \psi_{i-1}, \alpha) \prod_{j=1}^{J} f(\boldsymbol{d}_j \,|\, \psi_{i(j)}, \alpha). \tag{44}$$

Sequential Processing of Measurements

It is shown in [36] and [37] that, in the case of time-correlated model errors, it is possible to reformulate the problem as a first-order Markov process by augmenting the model errors to the model-state vector. A simple equation forced by white noise can be used to simulate the time evolution of the model errors.

In [38] it is shown that a general smoother and filter can be derived from the Bayesian formulation given in (44). We now rewrite (44) as a sequence of iterations

$$f(\psi_1, \ldots, \psi_{i(j)}, \alpha, \psi_0, \psi_b \,|\, \boldsymbol{d}_1, \ldots, \boldsymbol{d}_j) = \gamma f(\psi_1, \ldots, \psi_{i(j-1)}, \alpha, \psi_0, \psi_b \,|\, \boldsymbol{d}_1, \ldots, \boldsymbol{d}_{j-1}) \times \prod_{i=i(j-1)+1}^{i(j)} f(\psi_i \,|\, \psi_{i-1}, \alpha) f(\boldsymbol{d}_j \,|\, \psi_{i(j)}, \alpha). \tag{45}$$

Thus, we formulate the combined parameter and state-estimation problem using Bayesian statistics and see that, under the condition that measurement errors are independent in time and the dynamical model is a Markov process, a recursive formulation can be used for Bayes theorem. That is, the model state and parameters with their respective uncertainties are updated sequentially in time whenever the measurements become available.

We note again that this recursion does not introduce any significant approximations and thus describes the full inverse problem as long as the model is a Markov process and the measurements errors are independent in time. Further, for many problems the recursive processing of measurements provides a better posed approach for solving the inverse problem than trying to process all of the measurements simultaneously as is normally done in variational formulations. Sequential processing is also convenient for forecasting problems where new measurements can be processed when they arrive without recomputing the full inversion.

Ensemble Smoother

The ensemble smoother (ES) can be derived by assuming that the pdfs for the model prediction as well as the likelihood are Gaussian and by using the original Bayes theorem (41). The derivation requires that we approximate the pdfs resulting from an integration of the ensemble through the whole assimilation time period with Gaussian pdfs. We can then replace Bayes theorem with a least squares cost function similar to (1), but with the time dimension included, and the analysis becomes a standard variance minimizing analysis in space and time. All of the data are processed in one step, and the solution is updated as a function of space and time, using the space-time covariances estimated from the ensemble of model realizations. The ES in [39] is computed as a first-guess estimate, which is the mean of the freely evolving ensemble, plus a linear combination of time-dependent influence functions, which are calculated from the ensemble statistics. Thus, the method is equivalent to a variance-minimizing objective analysis method where the time dimension is included.

Ensemble Kalman Smoother

An assumption of Gaussian pdfs for the model prediction (the prior) and the distribution for the data (the likelihood function) in (45) allows us to replace the Bayesian update formula with a least squares cost function similar to (1) but additionally including the state vector at all previous times. Again the cost function is minimized using a standard variance-minimizing analysis scheme, involving a state variable defined from the initial time to the current update time. That is, we also update the state variables backward in time using the combined time and space ensemble covariances. This scheme results in the EnKS as in [38].

Ensemble Kalman Filter

The EnKF is just a special case of EnKS where the updates at previous times are skipped. EnKF is obtained by

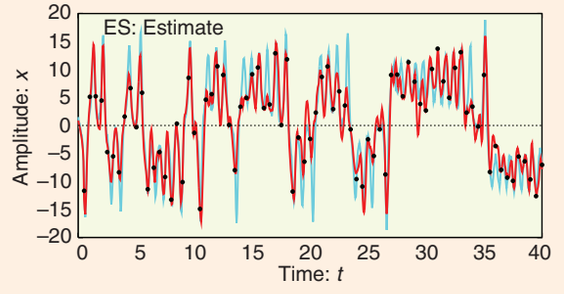


FIGURE 4 Ensemble smoother. (a) shows the inverse estimate (red line) and reference solution (blue line) for *x*. (b) shows the corresponding estimated standard deviations (red line) as well as the absolute value of the difference between the reference solution and the estimate, that is, the real posterior errors (blue line). (Reproduced from [38] with permission.)

integrating out the state variables at all previous times from (45) and assuming that the resulting model pdf for the current time as well as the likelihood function are Gaussian. The incremental update (45) can then be replaced by the penalty function (1), leading to the standard Kalman filter analysis equations. Thus, the measurements are filtered. At the final time, or, actually, from the latest update and for predictions into the future, EnKF and EnKS provide identical solutions.

EXAMPLE WITH THE LORENZ EQUATIONS

The example from [40] and [38] with the chaotic Lorenz model of [41] is now used to compare ES, EnKS, and EnKF. The Lorenz model consists of the coupled system of nonlinear ordinary differential equations given by

$$\frac{dx}{dt} = \gamma(y - x), \tag{46}$$

$$\frac{dy}{dt} = \rho x - y - xz, \tag{47}$$

$$\frac{dz}{dt} = xy - \beta z. \tag{48}$$

Here $x(t)$, $y(t)$, and $z(t)$ are the dependent variables, and we choose the parameter values $\gamma = 10$, $\rho = 28$, and $\beta = 8/3$. The initial conditions for the reference case are given by $(x_0, y_0, z_0) = (1.508870, -1.531271, 25.46091)$ and the time interval is $t \in [0, 40]$.

The observations and initial conditions are simulated by adding normally distributed white noise with zero mean and variance equal to 2.0 to the reference solution. All of the variables x, y, and z are measured. In the calculation of the ensemble statistics, an ensemble of 1000 members is used. The same simulation is rerun with various ensemble sizes, and the differences between the results are negligible with as few as 50 ensemble members.

The three methods discussed above are now examined and compared in an experiment where the time between measurements is $\Delta t_{\text{obs}} = 0.5$, which is similar to Experiment B in [40]. In the upper plots in figures 4–6, the red line denotes the estimate and the blue line is the reference solution. In the lower plots the red line is the standard deviation estimated from ensemble statistics, while the blue line is the true residuals with respect to the reference solution.

Ensemble Smoother Solution

The ES solution for the x-component and the associated estimated error variance are given in Figure 4. It is found that the ES performs rather poorly with the current data density. Note, however, that even if the fit to the reference trajectory is poor, the ES solution captures most of the transitions. The main problem is related to the estimate of the amplitudes in the reference solution. The problem is linked to the appearance of non-Gaussian contributions in the distribution for the model evolution, which can be expected in such a strongly nonlinear case.

Clearly, the error estimates evaluated from the posterior ensemble are not large enough at the peaks where the smoother performs poorly. The underestimated errors again result from neglecting the non-Gaussian contribution from the probability distribution for the model evolution. Otherwise, the error estimate looks reasonable with minima at the measurement locations and maxima between the measurements. Note again that if a linear model is used, then the posterior density becomes Gaussian and the ES provides, in the limit of an infinite ensemble size, the same solution as the EnKS and the Kalman smoother.

Ensemble Kalman Filter Solution

EnKF does a reasonably good job tracking the reference solution with the lower data density, as can be seen in Figure 5. One transition is missed near $t = 18$, while EnKF has problems, for example, at $t = 1, 5, 9, 10, 13, 17, 19, 23, 26$, and 34. The error variance estimate is consistent, showing large peaks at the locations where the estimate obviously has problems tracking the reference solution. Note also the similarity between the absolute value of the residual between the reference solution and the estimate, and the estimated standard deviation. For all peaks in the residual, a corresponding peak is present in the error variance estimate.

The error estimates show the same behavior as in [32] with very strong error growth when the model solution passes through the unstable regions of the state space and otherwise weak error variance growth or even decay in the stable regions. Note, for example, the low error variance for $t \in [28, 34]$ corresponding to the oscillation of the solution around one of the attractors.

In this case, the nonlinearity of the problem causes EnKF to perform better than the ES. In fact, at each update, the realizations are pulled toward the true solution and are not allowed to diverge toward the wrong attractors of the system. In addition, the Gaussian increments of the ensemble members lead to an approximately Gaussian ensemble distributed around the true solution. This property of the sequential updating is not exploited in the ES, where realizations evolve freely and lead to non-Gaussian ensemble distributions. Note again that if the model dynamics are linear, then, in the limit of an infinite ensemble size, EnKF gives the same solution as the Kalman filter and the ES solution gives a better result than EnKF.

Ensemble Kalman Smoother Solution

Figure 6 shows the solution obtained by EnKS. This solution is smoother in time than the EnKF solution and provides a better fit to the reference trajectory. All of the problematic locations in the EnKF solution are recovered in the smoother estimate. Note, for example, that the additional transitions at $t = 1$, 5, 13, and 34 in the EnKF solution are eliminated in the smoother. In addition, the missed transition at $t = 17$ is recovered by EnKS.

The error estimates are reduced throughout the time interval. In particular the large peaks in the EnKF solution are now significantly reduced. As for the EnKF solution, there are corresponding peaks in the error estimates for all the peaks in the residuals, which suggests that the EnKS error estimate is consistent with the true errors. In fact, in [40], it is found that the EnKS solution with $\Delta t_{\text{obs}} = 0.5$ seems to do as well or better than the EnKF solution with $\Delta t_{\text{obs}} = 0.25$.

Note that, if only z is measured in the Lorenz equations, the measured information is not sufficient to determine the solution. EnKF in this case develops realizations located at both attractors, and a bimodal distribution develops. The EnKF update breaks down with the bimodal distribution, but even the use of a Bayesian update in a particle filter does not suffice to determine the correct solution in this case since the bimodal distribution has the same probability for both peaks of the distribution. Note also that the assumption of Gaussian pdfs in the analysis equation is an approximation, whose severity must be judged on a case-by-case basis.

PRACTICAL IMPLEMENTATION

In [37] it is shown that the EnKF analysis scheme can be formulated in terms of the ensemble without reference to the ensemble covariance matrix, which allows for efficient numerical implementation and an alternative interpretation of the method. In the discussion below, we omit the time index, since all variables refer to the same update time.

Ensemble Representation of the Covariance

We define the matrix $\boldsymbol{A}$ whose columns are the ensemble members $\psi_i \in \mathcal{R}^n$ by

$$\boldsymbol{A} = (\psi_1, \psi_2, \ldots, \psi_N) \in \mathcal{R}^{n \times N}, \tag{49}$$

where N is the number of ensemble members and n is the size of the model state vector. The ensemble mean is stored in each column of $\overline{\boldsymbol{A}}$, which is defined as

$$\overline{\boldsymbol{A}} = \boldsymbol{A}\mathbf{1}_N, \tag{50}$$

where $\mathbf{1}_N \in \mathcal{R}^{N \times N}$ is the matrix whose entries are all equal $1/N$. We then define the ensemble perturbation matrix as

$$\boldsymbol{A}' = \boldsymbol{A} - \overline{\boldsymbol{A}} = \boldsymbol{A}(\boldsymbol{I} - \mathbf{1}_N). \tag{51}$$

The ensemble covariance matrix $\boldsymbol{C}^{\text{e}}_{\psi\psi} \in \mathcal{R}^{n \times n}$ can be defined as

$$\boldsymbol{C}^{\text{e}}_{\psi\psi} = \frac{1}{N-1}\boldsymbol{A}'(\boldsymbol{A}')^{\text{T}}. \tag{52}$$

Measurement Perturbations

Given a vector of measurements $\boldsymbol{d} \in \mathcal{R}^m$, where m is the number of measurements, we define the N vectors of perturbed observations as

$$\boldsymbol{d}_j = \boldsymbol{d} + \boldsymbol{\epsilon}_j, \; j = 1, \ldots, N, \tag{53}$$

which are stored in the columns of the matrix

$$\boldsymbol{D} = (\boldsymbol{d}_1, \boldsymbol{d}_2, \ldots, \boldsymbol{d}_N) \in \mathcal{R}^{m \times N}, \tag{54}$$

while the ensemble of perturbations, with ensemble mean equal to zero, are stored in the matrix

$$\boldsymbol{E} = (\boldsymbol{\epsilon}_1, \boldsymbol{\epsilon}_2, \ldots, \boldsymbol{\epsilon}_N) \in \mathcal{R}^{m \times N}, \tag{55}$$

from which we construct the ensemble representation of the measurement error covariance matrix

$$\boldsymbol{C}^{\text{e}}_{\epsilon\epsilon} = \frac{1}{N-1}\boldsymbol{E}\boldsymbol{E}^{\text{T}}. \tag{56}$$

Analysis Equation

The analysis equation (24), expressed in terms of the ensemble matrices, is

$$\boldsymbol{A}^{\text{a}} = \boldsymbol{A} + \boldsymbol{C}^{\text{e}}_{\psi\psi}\boldsymbol{M}^{\text{T}}(\boldsymbol{M}\boldsymbol{C}^{\text{e}}_{\psi\psi}\boldsymbol{M}^{\text{T}} + \boldsymbol{C}^{\text{e}}_{\epsilon\epsilon})^{-1}(d - \boldsymbol{M}\boldsymbol{A}). \tag{57}$$

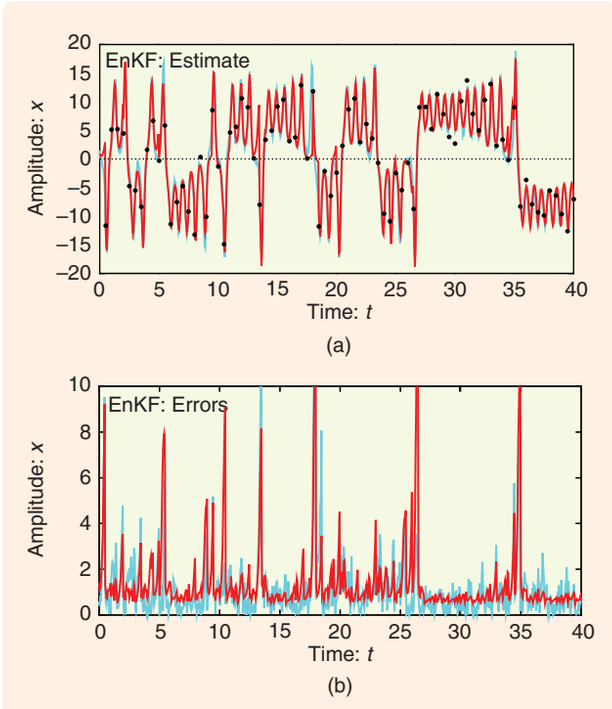


FIGURE 5 Ensemble Kalman filter. (a) shows the inverse estimate (red line) and reference solution (blue line) for x. (b) shows the corresponding estimated standard deviations (red line) as well as the absolute value of the difference between the reference solution and the estimate, that is, the real posterior errors (blue line). (Reproduced from [38] with permission.)

Using the ensemble of innovation vectors defined as

$$D' = D - MA, \tag{58}$$

along with the definitions of the ensemble error covariance matrices in (52) and (56), the analysis can be expressed as

$$A^{\mathrm{a}} = A + A'A'^{\mathrm{T}}M^{\mathrm{T}}(MA'A'^{\mathrm{T}}M^{\mathrm{T}} + EE^{\mathrm{T}})^{-1}D', \tag{59}$$

where all references to the error covariance matrices are eliminated.

We now introduce the matrix $S \in \mathcal{R}^{m\times N}$ holding the measurements of the ensemble perturbations by

$$S = MA', \tag{60}$$

and the matrix $C \in \mathcal{R}^{m\times m}$,

$$C = SS^{\mathrm{T}} + (N-1)C_{\epsilon\epsilon}. \tag{61}$$

Here we can use the full-rank, exact measurement error covariance matrix $C_{\epsilon\epsilon}$ as well as the low-rank representation $C^{\mathrm{e}}_{\epsilon\epsilon}$ defined in (56).

The analysis equation (59) can then be written as

$$\begin{aligned} A^{\mathrm{a}} &= A + A'S^{\mathrm{T}}C^{-1}D' \\ &= A + A(I - \mathbf{1}_N)S^{\mathrm{T}}C^{-1}D' \\ &= A(I + (I - \mathbf{1}_N)S^{\mathrm{T}}C^{-1}D') \\ &= A(I + S^{\mathrm{T}}C^{-1}D') \\ &= AX, \end{aligned} \tag{62}$$

where we use (51) and $\mathbf{1}_N S^{\mathrm{T}} \equiv \mathbf{0}$. The matrix $X \in \mathcal{R}^{N\times N}$ is defined as

$$X = I + S^{\mathrm{T}}C^{-1}D'. \tag{63}$$

Thus, the EnKF analysis becomes a combination of the forecast ensemble members and is searched for in the space spanned by the forecast ensemble.

It is clear that (62) is a stochastic scheme due to the use of randomly perturbed measurements. Thus, (62) allows for a nice interpretation of EnKF as a sequential Markov chain Monte Carlo algorithm, while making it easy to understand and implement the method. The efficient and stable numerical implementation of the analysis scheme is discussed in [8], including the case in which C is singular due to the number of measurements being larger than the number of realizations.

In practice, the ensemble size is critical since the computational cost scales linearly with the number of realizations. That is, each individual realization needs to be integrated forward in time. The cost associated with the ensemble integration motivates the use of an ensemble with the minimum number of realizations that can provide acceptable accuracy.

There are two major sources of sampling errors in EnKF, namely, the use of a finite ensemble of stochastic model realizations as well as the introduction of stochastic measurement perturbations [8], [42]. In addition, stochastic model errors influence the predicted error statistics, which is approximated by the ensemble. The sampling of physically acceptable model realizations and realizations of model errors is chosen to ensure that the ensemble matrix has full rank and good conditioning. Furthermore, stochastic perturbation of measurements used in EnKF can be avoided using a square root implementation of the analysis scheme, to be discussed below.

EnKF for Combined Parameter and State Estimation

When using EnKF to estimate poorly known model parameters, we start by representing the prior pdfs of the parameters by an ensemble of realizations, which is augmented to the state ensemble matrix A at the update steps. The poorly known parameters are then updated using the variance-minimizing analysis scheme, where the covariances between the predicted data and the parameters are used to update the parameters.

The updated ensemble for the parameters is included in the space defined by the initial ensemble of realizations. Thus, EnKF reduces the dimension of the combined parameter and state estimation problem to a size given by the dimension of the ensemble space. This simplification allows us to handle large sets of parameters, but it requires that the true parameters can be well represented in the ensemble space.

The parameter estimation approach used in EnKF and EnKS is a statistical minimization, or sampling of a posterior pdf, rather than a traditional minimization of a cost function. Thus, EnKF does not to the same extent suffer from the typical problems of converging to local minima as in parameter-estimation methods. EnKF rather has a problem with multimodal pdfs. However, the EnKF does not search for the mode but rather the mean of the distribution. Thus, in many cases where a minimization method might converge to a local minimum, EnKF provides an estimate that is the mean of the posterior. An important point is that the sequential updating used in EnKF reduces the risk of development of multimodal distributions, a result that is supported by the Lorenz example.

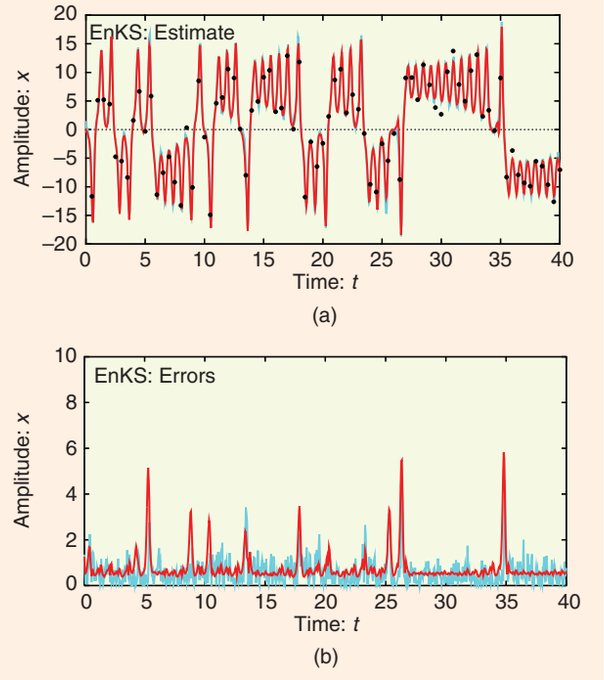


FIGURE 6 Ensemble Kalman smoother. (a) shows the inverse estimate (red line) and reference solution (blue line) for x. (b) shows the corresponding estimated standard deviations (red line) as well as the absolute value of the difference between the reference solution and the estimate, that is, the real posterior errors (blue line). (Reproduced from [38] with permission.)

DETERMINISTIC SQUARE ROOT SCHEME

The perturbation of measurements used in the EnKF standard analysis equation (57) is an additional source of sampling error. However, methods such as the square root scheme compute the analysis without perturbing the measurements [10]–[13], [43], [44].

Based on results from [10]–[13], a variant of the square root analysis scheme is derived in [42] and further elaborated on in [8]. The perturbation of measurements is avoided, and the scheme solves for the analysis without imposing any additional approximations, such as the assumption of uncorrelated measurement errors or knowledge of the inverse of the measurement error covariance matrix. This implementation requires the inverse of the matrix $\boldsymbol{C}$, defined in (61), which can be computed efficiently, either using the low-rank ensemble representation $\boldsymbol{C}_{\mathrm{e}}$ or by projecting the measurement error covariance matrix onto the space defined by the columns in $\boldsymbol{S}$ from (60). This version of the square root scheme is now presented.

Updating the Mean

In the square root scheme, the analyzed ensemble mean is computed from the standard Kalman filter analysis equation, which can be obtained by multiplying the first line in (62) from the right with $\mathbf{1}_N$, so that each column in the resulting equation for the mean becomes

$$\overline{\psi}^{\mathrm{a}} = \overline{\psi}^{\mathrm{f}} + \boldsymbol{A}'\boldsymbol{S}^{\mathrm{T}}\boldsymbol{C}^{-1}(\boldsymbol{d} - \boldsymbol{M}\overline{\psi}^{\mathrm{f}}). \tag{64}$$

Updating the Ensemble Perturbations

The deterministic algorithm used to update the ensemble perturbations is derived starting from the traditional analysis equation for the covariance update (28) in the Kalman filter. By using the ensemble covariances, (28) can be written as

$$\begin{aligned}&(\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{a}}\\ &\quad = (\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{f}} - (\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{f}}\boldsymbol{M}^{\mathrm{T}}(\boldsymbol{M}(\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{f}}\boldsymbol{M}^{\mathrm{T}} + \boldsymbol{R})^{-1}\boldsymbol{M}(\boldsymbol{C}^{\mathrm{e}}_{\psi\psi})^{\mathrm{f}},\end{aligned} \tag{65}$$

with the time index dropped for convenience. When using the ensemble representation for the error covariance matrix $\boldsymbol{C}^{\mathrm{e}}_{\epsilon\epsilon}$ defined in (52), (65) becomes

$$\boldsymbol{A}^{\mathrm{a}'}\boldsymbol{A}^{\mathrm{a}'\mathrm{T}} = \boldsymbol{A}'(\boldsymbol{I} - \boldsymbol{S}^{\mathrm{T}}\boldsymbol{C}^{-1}\boldsymbol{S})\boldsymbol{A}'^{\mathrm{T}}, \tag{66}$$

where $\boldsymbol{S}$ and $\boldsymbol{C}$ are defined in (60) and (61), and we drop the superscripts "f" on the forecast ensemble. We now derive an equation for updating the ensemble perturbations $\boldsymbol{A}'$ by defining a factorization of (66), which does not involve the measurements or measurement perturbations.

We start by forming $\boldsymbol{C}$ as defined in (61). For now we assume that $\boldsymbol{C}^{-1}$ exists, which requires that the rank of the ensemble be greater than the number of measurements. The low-rank case involves pseudo inversion [8]. Note also that the use of a full rank $\boldsymbol{C}^{\mathrm{e}}_{\epsilon\epsilon}$ can result in a full rank $\boldsymbol{C}$ even when $m \geq N$.

By computing the eigenvalue decomposition $\boldsymbol{Z}\Lambda\boldsymbol{Z}^{\mathrm{T}} = \boldsymbol{C}$, we obtain the inverse of $\boldsymbol{C}$ as

$$\boldsymbol{C}^{-1} = \boldsymbol{Z}\Lambda^{-1}\boldsymbol{Z}^{\mathrm{T}}, \tag{67}$$

where $\boldsymbol{Z} \in \mathcal{R}^{m\times m}$ is an orthogonal matrix and $\Lambda \in \mathcal{R}^{m\times m}$ is diagonal. The eigenvalue decomposition may be the most demanding computation required for the analysis when m is large. An efficient alternative inversion algorithm is presented in [8].

We now write (66) as

$$\begin{aligned}\boldsymbol{A}^{\mathrm{a}'}\boldsymbol{A}^{\mathrm{a}'\mathrm{T}} &= \boldsymbol{A}'(\boldsymbol{I} - \boldsymbol{S}^{\mathrm{T}}\boldsymbol{Z}\Lambda^{-1}\boldsymbol{Z}^{\mathrm{T}}\boldsymbol{S})\boldsymbol{A}'^{\mathrm{T}}\\ &= \boldsymbol{A}'(\boldsymbol{I} - (\Lambda^{-1/2}\boldsymbol{Z}^{\mathrm{T}}\boldsymbol{S})^{\mathrm{T}}(\Lambda^{-1/2}\boldsymbol{Z}^{\mathrm{T}}\boldsymbol{S}))\boldsymbol{A}'^{\mathrm{T}}\\ &= \boldsymbol{A}'(\boldsymbol{I} - \boldsymbol{X}_2^{\mathrm{T}}\boldsymbol{X}_2)\boldsymbol{A}'^{\mathrm{T}},\end{aligned} \tag{68}$$

where $\boldsymbol{X}_2 \in \mathcal{R}^{m\times N}$ is defined as

$$\boldsymbol{X}_2 = \Lambda^{-1/2}\boldsymbol{Z}^{\mathrm{T}}\boldsymbol{S}, \tag{69}$$

and where rank $(\boldsymbol{X}_2) = \min(m, N–1)$. Thus, $\boldsymbol{X}_2$ is a projection of $\boldsymbol{S}$ onto the eigenvectors of $\boldsymbol{C}$ scaled by the square root of the eigenvalues of $\boldsymbol{C}$.

Next we compute the singular value decomposition of $\boldsymbol{X}_2$ given by

$$\boldsymbol{U}_2\Sigma_2\boldsymbol{V}_2^{\mathrm{T}} = \boldsymbol{X}_2, \tag{70}$$

with $\boldsymbol{U}_2 \in \mathcal{R}^{m\times m}$, $\Sigma_2 \in \mathcal{R}^{m\times N}$ and $\boldsymbol{V}_2 \in \mathcal{R}^{N\times N}$. Since $\boldsymbol{U}_2$ and $\boldsymbol{V}_2$ are orthogonal matrices, (68) can be written

$$\begin{aligned}\boldsymbol{A}^{\mathrm{a}'}\boldsymbol{A}^{\mathrm{a}'\mathrm{T}} &= \boldsymbol{A}'(\boldsymbol{I} - [\boldsymbol{U}_2\Sigma_2\boldsymbol{V}_2^{\mathrm{T}}]^{\mathrm{T}}[\boldsymbol{U}_2\Sigma_2\boldsymbol{V}_2^{\mathrm{T}}])\boldsymbol{A}'^{\mathrm{T}}\\ &= \boldsymbol{A}'(\boldsymbol{I} - \boldsymbol{V}_2\Sigma_2^{\mathrm{T}}\Sigma_2\boldsymbol{V}_2^{\mathrm{T}})\boldsymbol{A}'^{\mathrm{T}}\\ &= \boldsymbol{A}'\boldsymbol{V}_2(\boldsymbol{I} - \Sigma_2^{\mathrm{T}}\Sigma_2)\boldsymbol{V}_2^{\mathrm{T}}\boldsymbol{A}'^{\mathrm{T}}\\ &= (\boldsymbol{A}'\boldsymbol{V}_2\sqrt{\boldsymbol{I} - \Sigma_2^{\mathrm{T}}\Sigma_2})(\boldsymbol{A}'\boldsymbol{V}_2\sqrt{\boldsymbol{I} - \Sigma_2^{\mathrm{T}}\Sigma_2})^{\mathrm{T}}.\end{aligned} \tag{71}$$

Thus, a solution for the analysis ensemble perturbations is

$$\boldsymbol{A}^{\mathrm{a}'} = \boldsymbol{A}'\boldsymbol{V}_2\sqrt{\boldsymbol{I} - \Sigma_2^{\mathrm{T}}\Sigma_2}. \tag{72}$$

As noted in [45] the update equation (72) does not conserve the mean of the ensemble perturbations and in fact leads to the production of outliers that contain most of the ensemble variance as explained in [46] and [8], which is further illustrated in the example below.

We now write the square root update in the more general form

$$\boldsymbol{A}^{\mathrm{a}'} = \boldsymbol{A}'\boldsymbol{T}, \tag{73}$$

where $\boldsymbol{T}$ is a square root transformation matrix.

It is shown in [44] and [47] that for the square root analysis scheme to be unbiased and preserve the zero mean in the updated perturbations, the vector $(1/N)\mathbf{1}$, where $\mathbf{1} \in \mathcal{R}^N$ has all components equal to one, must be an eigenvector of the square root transformation matrix $\boldsymbol{T}$. As noted in [44] and [47], this condition is not satisfied for the update in (72).

Multiplying (73) from the right with the vector $\mathbf{1}$ and assuming that $(1/N)\mathbf{1}$ is an eigenvector of $\boldsymbol{T}$, we can write

$$\boldsymbol{0} = \boldsymbol{A}^{\mathrm{a}'}\boldsymbol{1} = \boldsymbol{A}'\boldsymbol{T}\boldsymbol{1} = \lambda\boldsymbol{A}'\boldsymbol{1} = \boldsymbol{0}. \tag{74}$$

Equation (74) shows that a sufficient condition for the mean to be unbiased is that $(1/N)\mathbf{1}$ be an eigenvector of $\boldsymbol{T}$. If the transform matrix is of full rank, then this condition is also necessary [47].

The symmetric square root solution for the analysis ensemble perturbations is defined as

$$\boldsymbol{A}^{\mathrm{a}'} = \boldsymbol{A}'\boldsymbol{V}_2\,(\boldsymbol{I} - \Sigma_2^{\mathrm{T}}\Sigma_2)^{1/2}\boldsymbol{V}_2^{\mathrm{T}}. \tag{75}$$

It is easy to show that (75) is also a factorization of (71) since $\boldsymbol{V}_2$ is an orthogonal matrix. As shown in [44], [47], the symmetric square root has an eigenvector equal to $(1/N)\mathbf{1}$ and is unbiased. In addition, the symmetric square root resolves the issue with outliers in the factorization used in (72). The analysis update of the perturbations becomes a symmetric contraction of the forecast ensemble perturbations. Thus, if the predicted ensemble members have a non-Gaussian distribution, then the updated distribution retains the shape but the variance is reduced.

A randomization of the analysis update can be used to generate updated perturbations that better resemble a Gaussian distribution [42]. Thus, we write the symmetric square root solution (75) as

$$\boldsymbol{A}^{\mathrm{a}'} = \boldsymbol{A}'\boldsymbol{V}_2\,(\boldsymbol{I} - \Sigma_2^{\mathrm{T}}\Sigma_2)^{1/2}\boldsymbol{V}_2^{\mathrm{T}}\Phi^{\mathrm{T}}, \tag{76}$$

where Φ is a mean-preserving random orthogonal matrix, which can be computed using the algorithm from [44].

The properties of the square root schemes are illustrated in Figure 7, which shows the resulting ensemble updates using several variants of the EnKF analysis scheme. The Lorenz equations (46)–(48) are used since the strong nonlinearities lead to the development of a non-Gaussian distribution for the forecast ensemble. Three observations are used in the update step. Each ensemble member is plotted as a circle in the x, y plane. In Figure 7(a) and (b) the forecast ensemble members are plotted as the blue circles, which have a non-Gaussian distribution in the x, y plane.

In Figure 7(a) the updated analysis from the "one-sided" square root scheme in (72) is shown as the yellow circles. It can be seen that $N - 3$ of the updated ensemble perturbations collapse onto $(0, 0)$, while the three nonzero "outliers," one for each measurement, determine the ensemble variance. However, one of the outliers is

too close to zero to be distinguished from the other points at zero. The variance of the updated ensemble is correct, but the analysis introduces a bias through a shift in the ensemble mean. The shift in the mean should come as no surprise since we do not impose a condition for the conservation of the mean when the update equation is derived. The particular ensemble collapse related to the use of (72) is discussed and explained in [8]. It is in fact shown that with three measurements and a diagonal measurement error covariance matrix, we obtain an ensemble with three outliers, while the remainder of the perturbations collapse onto zero.

In Figure 7(a) the updated analysis from the symmetric square root scheme in (75) is shown as the red circles. This scheme has the property that it rescales the ensemble of perturbations without changing the original shape of the perturbations. Thus, the scheme allows for preserving possible non-Gaussian structures in the ensemble during the update. We also note that the symmetric square root scheme from (75) is unbiased and thus preserves the mean [44].

In Figure 7(b) the updated analysis from the symmetric square root scheme from (76), which includes an additional mean-preserving random rotation, is plotted using the green circles. It is clear that the ensemble of updated perturbations now has a Gaussian shape, and the non-Gaussian shape of the forecast ensemble perturbations is lost. The random rotation completely destroys any prior structure in the ensemble by randomly redistributing the variability among all of the ensemble members. Thus, the random rotation acts as a complete resampling from a Gaussian distribution, while preserving the ensemble mean and variance.

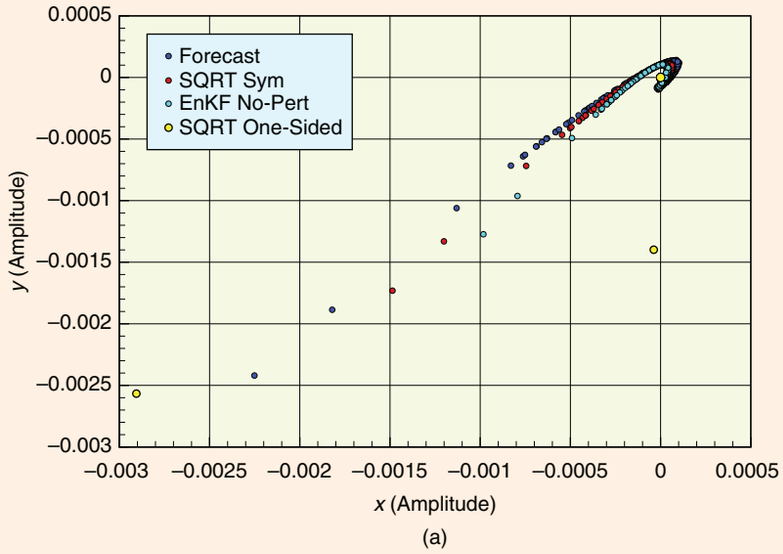


FIGURE 7 Forecast and analysis ensembles for the Lorenz equations illustrating properties of the analysis schemes discussed in the text. The data used in these plots were contributed by Dr. Pavel Sakov.

Figure 7(b) also shows the updated analysis from the standard EnKF scheme from (62), where the measurements are randomly perturbed to represent their uncertainty. The standard EnKF analysis becomes similar to the symmetric square root analysis with random rotation. As with the symmetric square root analysis, most of the non-Gaussian shape of the forecast ensemble is lost. However, only the increment in the standard EnKF analysis is Gaussian, and some of the non-Gaussian properties of the forecast ensemble is retained, as indicated by the two outliers that represent the tail of the distribution seen in the forecast ensemble.

It is also interesting to consider the standard EnKF scheme when used without perturbation of measurements. It is then clear from (28) that the variance is reduced twice by the additional multiplication with $I - K_e M$ resulting from $C_{\epsilon\epsilon}^{e}$ in (28) being identical to zero when the measurements are not treated as stochastic variables. Figure 7(a) shows that the EnKF scheme without perturbation of measurements preserves the shape of the forecast distribution in the same

To a large extent, EnKF overcomes two problems associated with the traditional KF.

way as the symmetric square root scheme, although the variance is too low. Thus, the perturbation of measurements in EnKF both increases the ensemble variance to the "correct" value, and introduces additional randomization. The randomization is different from the one observed in (76) since only the increments are randomized in the EnKF scheme with perturbation of measurements.

It is currently not clear which of the analysis schemes, that is, the standard EnKF (62), the symmetric square root (75), or the symmetric square root with random rotation (76), is best in practice. Probably the choice of analysis scheme depends on the dynamical model and possibly also on the measurement density and ensemble size used. For a linear dynamical model, the forecast distribution is Gaussian, and the random rotation is not needed. Thus, we then expect the symmetric square root (75) to be the best choice. On the other hand, for a strongly nonlinear dynamical model where non-Gaussian effects are dominant in the predicted ensemble, the symmetric square root with a random rotation (76) or EnKF with perturbed measurements (62) may work better. Both of these schemes introduce Gaussianity into the analysis update, while a Gaussian forecast ensemble may lead to more consistent analysis updates.

The random rotation might be considered as a resampling from a Gaussian distribution at each analysis update. Note again that the random rotation in the square root filter, contrary to the measurement perturbation used in EnKF, completely eliminates all previous non-Gaussian structures that may be contained in the forecast ensemble.

SPURIOUS CORRELATIONS, LOCALIZATION, AND INFLATION

Since EnKF is a Monte Carlo method, making this method affordable for large systems requires the use of a sufficiently small ensemble of model realizations. Around 100 realizations in the ensemble is typical in applications, and in many cases we see only marginal improvements when the ensemble size is further increased, which is explained by the slow convergence, proportional to $\sqrt{N}$, of Monte Carlo methods, together with the fact that a large part of the variability in the state and parameters often is well represented by an ensemble of 100 model realizations. On the other hand, even $\mathcal{O}(100)$ model realizations become extremely computationally demanding in many applications, which is an incentive for using as few realizations as possible. In the following we discuss the problems caused by using a finite ensemble size and present some remedies that can reduce the impact of sampling errors.

Spurious Correlations

The use of a finite ensemble size to approximate the error covariance matrix introduces sampling errors that are seen as spurious correlations over long spatial distances or between variables known to be uncorrelated. A result of these sampling errors is that the updated ensemble variance is underestimated. On the other hand, the consistency of the updated variance improves when a larger ensemble is used. A spurious correlation between a predicted measurement and a variable leads to a small nonphysical update of the variable in each ensemble member, and thus an associated variance reduction. This problem is present in all EnKF applications and can lead to filter divergence.

The following example, which is based on the linear advection case from Figure 2, illustrates the variance reduction resulting from spurious correlations. We use the form (62) for the EnKF analysis scheme with the update matrix $\boldsymbol{X}$ defined from (63).

An additional ensemble $\boldsymbol{B} \in \mathcal{R}^{n_{\text{rand}} \times N}$ is generated, where each row contains random samples from a Gaussian distribution with mean equal to zero and variance equal to one, and the entries in different rows are sampled independently. Thus, $\boldsymbol{B}$ is the ensemble matrix for a state vector of independent variables with zero mean and unit variance. At analysis times we compute the updates

$$\begin{pmatrix} \boldsymbol{A}^{\text{a}} \\ \boldsymbol{B}^{\text{a}} \end{pmatrix} = \begin{pmatrix} \boldsymbol{A}^{\text{f}} \\ \boldsymbol{B}^{\text{f}} \end{pmatrix} \boldsymbol{X}. \tag{77}$$

The predicted ensemble $\boldsymbol{A}^{\text{f}}$ is the result of the ensemble integration using the advection model, while $\boldsymbol{B}^{\text{f}}$ does not evolve according to any dynamical equation and at an update time equals $\boldsymbol{B}^{\text{a}}$ at the previous update time.

Since the correlations between $\boldsymbol{B}$ and the predicted measurement perturbations $\boldsymbol{S}$ become zero in the limit of an infinite ensemble size, it follows that

$$\lim_{N \to \infty} \frac{\boldsymbol{B}\boldsymbol{S}^{\text{T}}}{N-1} = \boldsymbol{0}. \tag{78}$$

However, due to the finite ensemble size, (78) cannot be exactly satisfied, and $\boldsymbol{B}^{\text{a}}$ experiences a small update and associated reduction of variance through the update in (77).

As in the advection example, we compute the matrix $\boldsymbol{X}$ based on the four measurements, and then apply it to $\boldsymbol{B}$ according to (77) at every analysis time. The value $n_{\text{rand}} = 100$ is found to be sufficient to obtain a consistent result that is independent of the random sampling of $\boldsymbol{B}$.

The variance reduction resulting from the spurious correlations is illustrated in Figure 8, which shows the decrease of the average variance of the random ensemble $\boldsymbol{B}$, resulting from EnKF with 100 and 250 realizations, and from the symmetric square root scheme using 100 realizations.

EnKF with 100 realizations is repeated five more times using different random seeds to verify that the result is independent of the seed. A nearly linear decrease of variance is obtained during the first 50 updates, while for the final 12 updates the decrease is lower. The reason for the lower error variance reduction in the final part of the experiment is that the information assimilated at one measurement location propagates to the next measurement location during 50 updates. Thus, after 50 updates the ensemble variance is lower at the measurement locations, and the relative weight on the data compared to the prediction is decreased. EnKF with 250 realizations experiences a significantly lower impact from spurious correlations, as expected.

The square root scheme is slightly less influenced by the spurious correlations, and an explanation can be that the measurement perturbations in the EnKF update increases the strength of the update of individual realizations and thus amplifies the impact of the spurious correlations.

In many dynamical systems, the variance decrease caused by spurious correlations may be masked by strong dynamical instabilities. The impact of the spurious correlations may then be less significant. On the other hand, in parameter-estimation problems, the spurious correlations clearly lead to an underestimate of the ensemble variance of the parameters.

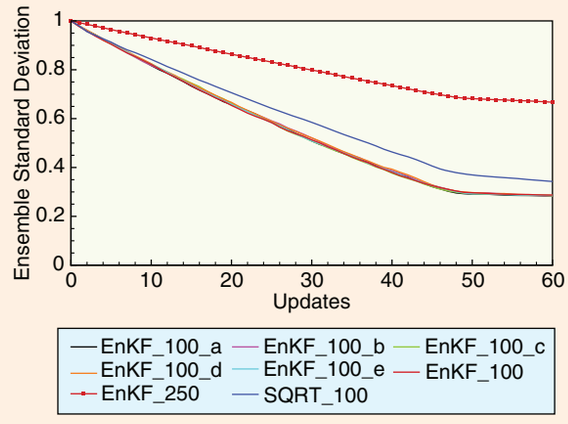


FIGURE 8 Variance reduction of a random ensemble due to spurious correlations, as a function of analysis updates. The ensemble Kalman filter (EnKF) with 100 realizations is compared with EnKF with 250 realizations as well as the square root scheme using 100 realizations. EnKF with 100 realizations is repeated using different seeds to ensure that the results are consistent.

Localization

We now discuss the use of localization to reduce spurious correlations [48]. Two classes of localization methods are currently used, covariance localization and local updating.

In [48] the ensemble covariance matrix is multiplied with a specified correlation matrix through a Schur product (entry-wise multiplication). The specified correlation functions are defined with local support and thus effectively truncate the long-range spurious correlations produced by the limited ensemble size. Covariance localization is used in [11], [12], [49], and [50].

We can assume that only measurements located within a certain distance from a gridpoint impact the analysis in that gridpoint. This assumption allows for an algorithm where the analysis is computed gridpoint by gridpoint, and only a subset of observations, located near the current gridpoint, is used in each local analysis. This approach is used in [51], [52], and [37] and is also the approach used in the local EnKF in [53]. In addition to reducing the impact of long-range spurious correlations, the localization methods make it simpler to handle large data sets where the number of measurements is much greater than the number of ensemble realizations.

Another reason for computing the local analysis is the fact that EnKF is computed in a space spanned by the ensemble members. This subspace may be rather small compared to the total dimension of the model state. Computing the analysis gridpoint by gridpoint implies that, for each gridpoint, a small model state is solved for in a relatively large ensemble space. The analysis then results from a different combination of ensemble members for each gridpoint, and the analysis scheme is allowed to reach solutions not originally represented by the ensemble. In many applications the local analysis scheme significantly reduces the impact of a limited ensemble size and allows for the use of EnKF with high-dimensional model systems.

The degree of approximation introduced by the local analysis depends on the range of influence defined for the observations. In the limit that this range becomes sufficiently large to include all of the data, the solution for all the gridpoints becomes identical to the standard global analysis. The range parameter must be tuned and should be large enough to include the information from measurements that contribute significantly but small enough to eliminate the spurious impact of remote measurements.

The local analysis algorithm goes as follows. We first construct the input matrices to the global EnKF, that is, the measured ensemble perturbations $\boldsymbol{S}$, the innovations $\boldsymbol{D}'$, and either the measurement perturbations $\boldsymbol{E}$ or the measurement error covariance matrix $\boldsymbol{C}_{\epsilon\epsilon}$. We then loop through the model grid, and, for each gridpoint, for example, (i, j) for a two-dimensional model, we extract the rows from these matrices corresponding to measurements

that are used in the current update, and then compute the matrix $\boldsymbol{X}_{(i,j)}$ that defines the update for gridpoint (i, j).

The analysis at gridpoint (i, j) becomes

$$\boldsymbol{A}^{\mathrm{a}}_{(i,j)} = \boldsymbol{A}_{(i,j)}\boldsymbol{X}_{(i,j)} \tag{79}$$

$$= \boldsymbol{A}_{(i,j)}\boldsymbol{X} + \boldsymbol{A}_{(i,j)}(\boldsymbol{X}_{(i,j)} - \boldsymbol{X}), \tag{80}$$

where $\boldsymbol{X}$ is the global solution, while $\boldsymbol{X}_{(i,j)}$ becomes the solution for a local analysis corresponding to gridpoint (i, j) where only the nearest measurements are used in the analysis. Thus, it is possible to compute the global analysis first and then add the corrections from the local analysis if these effects are significant.

The quality of the EnKF analysis is connected to the ensemble size used. We expect that to achieve the same quality of the result, a larger ensemble is needed for the global analysis than the local analysis. In the global analysis, a large ensemble is needed to properly explore the state space and to provide a consistent result that is as good as the local analysis. Note also that the use of a local analysis scheme is likely to introduce nondynamical modes, although the amplitudes of these modes are small if a large enough influence radius is used when selecting measurements. We also refer to the discussions on localization and filtering of long-range correlations by [54].

In adaptive localization methods, the assimilation system itself is used to determine the localization strategy. Such algorithms are useful since the dynamical covariance functions change in space and time, and the spurious correlations depend on the ensemble size. Thus, every assimilation problem and ensemble size requires a separate tuning of the localization parameters.

The hierarchical approach in [55] uses several small ensembles to explore the need for using localization in the analysis. This approach uses a Monte Carlo method based on splitting the ensemble into several small ensembles to assess the sampling errors and the spurious correlations. This method is a statistically consistent approach to the problem. However, the localization is optimized for a small ensemble and may become suboptimal when used with the full ensemble including all realizations.

An alternative localization method in [56] is based on the online computation of a flow-dependent moderation function that is used to damp long-range and spurious correlations. This method is named SENCORP for "smoothed ensemble correlations raised to a power." The idea is that the moderation functions can be generated from a smoothed covariance function, which, when raised to a power, damps small correlations.

In [57] a local analysis method handles measurements that are integral parameters of the model state. The idea is that the covariance matrix of the predicted measurements is computed globally using the full model state, while the updates are computed locally gridpoint by gridpoint, and only the measurements that have significant correlations with the model variables in the local gridpoint are assimilated.

Thus, while traditional localization methods are distance based, [55]–[57] discuss adaptive localization methods where the assimilation system determines whether correlations are significant or spurious, and whether a particular measurement shall be used in the update of a particular model variable. The further development of adaptive localization methods is important for many applications where distance-based methods are less suitable, an example being the use of measurements that are integral functions of the model state as in [57].

Finally, it is not clear how the local analysis scheme is best implemented in EnKS. One approach is to define the local analysis to use only measurements in a certain space-time domain, taking into account the propagation of information in the model together with the time scales of the model. In [58] EnKS is used with a high-dimensional atmospheric circulation model. The impact of spurious correlations related to the lag time in a lagged EnKS is studied, and it is pointed out that the lagged implementation facilitates localization in time.

Inflation

A covariance inflation procedure [59] can be used to counteract the variance reduction observed due to the impact of spurious correlations as well as other effects leading to underestimation of the ensemble variance. The impact of ensemble size on noise in distant covariances is examined in [49], while the impact of using an "inflation factor" as discussed in [59] is evaluated. The inflation factor is used to replace the forecast ensemble according to

$$\psi_j = \rho(\psi_j - \overline{\psi}) + \overline{\psi}, \tag{81}$$

with ρ slightly greater than one (typically 1.01). The inflation procedure is also used in [60], where the EnKF is examined in an application with the Lorenz attractor, and results are compared with those obtained from different versions of the singular evolutive extended Kalman (SEEK) filter and a particle filter. In [60], ensembles with very few members are used, which favors methods like the SEEK where the "ensemble" of empirical orthogonal functions (EOFs) is selected to best represent the model attractor.

Several approaches adaptively estimate an optimal inflation parameter. In [61] the covariance inflation is estimated based on the sequence of innovation statistics, while in [62] a method is presented that is based on augmenting the inflation parameter to the model state where it is updated as a parameter in the EnKF analysis computations. Online estimation of the inflation parameter is also studied in [63] together with the simultaneous estimation of observation errors. It is found that the estimation of inflation alone does not work appropriately without accurate observation error statistics, and vice versa.

Clearly, the inflation parameter becomes a tuning parameter, and optimally it is best estimated adaptively. The need for inflation depends on the use of a local versus global analysis scheme, and the use of a local scheme can to a large extent reduce the need for an additional inflation.

Here we describe an alternative approach for estimating the inflation coefficient. In the spurious correlation example, as presented in Figure 8, an independent ensemble is used to quantify the variance reduction due to spurious correlations. A simple algorithm for correcting the analyzed ensemble perturbations in each analysis step goes as follows.

At each analysis time we generate the additional ensemble matrix $\boldsymbol{B}^{\mathrm{f}}$ with random normally distributed numbers, such that the mean in each row is exactly zero, and the variance is exactly equal to one. We thus sample the matrix randomly from $\mathcal{N}(0, 1)$. Then, for each row, first subtract any nonzero mean, then compute the standard deviation and scale all entries by it. Then, compute the analysis update according to (77). For each row in $\boldsymbol{B}^{\mathrm{a}}$, compute the standard deviation. The inflation factor ρ is then defined as one over the average of the standard deviations from each row in $\boldsymbol{B}^{\mathrm{a}}$. The accuracy of the estimated inflation factor depends on the number of realizations used as well as the number of rows in $\boldsymbol{B}$. It is expected that with a low number of realizations additional rows in $\boldsymbol{B}$ might compensate for the sampling errors when computing the inflation factor.

This algorithm provides a good first approximation of the inflation factor needed to counteract variance reduction due to long-range spurious correlations resulting from sample noise. The estimated inflation factor depends on the number of realizations used, the number of measurements, and the strength of the update determined by the innovation vector and both the predicted and measurement error covariance matrices. A question remains, as to whether the inflation is best applied equally for the whole model state, including at the measurement locations.

CONCLUSIONS

This article provides a fundamental theoretical basis for understanding EnKF and serves as a useful text for future users. Data assimilation and parameter-estimation problems are explained, and the concept of joint parameter and state estimation, which can be solved using ensemble methods, is presented. KF and EKF are briefly discussed before introducing and deriving EnKF. Similarities and differences between KF and EnKF are pointed out. The benefits of using EnKF with high-dimensional and highly nonlinear dynamical models are illustrated by examples. EnKF and EnKS are also derived from Bayes theorem, using a probabilistic approach. The derivation is based on the assumption that measurement errors are independent in time and the model represents a Markov process, which allows for Bayes theorem to be written in a recursive form, where measurements are processed sequentially in time. The practical implementation of the analysis scheme is discussed, and it is shown that it can be computed efficiently in the space spanned by the ensemble realizations. The square root scheme is discussed as an alternative method that avoids the perturbation of measurements. However, the square root scheme has other pitfalls, and it is recommended to use the symmetric square root with or without a random rotation. The random rotation introduces a stochastic component to the update, and the quality of the scheme may then not improve compared to the original stochastic EnKF scheme with perturbed measurements.

ACKNOWLEDGMENTS

I am grateful for many fruitful discussions with Pavel Sakov and Laurent Bertino during the preparation of this article, and I thank Pavel Sakov for providing the data used to illustrate the properties of the different analysis schemes in Figure 7 and for pointing out that the use of an ensemble representation of the measurement error covariance matrix leads to an exact cancellation in the second last line in (27). The extensive feedback provided by the reviewers contributed to significantly improving the quality of the manuscript. I am partly supported by the Norwegian Research Council through the Evita EnKF Project.

AUTHOR INFORMATION

Geir Evensen (geve@statoilhydro.com) received the Ph.D. in applied mathematics from the University of Bergen, Norway, in 1992. He has published more than 40 articles related to data assimilation, including *Data Assimilation: The Ensemble Kalman Filter*. He currently works on data assimilation and parameter estimation as a research director at StatoilHydro in Bergen and holds an associate position at the Nansen Center in Bergen.

REFERENCES

[1] G. Evensen, "Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics," *J. Geophys. Res.*, vol. 99, no. C5, pp. 10143–10162, 1994.

[2] R. E. Kalman, "A new approach to linear filter and prediction problems," *J. Basic Eng.*, vol. 82, pp. 35–45, 1960.

[3] R. E. Kalman and R. S. Bucy, "New results of linear filtering and prediction theory," *J. Basic Eng.*, vol. 83, pp. 95–108, 1961.

[4] O. Talagrand and P. Courtier, "Variational assimilation of meteorological observations with the adjoint vorticity equation. I. Theory," *Q. J. R. Meteorol. Soc.*, vol. 113, pp. 1311–1328, 1987.

[5] E. Kalnay, H. Li, T. Miyoshi, S.-C. Yang, and J. Ballabrera-Poy, "4D-Var or ensemble Kalman filter?," *Tellus A*, vol. 59, pp. 758–773, 2007.

[6] E. J. Fertig, J. Harlim, and B. R. Hunt, "A comparative study of 4D-Var and a 4D ensemble Kalman filter: Perfect model simulations with Lorenz-96," *Tellus A*, vol. 59, pp. 96–101, 2006.

[7] G. Evensen, "Using the extended Kalman filter with a multilayer quasi-geostrophic ocean model," *J. Geophys. Res.*, vol. 97, no. C11, pp. 17905–17924, 1992.

[8] G. Evensen, *Data Assimilation: The Ensemble Kalman Filter*. New York: Springer, 2007.

[9] G. Burgers, P. J. van Leeuwen, and G. Evensen, "Analysis scheme in the ensemble Kalman filter," *Mon. Weather Rev.*, vol. 126, pp. 1719–1724, 1998.

[10] J. L. Anderson, "An ensemble adjustment Kalman filter for data assimilation," *Mon. Weather Rev.*, vol. 129, pp. 2884–2903, 2001.

[11] J. S. Whitaker and T. M. Hamill, "Ensemble data assimilation without perturbed observations," *Mon. Weather Rev.*, vol. 130, pp. 1913–1924, 2002.

[12] C. H. Bishop, B. J. Etherton, and S. J. Majumdar, "Adaptive sampling with the ensemble transform Kalman filter. Part I: Theoretical aspects," *Mon. Weather Rev.*, vol. 129, pp. 420–436, 2001.
[13] M. K. Tippett, J. L. Anderson, C. H. Bishop, T. M. Hamill, and J. S. Whitaker, "Ensemble square-root filters," *Mon. Weather Rev.*, vol. 131, pp. 1485–1490, 2003.
[14] A. Doucet, N. de Freitas, and N. Gordon, Eds. *Sequential Monte Carlo Methods in Practice (Statistics for Engineering and Information Science).* New York: Springer-Verlag, 2001.
[15] C. L. Keppenne and M. Rienecker, "Initial testing of a massively parallel ensemble Kalman filter with the Poseidon isopycnal ocean general circulation model," *Mon. Weather Rev.*, vol. 130, pp. 2951–2965, 2002.
[16] C. L. Keppenne and M. Rienecker, "Assimilation of temperature into an isopycnal ocean general circulation model using a parallel ensemble Kalman filter," *J. Mar. Syst*., vol. 40-41, pp. 363–380, 2003.
[17] C. L. Keppenne, M. Rienecker, N. P. Kurkowski, and D. A. Adamec, "Ensemble Kalman filter assimilation of temperature and altimeter data with bias correction and application to seasonal prediction," *Nonlinear Process. Geophys*., vol. 12, pp. 491–503, 2005.
[18] I. Szunyogh, E. J. Kostelich, G. Gyarmati, D. J. Patil, B. R. Hunt, E. Kalnay, E. Ott, and J. A. Yorke, "Assessing a local ensemble Kalman filter: Perfect model experiments with the national centers for environmental prediction global model," *Tellus A*, vol. 57, pp. 528–545, 2005.
[19] P. L. Houtekamer and H. L. Mitchell, "Ensemble Kalman filtering," *Q. J. R. Meteorol. Soc*., vol. 131, pp. 3269–3289, 2005.
[20] B. R. Hunt, E. J. Kostelich, and I. Szunyogh, "Efficient data assimilation for spatiotemporal chaos: A local ensemble transform Kalman filter," *Physica D*, vol. 230, pp. 112–126, 2007.
[21] I. Szunyogh, E. J. Kostelich, G. Gyarmati, E. Kalnay, B. R. Hunt, E. Ott, E. Satterfield, and J. A. Yorke, "A local ensemble transform Kalman filter data assimilation system for the NCEP global model," *Tellus A*, vol. 60, pp. 113–130, 2008.
[22] J. S. Whitaker, T. M. Hamill, X. Wei, Y. Song, and Z. Toth, "Ensemble data assimilation with the NCEP global forecast system," *Mon. Weather Rev.*, vol. 136, pp. 463–482, 2008.
[23] K. Eben, P. Juru, J. Resler, M. Belda, E. Pelikán, B. C. Krger, and J. Keder, "An ensemble Kalman filter for short-term forecasting of tropospheric ozone concentrations," *Q. J. R. Meteorol. Soc*., vol. 131, pp. 3313–3322, 2005.
[24] O. Barrero Mendoza, B. De Moor, and D. S. Bernstein, "Data assimilation for magnetohydrodynamics systems," *J. Comput. Appl. Math*., vol. 189, no. 1, pp. 242–259, 2006.
[25] Y. Zhou, D. McLaughlin, and D. Entekhabi, "An ensemble-based smoother with retrospectively updated weights for highly nonlinear systems," *Mon. Weather Rev*., vol. 135, pp. 186–202, 2007.
[26] M. Eknes and G. Evensen, "Parameter estimation solving a weak constraint variational formulation for an Ekman model," *J. Geophys. Res*., vol. 102, no. C6, pp. 12479–12491, 1997.
[27] J. C. Muccino and A. F. Bennett, "Generalized inversion of the Korteweg-de Vries equation," *Dyn. Atmos. Oceans*, vol. 35, pp. 227–263, 2001.
[28] A. F. Bennett, *Inverse Methods in Physical Oceanography*. Cambridge, U.K.: Cambridge Univ. Press, 1992.
[29] A. F. Bennett, *Inverse Modeling of the Ocean and Atmosphere*. Cambridge, U.K.: Cambridge Univ. Press, 2002.
[30] G. Evensen, J. Hove, H. C. Meisingset, E. Reiso, K. S. Seim, and Ø. Espelid, "Using the EnKF for assisted history matching of a North Sea reservoir model," Houston, TX, Soc. Petroleum Eng., Inc., SPE 106184, 2007.
[31] R. N. Miller, "Perspectives on advanced data assimilation in strongly nonlinear systems," in *Data Assimilation: Tools for Modelling the Ocean in a Global Change Perspective (NATO ASI, vol. I 19)*, P. P. Brasseur and J. C. J. Nihoul, Eds. Berlin: Springer-Verlag, 1994, pp. 195–216.
[32] R. N. Miller, M. Ghil, and F. Gauthiez, "Advanced data assimilation in strongly nonlinear dynamical systems," *J. Atmos. Sci*., vol. 51, pp. 1037–1056, 1994.
[33] P. Gauthier, P. Courtier, and P. Moll, "Assimilation of simulated wind lidar data with a Kalman filter," *Mon. Weather Rev*., vol. 121, pp. 1803–1820, 1993.
[34] F. Bouttier, "A dynamical estimation of forecast error covariances in an assimilation system," Mon. Weather Rev., vol. 122, pp. 2376–2390, 1994.
[35] A. H. Jazwinski, *Stochastic Processes and Filtering Theory*. San Diego, CA: Academic, 1970.
[36] R. H. Reichle, D. B. McLaughlin, and D. Entekhabi, "Hydrologic data assimilation with the ensemble Kalman filter," *Mon. Weather Rev*., vol. 130, pp. 103–114, 2002.
[37] G. Evensen, "The ensemble Kalman filter: Theoretical formulation and practical implementation," *Ocean Dyn*., vol. 53, pp. 343–367, 2003.
[38] G. Evensen and P. J. van Leeuwen, "An ensemble Kalman smoother for nonlinear dynamics," *Mon. Weather Rev*., vol. 128, pp. 1852–1867, 2000.
[39] P. J. van Leeuwen and G. Evensen, "Data assimilation and inverse methods in terms of a probabilistic formulation," *Mon. Weather Rev*., vol. 124, pp. 2898–2913, 1996.
[40] G. Evensen, "Advanced data assimilation for strongly nonlinear dynamics," *Mon. Weather Rev*., vol. 125, pp. 1342–1354, 1997.
[41] E. N. Lorenz, "Deterministic nonperiodic flow," *J. Atmos. Sci*., vol. 20, pp. 130–141, 1963.
[42] G. Evensen, "Sampling strategies and square root analysis schemes for the EnKF," *Ocean Dyn*., vol. 54, pp. 539–560, 2004.
[43] O. Leeuwenburgh, "Assimilation of along-track altimeter data in the Tropical Pacific region of a global OGCM ensemble," *Q. J. R. Meteorol. Soc.*, vol. 131, pp. 2455–2472, 2005.
[44] P. Sakov and P. R. Oke, "Implications of the form of the ensemble transform in the ensemble square root filters," *Mon. Weather Rev*., vol. 136, no. 3, pp. 1042–1053, 2008.
[45] X. Wang, C. H. Bishop, and S. J. Julier, "Which is better, an ensemble of positive-negative pairs or a centered spherical simplex ensemble," *Mon. Weather Rev*., vol. 132, pp. 1590–1605, 2004.
[46] O. Leeuwenburgh, G. Evensen, and L. Bertino, "The impact of ensemble filter definition on the assimilation of temperature profiles in the Tropical Pacific," *Q. J. R. Meteorol. Soc*., vol. 131, pp. 3291–3300, 2005.
[47] D. M Livings, S. L. Dance, and N. K. Nichols, "Unbiased ensemble square root filters," *Physica D*, vol. 237, pp. 1021–1028, 2008.
[48] P. L. Houtekamer and H. L. Mitchell, "A sequential ensemble Kalman filter for atmospheric data assimilation," *Mon. Weather Rev*., vol. 129, pp. 123–137, 2001.
[49] T. M. Hamill, J. S. Whitaker, and C. Snyder, "Distance-dependent filtering of background error covariance estimates in an ensemble Kalman filter," *Mon. Weather Rev*., vol. 129, pp. 2776–2790, 2001.
[50] J. L. Anderson, "A local least squares framework for ensemble filtering," *Mon. Weather Rev*., vol. 131, pp. 634–642, 2003.
[51] V. E. Haugen and G. Evensen, "Assimilation of SLA and SST data into an OGCM for the Indian ocean," *Ocean Dyn*., vol. 52, pp. 133–151, 2002.
[52] K. Brusdal, J. M. Brankart, G. Halberstadt, G. Evensen, P. Brasseur, P. J. van Leeuwen, E. Dombrowsky, and J. Verron, "An evaluation of ensemble based assimilation methods with a layered OGCM," *J. Mar. Syst*., vol. 40-41, pp. 253–289, 2003.
[53] E. Ott, B. Hunt, I. Szunyogh, A. V. Zimin, E. Kostelich, M. Corazza, E. Kalnay, D. J. Patil, and J. A. Yorke, "A local ensemble Kalman filter for atmospheric data assimilation," *Tellus A*, vol. 56, pp. 415–428, 2004.
[54] H. L. Mitchell, P. L. Houtekamer, and G. Pellerin, "Ensemble size, and model-error representation in an ensemble Kalman filter," *Mon. Weather Rev*., vol. 130, pp. 2791–2808, 2002.
[55] J. L. Anderson, "Exploring the need for localization in the ensemble data assimilation using a hierarchical ensemble filter," *Physica D*, vol. 230, pp. 99–111, 2007.
[56] C. H. Bishop and D. Hodyss, "Flow-adaptive moderation of spurious ensemble correlations and its use in ensemble-based data assimilation," *Q. J. R. Meteorol. Soc*., vol. 133, pp. 2029–2044, 2007.
[57] E. J. Fertig, B. R. Hunt, E. Ott, and I. Szunyogh, "Assimilating nonlocal observations with a local ensemble Kalman filter," *Tellus A*, vol. 59, pp. 719–730, 2007.
[58] S. P. Khare, J. L. Anderson, T. J. Hoar, and D. Nychka, "An investigation into the application of an ensemble Kalman smoother to high-dimensional geophysical systems," *Tellus A*, vol. 60, pp. 97–112, 2008.
[59] J. L. Anderson and S. L. Anderson, "A Monte Carlo implementation of the nonlinear filtering problem to produce ensemble assimilations and forecasts," *Mon. Weather Rev*., vol. 127, pp. 2741–2758, 1999.
[60] D. T. Pham, "Stochastic methods for sequential data assimilation in strongly nonlinear systems," *Mon. Weather Rev*., vol. 129, pp. 1194–1207, 2001.
[61] X. Wang and C. H. Bishop, "A comparison of breeding and ensemble transform Kalman filter ensemble forecast schemes," *J. Atmos. Sci*., vol. 60, pp. 1140–1158, 2003.
[62] J. L. Anderson, "An adaptive covariance inflation error correction algorithm for ensemble filters," Tellus A, vol. 59, pp. 210–224, 2007.
[63] H. Li, E. Kalnay, and T. Miyoshi, "Simultaneous estimation of covariance inflation and observation errors within ensemble Kalman filter," *Q. J. R. Meteorol. Soc*., vol. 135, no. 639, pp. 523–533, 2009.

Application of a dual unscented Kalman filter for simultaneous state and parameter estimation in problems of surface-atmosphere exchange

J. H. Gove[1] and D. Y. Hollinger[1]

Received 25 March 2005; revised 1 August 2005; accepted 10 December 2005; published 22 April 2006.

[1] A dual unscented Kalman filter (UKF) was used to assimilate net CO_2 exchange (NEE) data measured over a spruce-hemlock forest at the Howland AmeriFlux site in Maine, USA, into a simple physiological model for the purpose of filling gaps in an eddy flux time series. In addition to filling gaps in the measurement record, the UKF approach provides continuous estimates of model parameters and uncertainty. The process explicitly recognizes uncertainty in the measurement data and model structure, providing approximate, effectively optimal state and parameter estimates with less subjectivity than in many previous gap-filling methods. The dual UKF is a recursive predictor-corrector estimation method whereby noisy measurement data are used to continuously update nonlinear process model predictions of the desired states, in this case net ecosystem exchange, among others. Two parallel filters are run simultaneously in the dual approach, one for state and the other for parameter estimation. The unscented transformation employs a deterministic sampling of "sigma points" from the joint density that captures the first two moments of the distribution to the second order. Nonlinear process models are applied to these sigma points to propagate the joint density within the filter framework. The UKF estimate of annual NEE in 2000 at the Howland Forest totaled -296.4 ± 2.4 g carbon m^{-2} (mean ± standard deviation) using nocturnal data when the square root of the momentum flux (u^*) exceeded 0.25 m s^{-1}. This NEE value is about 9% higher than a previous estimate where gaps were filled by physiological models fitted to monthly, seasonal, and annual data. Model estimates are sensitive to the threshold set for accepting or rejecting nocturnal flux data ("u^* threshold"), and we show that uncertainty in annual estimates is dominated by the choice of u^* threshold.

Citation: Gove, J. H., and D. Y. Hollinger (2006), Application of a dual unscented Kalman filter for simultaneous state and parameter estimation in problems of surface-atmosphere exchange, *J. Geophys. Res.*, *111*, D08S07, doi:10.1029/2005JD006021.

1. Introduction

[2] Exchanges of matter and energy between the surface and atmosphere are fundamental to the operation of the Earth's climate system, and provide critical constraints on many important biogeochemical cycles. The eddy covariance method is increasingly applied to measure surface-atmosphere exchanges of latent and sensible heat, carbon dioxide, and a variety of trace gases [*Baldocchi*, 2003]. This method is robust and reliable when applied over flat, homogeneous surfaces during times of well-developed atmospheric turbulence. Eddy covariance has many advantages including high temporal resolution (data are frequently recorded at 10 or more Hz and integrated over 30–60 minutes), and good spatial integration. Unfortunately, stable atmospheric conditions often occur at night, uncoupling surface exchange from the measurement system. Additionally, eddy covariance equipment may not operate during certain weather conditions (heavy rain, ice), and may fail, like all measurement systems, for a variety of reasons. The result is that long-term flux records have many gaps. If the goal of the research requires complete records, such as for studies of evaporation or carbon sequestration, gaps must be filled. A variety of methods have been applied to address the problem of filling flux data gaps, including linear interpolation, lookup tables, simple process models, neural networks, and Kalman filter approaches [*Falge et al.*, 2001a, 2001b; *van Wijk and Bouten*, 1999; *Jarvis et al.*, 2004], with simple process models being the predominant approach.

[3] Another important use for eddy flux data is to provide parameter estimates for biological models that predict fluxes on the basis of environmental data such as solar radiation and temperature [*Luo and Reynolds*, 1999; *Reichstein et al.*, 2003]. Those using this data-based or inverse modeling approach with flux data have derived model parameters using nonlinear least squares, maximum likelihood, and Kalman filter methods [e.g., *Hollinger et al.*, 1999; *van Wijk and Bouten*, 2002; *Jarvis et al.*, 2004; *Williams et al.*, 2005].

[1]Northeastern Research Station, U.S. Department of Agriculture, Forest Service, Durham, New Hampshire, USA.

[4] The Kalman filter is an optimal, minimum mean square error estimator for linear systems. When system dynamics are intrinsically nonlinear, the extended Kalman filter has customarily been used, although other approaches such as statistical linearization have been developed [*Gelb*, 1974, p. 203; *Maybeck*, 1982, p. 243]. In general, the extended Kalman filter performs a truncated first-order Taylor linearization on the system equations about the current state, to which the linear filter equations are applied. The extended Kalman filter has been used extensively [e.g., *Gelb*, 1974; *Jazwinski*, 1970; *Lewis*, 1986], however, it does suffer from possible divergence problems because the linearization does not always capture the correct dynamics of the underlying system [e.g., *Jazwinski*, 1969; *Fitzgerald*, 1971]. As a result, several new filtering methods have recently been introduced on the basis of the Kalman filter. Rather than seeking to linearize the nonlinear dynamics of the system, these new derivativeless methods deterministically sample the joint density of the states in such a way that the mean and covariance are preserved. The full nonlinear system dynamics are then applied to these sample points in order to propagate the density through the prediction step of the filter.

[5] Kalman filters generally take the form of a two step recursion, unless these steps have been purposefully combined [e.g., *Lewis*, 1986, p. 70]. In the prediction step, the filter uses the system process equations to perform a time update on the previous filter states before any new measurements are available at the next time period. Once the measurements become available at the new time period, they are combined in an optimal adjustment to the previous prediction of the states in the measurement update step. This two-step predictor-corrector recursion is then applied at each successive time period.

[6] A formulation to the nonlinear system equations that is adopted here assumes additive noise terms plus unknown parameters. In this case, the equations can be written

$$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}; \mathbf{w}_{k-1}) + \boldsymbol{\nu}_{k-1} \tag{1}$$

$$\mathbf{y}_k = \mathbf{h}(\mathbf{x}_k; \mathbf{w}_{k-1}) + \boldsymbol{\eta}_k \tag{2}$$

where $\mathbf{x}_k$ ($n_x \times 1$) is the state of the system at time k, $\mathbf{w}_k$ ($n_w \times 1$) are parameters, which both must be estimated simultaneously from the noisy measurements $\mathbf{y}_k$ ($n_y \times 1$). In addition, the $\boldsymbol{\nu}_k$ are the zero-mean random process noises with covariance $\mathbf{Q}_k$ that drive the nonlinear system function $\mathbf{f}$. The $\boldsymbol{\eta}_k$ are the zero-mean measurement noises with covariance $\mathbf{R}_k$ that have corrupted the measurements $\mathbf{y}_k$. The system function $\mathbf{h}$ may be either linear or nonlinear. Note that no normality assumption has been placed on the noise sequences. Kalman's original derivation [*Kalman*, 1960] did not assume normality, only that the joint density of the system states could be propagated by their first two moments. In addition, many other formulations to the system equations exist; these allow for, among other things, deterministic inputs and correlated noise sequences [*Gelb*, 1974; *Maybeck*, 1982].

[7] The Kalman filter was developed for time series where the data are autocorrelated, which makes it an attractive candidate for estimating missing flux data. In this paper we demonstrate the use of the unscented Kalman filter (UKF) for estimating annual net ecosystem CO_2 exchange (NEE) using incomplete and noisy eddy covariance data recorded above a spruce-hemlock forest in the northeastern U.S. In addition to providing an objective and unbiased method for filling gaps in the data record, we show how interpretations derived from the time-varying model parameter estimates from this filter are consistent with our understanding of the ecological processes that regulate surface-atmosphere exchanges.

2. Methods

2.1. Flux Measurements

[8] Measurements were made at the Howland Forest AmeriFlux site in central Maine, USA (45°15′N, 68°44′W, 60 m asl). This site is composed of approximately 20 m tall commercial softwood forest owned by GMO Renewable Resources, LLC. Forest stands are dominated by red spruce (*Picea rubens* Sarg.) and eastern hemlock (*Tsuga canadensis* (L.) Carr.) with lesser quantities of other conifers and hardwoods. Forest live basal area measured in plots around the main Howland research tower (45.20407°N, 68.74020°W) was about 50 m^2 ha^{-1} with live biomass about 120 t C ha^{-1} [*Hollinger et al.*, 2004]. Fluxes were measured at a height of 29 m with a system consisting of model SAT-211/3K 3-axis sonic anemometer (Applied Technologies, Inc., Longmont, Colorado, USA) and model LI-6262 fast response CO_2/H_2O infrared gas analyzer (LiCor, Inc., Lincoln, Nebraska, USA), with data recorded at 5 Hz. The flux measurement system and calculations are described in detail by *Hollinger et al.* [1999, 2004]. Deficiencies in the low- and high-frequency response of the flux system were corrected by using the Horst/Massman approach of calculating a transfer function based on stability and theoretical spectra [e.g., *Massman and Lee*, 2002] to correct for missing low-frequency contributions and a ratio of filtered to unfiltered heat fluxes to account for missing high-frequency fluctuations. Half-hourly flux values were excluded from further analysis if the wind speed was below 0.5 m s^{-1}, scalar variance was excessively high or extremely low, rain or snow was falling, for incomplete half-hour sample periods, or instrument malfunction. Data from nocturnal periods were excluded when the friction velocity, u^*, was less than a threshold of 0.25 m s^{-1}. The sign convention used is that carbon flux into the ecosystem is defined as negative.

[9] The measurements were collected at 30 minute intervals k, $k = 1,\ldots, N$ in the calendar year 2000 yielding $N =$ 17,568 half-hour measurement intervals in that year. The measurements used in the subsequent analysis are assumed to be corrupted with noise, as formally shown in the filter measurement equation (2). The measurement vector $\mathbf{y}_k$ at time k is given as

$$\begin{aligned}\mathbf{y}_k &= [\mathrm{NEE} \quad \mathrm{PPFD} \quad \mathrm{T}]' \\ &= \begin{bmatrix} y_{1,k} & y_{2,k} & y_{3,k} \end{bmatrix}'\end{aligned}$$

where NEE is net ecosystem exchange of CO_2 (μmol m^{-2} s^{-1}), PPFD is photosynthetically active photon flux density (μmol photons m^{-2} s^{-1}), and T is air temperature

in °C. Diurnal measurement periods were differentiated by defining PPFD $\leq$ 5 μmol m^{-2} s^{-1} as night. There were no missing data in the photon flux and temperature measurements. However, 42 percent of the NEE measurements were missing in the year 2000 because of either instrument failure, or exclusion for one or more of the reasons described previously.

2.2. Process Model

[10] In general, the Kalman filter prediction step is driven by models of the system dynamics. The filter can accommodate a large degree of complexity in the underlying models, e.g., ranging from stochastic differential equations in the continuous time filter, to simple models that predict the state of the system on the basis of the noisy measurements, or even simple random walk formulations when the process structure is unknown. Regardless of the complexity of the underlying models, they should provide a sensible mathematical and biological description of the system. This is especially important in the case where there are large gaps of missing data that require filling because the filtered estimates for these gaps, which rely on the system models, must make sense biologically.

[11] Our goal was to use simple process models that adequately describe the system states with a minimal number of parameters. The reason for this is twofold. First, it serves to illustrate that the Kalman filter does not require overly complex system models in order to produce very good estimates. Second, because we are illustrating a new method, a simple, tractable system model will not detract from the understanding of the filtering techniques employed. As model complexity increases, it becomes increasingly likely that no single set of parameter values will be optimal, but that many model parameter sets will fit the data more or less equally well: equifinality [*Franks et al.*, 1997].

[12] The model for NEE chosen here is the frequently used "big leaf" Michaelis-Menten model, namely,

$$\mathrm{NEE} = \frac{A \times \mathrm{PPFD}}{K + \mathrm{PPFD}} + \mathrm{R} \tag{3}$$

where A is the maximum rate of uptake (μmol CO_2 m^{-2} s^{-1}), K is the half saturation constant (μmol photons m^{-2} s^{-1}), and R is ecosystem respiration (μmol CO_2 m^{-2} s^{-1}). Ecosystem respiration could be treated as either a parameter, or a system state. The latter definition is used here. Thus we chose to model respiration with the simple *Lloyd and Taylor* [1994] model

$$\mathrm{R} = R_p \exp^{\frac{-E_0}{\mathrm{T}+273.15-T_0}} \tag{4}$$

where R_p is baseline respiration rate relative to pool size (μmol CO_2 m^{-2} s^{-1}), E_0 is akin to an activation energy in °K, and $0 \leq T_0 \leq \mathrm{T} + 273.15$.

[13] The system state vector, $\mathbf{x}_k$ at time k is composed of the same first three elements as the measurements, plus R and an estimate of the integration of NEE. The system equation for the integration of NEE (INEE) is discretized for the filter as

$$\mathrm{INEE}_k = \mathrm{INEE}_{k-1} + \mathrm{NEE}_k \Delta k \tag{5}$$

where $\Delta k = 30 \times 60 = 1800$ s is the time step. The remaining two system states (PPFD and T) are modeled as a simple random walk, in lieu of a more complicated description of the process. In general, the random walk model is

$$x_k = x_{k-1} + \nu_k \tag{6}$$

where ν_k is a zero mean random noise perturbation with positive variance. The filter provides a mechanism to include similar random noise variation in each of the process equations. Therefore the final state vector is given as

$$\begin{aligned}\mathbf{x}_k &= [\mathrm{NEE} \quad \mathrm{PPFD} \quad \mathrm{T} \quad \mathrm{INEE} \quad \mathrm{R}]' \\ &= \begin{bmatrix} x_{1,k} & x_{2,k} & x_{3,k} & x_{4,k} & x_{5,k} \end{bmatrix}'\end{aligned}$$

[14] The above system process models are defined for the growing season. However, the Howland research site is characterized by a 4-month-long winter season with temperatures well below freezing and a 1 m deep snowpack. Neither the NEE nor Lloyd and Taylor models are applicable at this time of the year. The NEE model cannot be applied since the first term, photosynthesis, is inhibited by subzero temperatures [*Teskey et al.*, 1995]. The Lloyd and Taylor model fails in winter because our formulation is based solely on air temperature. During the growing season, respiration at Howland comes from both belowground and aboveground sources in a ratio of about 60:40. At this time of year, the soil acts to low-pass air temperature variations, and air temperature provides a reasonable estimate of the system state. In the winter, however, when respiration comes mostly from the soil [*Davidson et al.*, 2006], the deep snowpack uncouples soil temperature from air temperature and so air temperature does not accurately represent the temperature of the respiring material. We therefore exclude T and adjust these two process equations for the dormant season to

$$\mathrm{R} = R_p \exp^{\frac{-E_0}{273.15-T_0}} \tag{7}$$

$$\mathrm{NEE} = \begin{cases} \mathrm{R}, & \mathrm{PPFD} < 5 \mu\mathrm{mol}\ m^{-2}\ s^{-1} (\mathrm{nighttime}) \\ A, & \mathrm{otherwise (daytime)} \end{cases} \tag{8}$$

Note that the model for NEE has different day and nighttime definitions. This was necessary because there can be warm days in the fall, and during midwinter thaws, subsequent to the switch to the dormant season models, in which photosynthesis occurs. Unfortunately, however, the modified Lloyd and Taylor model (7) only allows for respiration to occur. The maximum rate of uptake parameter, A, has therefore been used during the daytime of the dormant season as a simple time-varying parameter to accommodate this potential late-season photosynthetic activity. Note that it also accommodates daytime respiration for the majority of days where there is no photosynthesis occurring.

[15] Determination of the initiation and completion of the growing season was made using a deterministic switch based

on recorded soil temperature. When the daily soil temperature reaches 0°C, in both spring and fall, the switch is made to the alternate process model formulations. For the year 2000, this happened at Julian days 84 and 328. In our current model, the switch occurs only once each in spring and fall; alternative possible formulations will be discussed later.

[16] There are five system parameters in this model formulation: A, K, R_p, E_0 and T_0. However, because of a high degree of correlation between E_0 and T_0, we fixed T_0 = 261.2°K, on the basis of a least squares fit of the June 2000 data to model (4). This left a total of four unknown parameters, which are collected into the parameter vector $\mathbf{w}_k$, namely,

$$\begin{aligned}\mathbf{w}_k &= \begin{bmatrix} A & K & E_0 & R_p \end{bmatrix}' \\ &= \begin{bmatrix} w_{1,k} & w_{2,k} & w_{3,k} & w_{4,k} \end{bmatrix}'\end{aligned}$$

[17] The vectors $\mathbf{y}_k$, $\mathbf{x}_k$ and $\mathbf{w}_k$ make up the system noisy measurements, unknown states and unknown parameters, respectively. It should be noted explicitly that the parameters are time-varying in this formulation.

2.3. Scaled Unscented Transformation

[18] The unscented transformation (UT) was first introduced by *Uhlmann* [1995]. The main idea behind the unscented transformation, alluded to earlier, is to completely capture the first two moments of the joint density in both of the filter steps with a deterministic sampling of "sigma points" from that density, and subsequently apply the nonlinear dynamics to these sampled points. It was founded on the simple observation that it should be easier to approximate a probability distribution than an arbitrary nonlinear function [*Uhlmann*, 1995; *Julier and Uhlmann*, 2004]. It should be noted that the unscented transformation is quite general in that it can be applied to any problem that requires nonlinear transformations of probability distributions, not just filtering problems. However, the main motivation for its development was to address the shortcomings of the EKF linearization approach [*Julier and Uhlmann*, 2004]. These shortcomings are manifested in two primary ways. First, derivation of the Jacobian matrices used in the extended Kalman filter, if they exist, can be difficult, and programming these often large calculations is error prone. Secondly, it has been known for some time that the extended Kalman filter can indeed diverge because the truncation of higher-order terms in the Taylor linearization of the dynamics can yield poor mean and covariance estimates of the system state [e.g., *Jazwinski*, 1969; *Fitzgerald*, 1971]. Some corrections and filter "tuning" can be made to address this second issue, such as Monte Carlo analysis and the addition of stabilizing noise. However, ameliorations must be undertaken on a case-by-case basis, as such problems tend to be problem-specific. In addition, these methods may actually result in the unintended consequence of inflating the variance in some portions of the state space, while providing no correction for the bias [*Julier and Uhlmann*, 2004]. Other specialized methods also exist, but are complex and may be applicable only to specific classes of problems, e.g., where the error distributions are Gaussian.

[19] *Julier and Uhlmann* [1997, 2004] present the derivation for the UT along with an example inherent to target tracking, illustrating that converting from polar to Cartesian coordinates using linearization is both biased and inconsistent. For comparison, the UT shows no bias and correctly captures the state covariance. The basic UT has been developed further, to incorporate higher-order moments of the distribution, among other things [e.g., *Tenne and Singh*, 2003; *Julier and Uhlmann*, 2004]. Extending the sigma point set to incorporate higher-order moments requires a larger set of points and increases the computational burden in the filter. Pragmatically, if only the first two moments of the distribution are estimated, as in the Kalman filter, using an extended sigma point set incorporating higher-order moments, such as skewness, could actually be detrimental, for example, if the skew of the sigma point set does not align with that of the true distribution. However, the scaled sigma point set can incorporate the effect of higher-order moments through a scaling constant that shrinks or expands the set as desired [*Julier and Uhlmann*, 2004]. Choosing a small scaling factor concentrates the points about the mean, and thus minimizes the effect of higher-order moments, and vice versa. An additional weighting scheme allows incorporation of higher-order moments, such as kurtosis, when known.

[20] The UT and UKF have been applied to nonlinear estimation problems in a variety of fields such as global positioning systems [*van der Merwe and Wan*, 2004], spacecraft attitude estimation [*Crassidis and Markley*, 2003], ballistic missile tracking [*Saulson and Chang*, 2004] and object tracking in image analysis systems [*Chen et al.*, 2002]. In this paper, we use a modification of the scaled unscented transformation, originally developed by *Julier and Uhlmann* [2002]; the modification is due to *van der Merwe* [2004], who presented an alternative rearrangement that allows the sigma point selection and scaling to occur in one step, thereby reducing calculations. As before, let n_x be the dimension of the state space, $\mathbf{x}$, with mean $\bar{\mathbf{x}}$ and covariance $\mathbf{P}_\mathbf{x}$. Select $2n_x + 1$ sigma points by letting $\lambda = \alpha^2(n_x + \kappa) - n_x$ and drawing

$$\begin{aligned}
\mathcal{X}_0 &= \bar{\mathbf{x}} \\
\mathcal{X}_i &= \bar{\mathbf{x}} + \left(\sqrt{(n_x+\lambda)\mathbf{P}_\mathbf{x}}\right)_i && i = 1, \ldots, n_x \\
\mathcal{X}_i &= \bar{\mathbf{x}} - \left(\sqrt{(n_x+\lambda)\mathbf{P}_\mathbf{x}}\right)_{i-n_x} && i = n_x+1, \ldots, 2n_x \\
w_0^{(m)} &= \frac{\lambda}{n_x+\lambda} && i = 0 \\
w_0^{(c)} &= \frac{\lambda}{n_x+\lambda} + \left(1-\alpha^2+\beta\right) && i = 0 \\
w_i^{(m)} &= w_i^{(c)} = \frac{1}{2(n_x+\lambda)} && i = 1, \ldots, 2n_x
\end{aligned} \tag{9}$$

The sigma point set $\mathcal{S} = \{\mathcal{X}_i, w_i^{(j)}; i = 0, \ldots, 2n_x, j \in (m, c)\}$ is composed of the sigma points $\mathcal{X}_i$ and their respective mean (m) and covariance (c) weights $w_i^{(j)}$. The weights can be positive or negative and must sum to one [*Julier and Uhlmann*, 2004]. The parameters $0 \leq \alpha \leq 1$ and $\beta \geq 0$ control the spread of the sigma points and weighting for higher-order moments; parameter $\kappa \geq 0$ is not critical and is often set to zero [*van der Merwe*, 2004, p. 56]. For α, the

smaller the value, the smaller the sigma-point spread and the less likely to pick up anomalous effects in the distribution. For Gaussian distributions, $\beta = 2$ is optimal [*Julier and Uhlmann*, 2002, 2004].

[21] Given an arbitrary nonlinear transformation on the state $\mathbf{g}$, the scaled unscented transformation consists of the following four steps [*van der Merwe*, 2004, p. 56]:

[22] 1. Choose the parameters α, β and κ as described.

[23] 2. Determine the set $\mathcal{S}$ of $2n_x + 1$ sigma points and associated weights from (9).

[24] 3. Apply the nonlinear transformation $\mathbf{g}$ to each sigma point, namely,

$$\mathcal{Y}_i = \mathbf{g}(\mathcal{X}_i), \qquad i = 0, \ldots, 2n_x$$

[25] 4. Calculate the mean, covariance and cross covariance, respectively, of the transformed sigma points as

$$\bar{\mathbf{y}} = \sum_{i=0}^{2n_x} w_i^{(m)} \mathcal{Y}_i$$

$$\mathbf{P}_{\mathbf{y}} = \sum_{i=0}^{2n_x} w_i^{(c)} (\mathcal{Y}_i - \bar{\mathbf{y}})(\mathcal{Y}_i - \bar{\mathbf{y}})'$$

$$\mathbf{P}_{\mathbf{xy}} = \sum_{i=0}^{2n_x} w_i^{(c)} (\mathcal{X}_i - \bar{\mathbf{x}})(\mathcal{Y}_i - \bar{\mathbf{y}})'$$

[26] A simple example will provide some intuition into the UT and the effect that the choice of parameter values can have in drawing the sigma points. Let the untransformed state vector for this example be composed of PPFD and T. The mean and covariance were calculated from June 2000 measurements where PPFD $\geq$ 1500 μmol photons m^{-2} s^{-1}, yielding

$$\bar{\mathbf{x}} = [1753.54 \quad 16.5]$$

$$\mathbf{P}_{\mathbf{x}} = \begin{bmatrix} 18766 & -62.3 \\ -62.3 & 16.6 \end{bmatrix}$$

The transformation $\mathbf{g}$ is from a maximum likelihood fit of models (3) and (4) to the June 2000 data

$$\mathrm{R} = 35 \exp^{\frac{-46.4}{\mathrm{T}+273.15-262.2}}$$

$$\mathrm{NEE} = \frac{-20 \times \mathrm{PPFD}}{524 + \mathrm{PPFD}} + \mathrm{R}$$

Thus the states PPFD and T are mapped into the states NEE and R through $\mathbf{g}$. The results of applying the unscented transformation on this example with $\beta = 2$ are shown in Figure 1. In Figure 1a, $\alpha = 1$, $\kappa = 0.15$ and it will be noted that the sigma points align with the 66 percent confidence ellipse. In addition, note that the weight $w_0^{(m)}$ is small and positive. In Figure 1b, $\alpha = 0.4$, $\kappa = 0$; the resulting set of points is much tighter around the mean, illustrating the effect of reducing α. As mentioned above, adjusting the spread of the sigma points allows one to include the effect of higher-order moments if desired. Finally, notice that the weight $w_0^{(m)}$ is large and negative. The bottom panel shows the effect of applying the nonlinear mapping $\mathbf{g}$. Note that the sigma points have transformed accordingly to capture the new mean and covariance of the states NEE and R. In addition, this example serves to illustrate the notable difference that can occur in the first two moments with even a simple nonlinear transformation.

[27] This simple example also illustrates that while the sigma points capture the essential information of the joint distribution, they are not a probability density function [*Julier and Uhlmann*, 2004]. Thus the weights can be positive or negative, providing they meet the constraint of summing to unity. As mentioned by *Julier and Uhlmann* [2004], there are a number of important properties of the UT. First, it is algorithmic, and can be easily programmed in a "black-box" sense regardless of the complexity of the nonlinear transformations, with computational cost on the same order of magnitude as the EKF. Relatedly, because the sigma points can straddle discontinuities, $\mathbf{g}$ need not be a continuous function set. Finally, the sigma points from the UT, including the scaled UT, calculates the projected mean and covariance to the second order (i.e., in the Taylor series expansion) and therefore implicitly includes a second-order "bias correction" that is not in the EKF.

2.4. Dual Unscented Kalman Filter

[28] A number of methods have been developed to handle uncertainty in parameter estimates in addition to state estimation. For example, the Schmidt-Kalman filter [*Jazwinski*, 1970, p. 285] accounts for the effect of uncertain parameters on the estimate of the state without explicitly estimating the parameters themselves. In addition, the state-dependent approach [*Young*, 2000] is a general method used to estimate unknown states and parameters considered to be statistically dependent, through a multistep autoregressive filtering and modeling process. Two other general methods, have been developed to simultaneously estimate the unknown states and parameters from the noisy measurements using the Kalman filter, namely, joint and dual estimation [*van der Merwe*, 2004; *Wan and van der Merwe*, 2001]. While the former methods are noteworthy, we concentrate on the latter two methods, specifically the dual filtering paradigm.

[29] The joint Kalman filter takes the approach of augmenting the state vector with the parameters to form a joint state vector $\tilde{\mathbf{x}}_k = [\mathbf{x}_k', \mathbf{w}_k']'$ [*Gelb*, 1974, p. 348; *Jazwinski*, 1970, p. 282]. This joint state is now estimated through a single Kalman filter recursion. The alternative is to run two parallel filters, one on the state and the other on the parameters; this is known as the dual approach. In the dual setting, the parameters are treated as known within the state filter at any given time, k, while the states are treated as known in the parallel parameter filter. Both the joint and dual approaches can be run within either the extended or unscented Kalman filter frameworks [*van der Merwe*, 2004; *Wan and Nelson*, 2001; *Wan and van der Merwe*, 2001]. The main difference between the two approaches, aside from the number of filters required, is that the joint filter explicitly allows for cross parameter and state dependencies. For example, let the joint covariance for the augmented state be

$$\mathbf{P}_k = \begin{bmatrix} \mathbf{P}_{\mathbf{x}_k} & \mathbf{P}_{\mathbf{x}_k \mathbf{w}_k} \\ \mathbf{P}_{\mathbf{w}_k \mathbf{x}_k} & \mathbf{P}_{\mathbf{w}_k} \end{bmatrix}$$

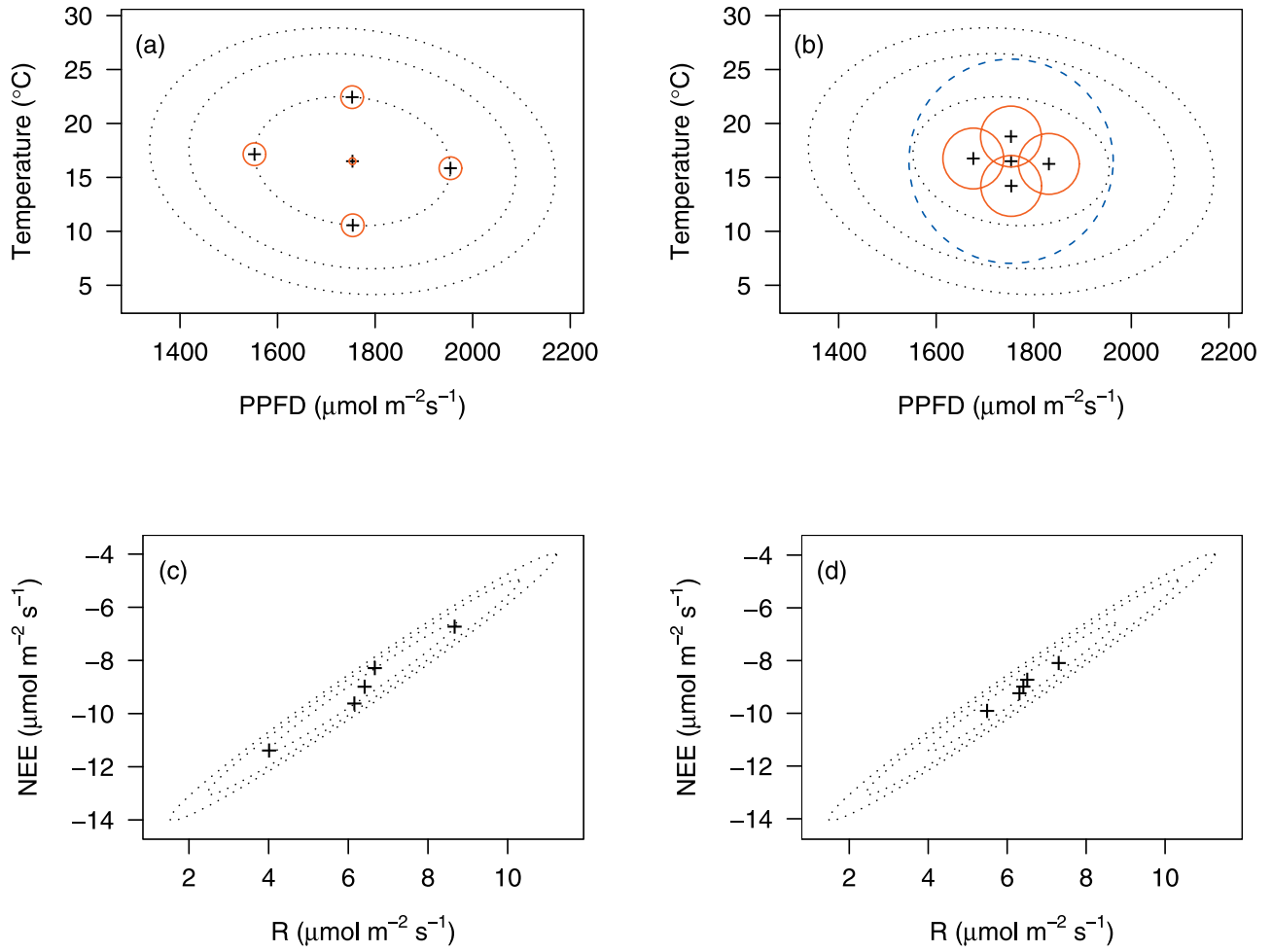


Figure 1. Bivariate sigma points (a and b) before transformation and (c and d) after transformation. The left- and right-hand plots show the effect that differing values for parameters α and κ have on the spread and weighting of the sigma points. In Figures 1a and 1b, the sigma points (pluses) are weighted proportional to $\alpha w_i^{(m)}$ as shown by the circles; solid circles imply positive weights, and dashed circles imply negative weights. The confidence ellipses are for a bivariate normal distribution at the 66, 90 and 95 percentiles. Figures 1c and 1d illustrate the transformed sigma points. Details on parameter settings are given in the text.

In the dual filtering approach the cross covariances are not explicitly estimated, so that it effectively assumes that the cross covariances $\mathbf{P}_{\mathbf{x}_k\mathbf{w}_k} = \mathbf{P}_{\mathbf{w}_k\mathbf{x}_k} = 0$. It could be argued, therefore, that if correlation is suspected between states and parameters, the joint approach would be preferred [*van der Merwe*, 2004, p. 104]. However, experiments performed by *van der Merwe* [2004] show little difference between the two approaches. The reason may be due to the fact that switching parameters and states between the dual filters, coupled with using the exact same measurements in both filters, acts as a type of constraint on the filters and implicitly develops the cross covariance terms, as our results will demonstrate.

[30] The joint and dual unscented Kalman filters have been largely developed to address problems in estimation with neural networks such as in the control of unmanned aerial vehicles [*van der Merwe*, 2004; *Wan and van der Merwe*, 2001]. In neural network problems, unknown parameters are often referred to as weights, and the estimation of the weights is often termed identification. *Sitz et al.* [2002] have also applied the joint UKF to a continuous time estimation scheme for nonlinear systems. They demonstrate the efficacy of the technique using the classical Lotka-Volterra and Lorenz systems. In addition, *VanDyke et al.* [2004] have applied the dual UKF approach to spacecraft attitude dynamics estimation problems. In the following, the dual UKF is presented in the form of the state and parameter filters. While the filtering equations themselves are similar, the differences in the system models require separate presentations.

2.4.1. State Filter

[31] *Julier and Uhlmann* [2004], *van der Merwe* [2004], and *Wan and van der Merwe* [2001] present a general system framework allowing for correlated system or process noise [see also *Gelb*, 1974, p. 78]. However, it is often the case that the noise structure is assumed to be additive,

which is the assumption here. The dual UKF system model for the state with additive stochastic noise is

$$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}; \hat{\mathbf{w}}_{k-1}) + \boldsymbol{\nu}_{k-1} \tag{10}$$

$$\mathbf{y}_k = \mathbf{h}(\mathbf{x}_k; \hat{\mathbf{w}}_{k-1}) + \boldsymbol{\eta}_k \tag{11}$$

with $\boldsymbol{\nu}_{k-1}$ and $\boldsymbol{\eta}_k$ noise sequences as defined in (1) and (2). The $\hat{\mathbf{w}}_{k-1}$ are estimates of the parameters from the previous time step in the parameter filter and are treated as constants in the state filter. The state filter recursions are given in the following steps [*Julier and Uhlmann*, 2004; *van der Merwe*, 2004; *Wan and van der Merwe*, 2001]:

[32] 1. Initialize the filter:

$$\hat{\mathbf{x}}_0 = \mathrm{E}[\mathbf{x}_0], \qquad \mathbf{P}_{\mathbf{x}_0} = \mathrm{E}\left[(\mathbf{x}_0 - \hat{\mathbf{x}}_0)(\mathbf{x}_0 - \hat{\mathbf{x}}_0)'\right]$$

[33] 2. Calculate the sigma point weights as in (9).

[34] 3. Repeat the following for $k = 1,\ldots, N$: (1) Calculate the sigma points as in (9):

$$\mathcal{X}_{k-1} = \left[\hat{\mathbf{x}}_{k-1},\ \hat{\mathbf{x}}_{k-1} + \sqrt{(n_x + \lambda)\mathbf{P}_{\mathbf{x}_{k-1}}},\ \hat{\mathbf{x}}_{k-1} - \sqrt{(n_x + \lambda)\mathbf{P}_{\mathbf{x}_{k-1}}}\ \right] \tag{12}$$

(2) filter prediction equations:

$$\mathcal{X}^*_{k|k-1} = \mathbf{f}(\mathcal{X}_{k-1}; \hat{\mathbf{w}}_{k-1}) \tag{13}$$

$$\hat{\mathbf{x}}_{k|k-1} = \sum_{i=0}^{2n_x} w_i^{(m)} \mathcal{X}^*_{i,k|k-1} \tag{14}$$

$$\mathbf{P}_{\mathbf{x}_{k|k-1}} = \sum_{i=0}^{2n_x} w_i^{(c)} \left(\mathcal{X}^*_{i,k|k-1} - \hat{\mathbf{x}}_{k|k-1}\right)\left(\mathcal{X}^*_{i,k|k-1} - \hat{\mathbf{x}}_{k|k-1}\right)' + \mathbf{Q}_k \tag{15}$$

$$\mathcal{X}_{k|k-1} = \left[\hat{\mathbf{x}}_{k|k-1},\ \hat{\mathbf{x}}_{k|k-1} + \sqrt{(n_x + \lambda)\mathbf{P}_{\mathbf{x}_{k|k-1}}},\ \hat{\mathbf{x}}_{k|k-1} - \sqrt{(n_x + \lambda)\mathbf{P}_{\mathbf{x}_{k|k-1}}}\ \right] \tag{16}$$

$$\mathcal{Y}_{k|k-1} = \mathbf{h}(\mathcal{X}_{k|k-1}; \hat{\mathbf{w}}_{k-1}) \tag{17}$$

$$\hat{\mathbf{y}}_{k|k-1} = \sum_{i=0}^{2n_x} w_i^{(m)} \mathcal{Y}_{i,k|k-1} \tag{18}$$

and (3) filter measurement equations:

$$\mathbf{P}_{\mathbf{y}_k} = \sum_{i=0}^{2n_x} w_i^{(c)} \left(\mathcal{Y}_{i,k|k-1} - \hat{\mathbf{y}}_{k|k-1}\right)\left(\mathcal{Y}_{i,k|k-1} - \hat{\mathbf{y}}_{k|k-1}\right)' + \mathbf{R}_k \tag{19}$$

$$\mathbf{P}_{\mathbf{x}_k\mathbf{y}_k} = \sum_{i=0}^{2n_x} w_i^{(c)} \left(\mathcal{X}_{i,k|k-1} - \hat{\mathbf{x}}_{k|k-1}\right)\left(\mathcal{Y}_{i,k|k-1} - \hat{\mathbf{y}}_{k|k-1}\right)' \tag{20}$$

$$\mathbf{K}_k = \mathbf{P}_{\mathbf{x}_k\mathbf{y}_k}\mathbf{P}_{\mathbf{y}_k}^{-1} \tag{21}$$

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k\left(\mathbf{y}_k - \hat{\mathbf{y}}_{k|k-1}\right) \tag{22}$$

$$\mathbf{P}_{\mathbf{x}_k} = \mathbf{P}_{\mathbf{x}_{k|k-1}} - \mathbf{K}_k\mathbf{P}_{\mathbf{y}_k}\mathbf{K}_k' \tag{23}$$

where $\mathbf{R}_k$ and $\mathbf{Q}_k$ are the system and process noise covariances, respectively. Note that we redraw new sigma points in (16) to incorporate the new information in the prediction density; *van der Merwe* [2004, p. 109] discusses other possible strategies.

[35] In our model formulation, the process models, $\mathbf{f}$, are given by (3)–(8). The measurement model is an identity mapping with

$$\mathbf{h} = \mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

Since this form of $\mathbf{h}$ is linear, we can substitute the linear Kalman filter measurement update recursions [e.g., *Gelb*, 1974, p. 110] for the above UKF measurement recursions, with the state update, (22), remaining the same, namely,

$$\mathbf{K}_k = \mathbf{P}_{\mathbf{x}_{k|k-1}}\mathbf{H}'\left(\mathbf{H}\mathbf{P}_{\mathbf{x}_{k|k-1}}\mathbf{H}' + \mathbf{R}_k\right)^{-1} \tag{24}$$

$$\mathbf{P}_{\mathbf{x}_k} = (\mathbf{I} - \mathbf{K}_k\mathbf{H})\mathbf{P}_{\mathbf{x}_{k|k-1}} \tag{25}$$

where $\mathbf{I}$ is the identity matrix of dimension $(n_x \times n_x)$. There are several other possible formulations and options available for the UKF as presented in the literature cited. Incidentally, the substitution of the linear form of the Kalman recursions for the update step exemplifies the flexibility of the UKF; alternatively, had our process dynamics been linear and our observation dynamics nonlinear, the UKF would have only been used in the update step [*Julier and Uhlmann*, 2004].

2.4.2. Parameter Filter

[36] It is well established [e.g., *Bell and Cathy*, 1993; *Wan and Nelson*, 2001; *Plummer*, 1995] that the EKF can be used in parameter estimation and approximates a recursive modification to Newton's method. The UKF, like the EKF, can be used for parameter estimation for both clean and noisy time series; in the latter case joint or dual estimation strategies are used. *van der Merwe* [2004] has shown that the UKF can be interpreted as an iterative stochastic Gauss-Newton method, which builds up an approximation to the inverse of the Fisher information matrix used in the Fisher scoring step (see *van der Merwe* [2004, chap. 4] for details). In addition, the UKF has been shown to outperform the

EKF in dual estimation in terms of smaller and more consistent covariance estimates, lessening the likelihood for filter divergence.

[37] The system equations for the parameter filter, which is coupled with (10) in the dual setting are

$$\mathbf{w}_k = \mathbf{w}_{k-1} + \boldsymbol{\upsilon}_{k-1} \tag{26}$$

$$\mathbf{y}_k = \mathbf{h}(\mathbf{f}(\hat{\mathbf{x}}_{k-1}; \mathbf{w}_k); \mathbf{w}_k) + \boldsymbol{\epsilon}_k \tag{27}$$

Here the $\boldsymbol{\upsilon}_k$ and $\boldsymbol{\epsilon}_k$ are the process and measurement noises, with covariances $\mathbf{Q}_{\mathbf{w}_k}$ and $\mathbf{R}_{\mathbf{w}_k}$, respectively.

[38] The unscented parameter filter recursion steps are as follows [*van der Merwe*, 2004; *Wan and van der Merwe*, 2001]:

[39] 1. Initialize the filter:

$$\hat{\mathbf{w}}_0 = \mathrm{E}[\mathbf{w}], \qquad \mathbf{P}_{\mathbf{w}_0} = \mathrm{E}\left[(\mathbf{w} - \hat{\mathbf{w}}_0)(\mathbf{w} - \hat{\mathbf{w}}_0)'\right]$$

[40] 2. Calculate the sigma point weights as in (9).

[41] 3. Repeat the following for $k = 1, \ldots, N$: (1) Filter prediction equations:

$$\hat{\mathbf{w}}_{k|k-1} = \hat{\mathbf{w}}_{k-1} \tag{28}$$

$$\mathbf{P}_{\mathbf{w}_{k|k-1}} = \mathbf{P}_{\mathbf{w}_{k-1}} + \mathbf{Q}_{\mathbf{w}_{k-1}} \tag{29}$$

(2) calculate the sigma points as in (9):

$$\mathcal{W}_{k|k-1} = \left[\hat{\mathbf{w}}_{k|k-1},\ \hat{\mathbf{w}}_{k|k-1} + \sqrt{(n_x + \lambda)\mathbf{P}_{\mathbf{w}_{k|k-1}}},\ \hat{\mathbf{w}}_{k|k-1} - \sqrt{(n_x + \lambda)\mathbf{P}_{\mathbf{w}_{k|k-1}}}\ \right] \tag{30}$$

and (3) filter measurement equations:

$$\mathcal{Y}_{k|k-1} = \mathbf{h}\left(\mathbf{f}\left(\mathcal{W}_{k|k-1}; \hat{\mathbf{x}}_{k-1}\right)\right) \tag{31}$$

$$\hat{\mathbf{y}}_{k|k-1} = \sum_{i=0}^{2n_x} w_i^{(m)} \mathcal{Y}_{i,k|k-1} \tag{32}$$

$$\mathbf{P}_{\mathbf{y}_k} = \sum_{i=0}^{2n_x} w_i^{(c)} \left(\mathcal{Y}_{i,k|k-1} - \hat{\mathbf{y}}_{k|k-1}\right)\left(\mathcal{Y}_{i,k|k-1} - \hat{\mathbf{y}}_{k|k-1}\right)' + \mathbf{R}_{\mathbf{w}_k} \tag{33}$$

$$\mathbf{P}_{\mathbf{w}_k\mathbf{y}_k} = \sum_{i=0}^{2n_x} w_i^{(c)} \left(\mathcal{W}_{i,k|k-1} - \hat{\mathbf{w}}_{k|k-1}\right)\left(\mathcal{Y}_{i,k|k-1} - \hat{\mathbf{y}}_{k|k-1}\right)' \tag{34}$$

$$\mathbf{K}_k = \mathbf{P}_{\mathbf{w}_k\mathbf{y}_k}\mathbf{P}_{\mathbf{y}_k}^{-1} \tag{35}$$

$$\hat{\mathbf{w}}_k = \hat{\mathbf{w}}_{k|k-1} + \mathbf{K}_k\left(\mathbf{y}_k - \hat{\mathbf{y}}_{k|k-1}\right) \tag{36}$$

$$\mathbf{P}_{\mathbf{w}_k} = \mathbf{P}_{\mathbf{w}_{k|k-1}} - \mathbf{K}_k\mathbf{P}_{\mathbf{y}_k}\mathbf{K}_k' \tag{37}$$

where $\mathbf{R}_{\mathbf{w}_k}$ is set equal to $\mathbf{R}_k$ in the state filter since the same measurements are used in each filter. Note that because of the nonlinear form of (31), it is not possible to use the linear Kalman filter recursions as replacements for the UKF update step in the parameter filter, as we did in the state filter.

[42] In the above, the diagonal matrix $\mathbf{Q}_{\mathbf{w}_k}$ can be set by one of several different methods [*van der Merwe*, 2004; *Nelson*, 2000]. We used a method analogous to recursive least squares

$$\mathbf{Q}_{\mathbf{w}_k} = \mathrm{diag}\left(\left(\tilde{\lambda}^{-1} - 1\right)\mathbf{P}_{\mathbf{w}_k}\right) \tag{38}$$

with $0 \leq \tilde{\lambda} \leq 1$ known as the forgetting factor and where "diag(·)" means diagonalize the resulting matrix by setting off-diagonals to zero. This scheme adjusts $\mathbf{P}_{\mathbf{w}_{k|k-1}}$ such that it is slightly larger than $\mathbf{P}_{\mathbf{w}_{k-1}}$, which has the effect of discarding older data more quickly. Smaller values of $\tilde{\lambda}$ increase this effect; however, we found that values of $\tilde{\lambda}$ close to one (0.9975) produced the most stable filter results.

2.5. Missing Observations

[43] Missing observations cause gaps in the measurement record that can sometimes be extensive. Filling gaps is a necessary component of the estimation process, whether for periodic (daily, weekly, etc.) or integrated estimates. The yearly integration of NEE is a key component to our flux analysis; therefore minimum mean square estimates of the missing values are a crucial component to obtaining this final estimate.

[44] In our measurement record, the series for PPFD and T were complete, but approximately 42 percent of the NEE observations were missing. In univariate time series, accommodating missing values is simple. The innovations, or prediction errors, are defined as the sequence

$$\mathbf{v}_k = \left(\mathbf{y}_k - \hat{\mathbf{y}}_{k|k-1}\right) \qquad k = 1, \ldots, N \tag{39}$$

Therefore, if a measurement is missing at time k, the innovation is also missing. In the univariate case, this is handled by setting the Kalman gain to zero ($\mathbf{K}_k = 0$), or, more formally by setting $\mathbf{R}$ to infinity. The result is that $\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_{k|k-1}$ and similarly, $\mathbf{P}_k = \mathbf{P}_{k|k-1}$. In other words, the update step could simply be skipped with this assignment [*Kitagawa*, 1999]. The motivation becomes clear when regarding the state update (22), which may be rewritten as $\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k\mathbf{v}_k$. Setting $\mathbf{K}_k = 0$ in the univariate case cancels the effect of the missing innovation. Similar observations may be made for the covariance update since both the covariance ($\mathbf{P}_{\mathbf{y}_k}$) and cross covariance ($\mathbf{P}_{\mathbf{x}_k\mathbf{y}_k}$) terms will be missing. Incidentally, this alternate form of the state update also serves to illustrate the predictor-corrector structure of the Kalman recursions; applying the optimal weighting matrix, $\mathbf{K}_k$, to the innovation generates a correction to be added to the state prediction $\hat{\mathbf{x}}_{k|k-1}$, obtaining $\hat{\mathbf{x}}_k$ [*Maybeck*, 1979, p. 218].

[45] In the multivariate case, where there are gaps for some variables and not for others, there is still valuable information in the nonmissing components of the measurement vector at time k so the update step should not be skipped. Regarding the definition of our measurement

vector $\mathbf{y}_k$, NEE appears as the first measurement variable $y_{1,k}$. Setting the first row of the measurement matrix $\mathbf{H}$ and the corresponding component of the innovation vector to zero will account for the missing observation and allow the information from nonmissing components of $\mathbf{y}_k$ to be utilized in the KF update [*Shumway and Stoffer*, 2000, p. 330]. This same procedure is used in both of the dual filters and for the linear or unscented update step.

[46] The forward pass of the Kalman filter as given in the dual UKF recursions provides minimum mean square estimates for the missing observations conditional on all of the past data. In addition, a smoother can be run, which provides estimates based on all of the data, past and future. If the intent is to use periods of the individual time series for, e.g., weekly estimates, then the smoother will provide estimates with lower variance. However, because the backward recursion of the smoother is initiated with the last state and covariance updates (i.e., $\hat{\mathbf{x}}_N$ and $\mathbf{P}_{\mathbf{x}_N}$), the smoother will not provide different estimates for the final yearly integrated estimate for NEE.

[47] The fixed interval smoother used here was due originally to *Fraser and Potter* [1969], and *Wan and van der Merwe* [2001] have applied it to the UKF state filter. The smoother estimate is developed by running a UKF backward in time, yielding state prediction and covariance ($\hat{\mathbf{x}}_{k|k+1}$, $\mathbf{P}_{\mathbf{x}_{k|k+1}}$) before including the observation at time k in the backward update step. The smoothed estimates are then given by

$$\mathbf{P}^{-1}_{\mathbf{x}_{k|N}} = \mathbf{P}^{-1}_{\mathbf{x}_k} + \mathbf{P}^{-1}_{\mathbf{x}_{k|k+1}} \tag{40}$$

$$\hat{\mathbf{x}}_{k|N} = \mathbf{P}_{\mathbf{x}_{k|N}} \left(\mathbf{P}^{-1}_{\mathbf{x}_{k|k+1}} \hat{\mathbf{x}}_{k|k+1} + \mathbf{P}^{-1}_{\mathbf{x}_k} \hat{\mathbf{x}}_k \right) \tag{41}$$

Equation (40) leads to the conclusion that $\mathbf{P}_{\mathbf{x}_{k|N}} \leq \mathbf{P}_{\mathbf{x}_k}$. Therefore the smoothed estimate will be at least as precise as the filtered estimate, and generally will be better. The exception is the terminal time N when they are identical [*Maybeck*, 1982, p. 7]. We use this approach to develop dual unscented Kalman smoother (UKS) estimates for the states in gap filling.

2.6. Filter Noise Covariances and Initial Conditions

[48] In the Kalman filter, the measurement and process noise covariances are assumed known without error. Determination of the noise covariance matrices and initial conditions for the Kalman filter has been the subject of a large body of research in both engineering and statistics. Unfortunately, much of this research has been done on linear Kalman filters, often under the assumption of Gaussian noise, conditional densities, and innovations. For example, maximum likelihood is commonly used in an off-line setting to estimate the hyperparameters in the system and process noise covariances $\mathbf{R}$ and $\mathbf{Q}$ [*Harvey*, 1989, p. 140]. In addition, the initial state of the filter is normally assumed known (e.g., see the initialization step for both filters). If unknown, a diffuse prior often is used by setting the initial covariance to $\psi\mathbf{I}$, where ψ is chosen to be suitably large such that $(\psi\mathbf{I})^{-1} \approx \mathbf{0}$ [p. 121 *Harvey*, 1989]. Other methods, such as the expectation-maximization (EM) algorithm have been developed to jointly estimate the initial states and the filter covariances [*Shumway and Stoffer*, 1982]. However, each of these maximum likelihood-based methods involves Gaussian assumptions, which *Hollinger and Richardson* [2005] have shown do not hold in the case of our flux measurements. In addition, it is doubtful that such methods directly adapt to nonlinear estimation problems, especially in the case of running simultaneous dual filters. A further complication arises when the covariance matrices are time-dependent, increasing the number of parameters that must be estimated.

[49] Acknowledging these difficulties, we have used a diffuse prior approach when initial conditions are unknown or inestimable, while other quantities have been estimated either through ML or Monte Carlo methods. Because of the seasonality in environmental conditions inherent at this latitude, it seems reasonable that some measurement and state variances will differ between the dormant and growing seasons, and this has been allowed for in our formulation. In all cases, the noises are assumed uncorrelated and diag[·] means construct a diagonal matrix.

[50] The filters are started with $k = 1$ beginning 1 January 2000 using the dormant season model (7) and (8). The initial conditions for the state filter use the diffuse prior approach with $\mathbf{P}_{\mathbf{x}_0} = 100\mathbf{I}$ and $\hat{\mathbf{x}}_0 = 0$. This latter value can be loosely interpreted as a reasonable value for the state on a winter night, but with low confidence. The parameter filter was initialized with values developed from process models fitted to the April and May 2000 data, since fits to dormant season data are not estimable for these models. In model (3), maximum likelihood estimates were used on the basis of the methods developed by *Hollinger and Richardson* [2005], while the associated variance estimates were determined from Monte Carlo runs as explained by these authors. The initial state and covariances are

$$\hat{\mathbf{w}}_0 = [0.5, \quad 386.9, \quad -25.0, \quad 4.9]$$
$$\mathbf{P}_{\mathbf{w}_0} = \mathrm{diag}[0.574, \quad 4662.6, \quad 17.1, \quad 0.322]$$

In addition, models (3) and (4) were coupled so that the initial parameters were estimated jointly by ML for the models as in the current filtering approach. In the state space formulation, R is a time-varying state; therefore the estimates for these initial conditions on the parameters can be assumed only to be some time-integrated average condition. The constant parameter $T_0 = 261.2°\mathrm{K}$ in (4) was also developed from a least squares fit of the model to the June 2000 data as mentioned earlier.

[51] The measurement model covariance was determined from the measurement record and from instrument precision. The identical measurement record was used for both filters; therefore $\mathbf{R}_{\mathbf{w}_k}$ and $\mathbf{R}_k$ are also identical. The measurement variances are

$$\mathbf{R}_k = \begin{cases} \mathrm{diag}[1, \quad 219, \quad 5], & \text{dormant season} \\ \mathrm{diag}[7.29, \quad 488, \quad 5], & \text{growing season} \end{cases}$$

Note that in the dormant season, the variability in both NEE and PPFD are less than during the growing season. This is a consequence of restricted respiration and photosynthetic activity coupled with shorter day length.

[52] The state filter requires covariances for the dormant and growing seasons in the $\mathbf{Q}_k$ matrix. For those states (PPFD and T) where process equations are assumed to be random walks, the corresponding elements in $\mathbf{Q}_k$ were set equivalent to those in $\mathbf{R}_k$. The growing season variance on NEE was taken as the overall variance from the model fitted to (3) as given by *Hollinger and Richardson* [2005], and was assumed to be half this value in the dormant season. In addition, the variance for R was the estimated model variance for the Lloyd and Taylor model, (4), fitted by *Richardson and Hollinger* [2005]. The process covariance matrices for both seasons are

$$\mathbf{Q}_k = \begin{cases} \text{diag}[7.37, \quad 219, \quad 5, \quad 10.0, \quad 0.15254], & \text{dormant} \\ \text{diag}[14.74, \quad 488, \quad 5, \quad 10.0, \quad 0.15254], & \text{growing} \end{cases}$$

The state variable INEE accumulates variance from all of its components; therefore it could justifiably be set to zero. However, because the magnitude of the spatial error in the integration is uncertain, we set it to a small, finite value on the order of the estimated variance for NEE.

[53] Lastly, we set $\mathbf{P}_{\mathbf{x}_{k|k-1}}$ and $\mathbf{P}_{\mathbf{w}_{k|k-1}}$ back to their initial conditions at the switch to the growing season in response to several separate idiosyncrasies in our model formulation. First, the process dynamics are different between dormant and growing seasons, as previously explained. Second, we constrain the variance on E_0 to be constant at half its initial value during the dormant season in order to keep this parameter tight and allow excess variation to go directly into R_p, while it remains unconstrained during the growing season. Thirdly, K remains unestimated (fixed) during the dormant season and as a result, the filter variance artificially inflates during this time period and must be reset. Finally, with the exception of A, the initial conditions for the parameter filter were developed on the basis of April and May data, and therefore any evolution to their variances over the dormant season must be discounted once the switch has been made. Inasmuch as the two filters are coupled, the state variances should be reset as well.

3. Results

[54] Figure 2 presents the time courses for three of the system states T, R, and INEE. One prominent feature of the INEE trajectory is the close association between our switching time with the ecosystem's actual switch from carbon source to sink. This is seen, with a slight lag period in the spring, as ecosystem photosynthesis begins to offset respiration. In the fall, it appears that respiration begins to dominate slightly before our switching point; however, because the soil is not frozen at this time, and some daily temperatures (Figure 2, top) do indeed rise above freezing for extended periods, the trajectory rises only gradually into December. The final filtered estimate for INEE was -296.45 g C m^{-2}, with estimated standard deviation of 2.44 g C m^{-2} (which is an estimate of random uncertainty as discussed later). This estimate of Howland forest annual NEE in 2000 is about 25 g C m^{-2} (9%) greater than previously published estimates [*Hollinger et al.*, 2004].

Table 1. Summary of Correlations Between Estimated Forward Filter States and Parameters for the 2000 Howland Data

	Parameters			
States	A	K[a]	E_0	R_p
	Dormant Season			
NEE	−0.024	. . .	0.062	0.042
PPFD	0.038	. . .	−0.161	0.212
T	0.028	. . .	0.054	0.372
INEE	0.362	. . .	−0.958	0.724
R	−0.434	. . .	0.423	0.387
	Growing Season			
NEE	−0.025	0.070	−0.014	0.065
PPFD	−0.065	−0.073	−0.122	0.108
T	−0.630	0.089	−0.528	0.659
INEE	0.706	−0.805	−0.041	−0.405
R	−0.775	0.145	−0.568	0.886

[a]The parameter K is unused during the dormant season.

The previous estimate relied on monthly models identical to equation (3) to fill in missing daytime data, and Fourier series to estimate missing nocturnal values. While difficult to judge from the scale of the figure, the confidence intervals on INEE increase as the year progresses because of the integral definition of this system state. Confidence intervals on the other system states remain fairly constant or increase during periods of missing values, as would be expected. Another apparent trend is the midsummer flattening of INEE trajectory in response to increasing ecosystem respiration and associated moderation of ecosystem photosynthesis (not shown).

[55] The time-varying parameter estimates from the forward filter are presented in Figure 3. There are several points worth noting in these trajectories. First, during the dormant season, A stays close to its initialization value of 0.5 μmol m^{-2} s^{-1}. At the switch, however, it quickly acclimates to its role in the big leaf model, and reaches a maximum (because of our sign convention) in August. A is negatively correlated with R throughout the year (Table 1). The correlation during the dormant season is due to the periods where photosynthesis occurs on warmer days as mentioned above. The strong negative correlation and associated mirror image of the two trajectories during the growing season clearly shows the seasonal relationship between A and R. This result probably stems from the well-known relationship between foliage photosynthetic capacity and foliage respiration [e.g., *Reich et al.*, 1998] and the postulated relationship between current photosynthesis and root respiration [*Hogberg et al.*, 2001]. Changes in the A parameter show gradual reestablishment of forest photosynthetic capacity in the springtime and a precipitous drop in this capacity following the first hard frost in the autumn. The plateauing of A and rapid increase of R_p in June appear in association with the production of new foliage.

[56] The K parameter (PPFD for half-maximum photosynthesis) decreases from our initialization value to a low of about 200 μmol m^{-2} s^{-1} in April, returns to about 400 μmol m^{-2} s^{-1} in May and then gradually rises through the autumn to about 800 μmol m^{-2} s^{-1} before dropping again in November. The increase in linearity in the PPFD:canopy photosynthesis relationship late in the season may be related

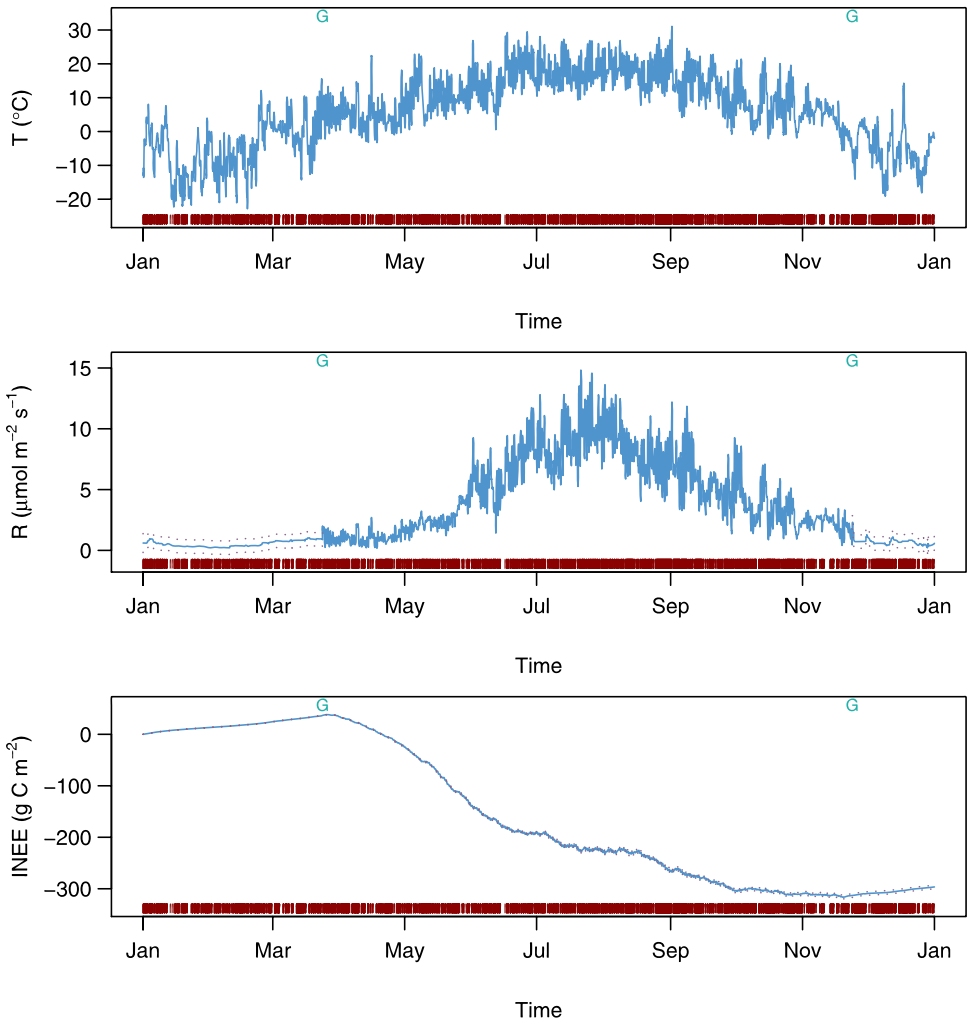


Figure 2. Kalman smoother estimates for the states T, R and INEE for the year 2000 data, with 95% confidence intervals (dashed) shown for the dormant season on R and the entire year on INEE. The vertical bars at the base of each graph show the missing values while the growing season is delineated at the top by letter G.

to the low sun angle and consequent lower PPFD intensities. Variation in leaf-level factors such as %N or canopy factors such as leaf area index could also contribute to the observed variation in the K parameter.

[57] The baseline respiration rate, R_p and ecosystem respiration are clearly highly correlated (Table 1) with very similar trajectories. This is expected from the construction of the respiration model where E_0 is a "shape factor" for the temperature response of respiration. The confidence interval widths at the switch between the dormant and growing season in March for R_p also illustrates the effect of resetting the initial conditions for the growing season at the switch. If the variances were not reset, it could allow R_p to shift more radically at this junction point. However, because we also relax the constraint on the variance of E_0 at this point, E_0 is allowed to vary freely as the full Lloyd-Taylor model comes online under the influence of temperature; these combined effects tend to keep R_p stable during this period, while allowing E_0 to absorb the variation in respiration at the onset of the growing season.

[58] It should be noted that there have been no constraints of any kind imposed upon our model at the switch from growing to dormant seasons in the fall. Yet all of the parameters seek their winter-spring dormant season values, bearing in mind the confidence of the estimates. Because the fall-winter dormant period is independent of the winter-spring season, our results suggest that the spring switch is the most sensitive of the two. Indeed, in numerous sensitivity runs, we found this to be true, resulting in the initial condition and variance switches described earlier.

[59] Figure 4 presents state trajectories for nine days in the month of August 2000. The two random walk models (PPFD and T) tend to track the measurements closely (note again that the measurement record for these states are complete). The exception is daytime PPFD on several of the more variable days (e.g., 7–9 August). Intermittent cloud cover will affect these measurements and the smoothed estimates tend to discount this higher observed variability; that is, they smooth the effect of passing clouds. This result is quite reasonable because the spatial average

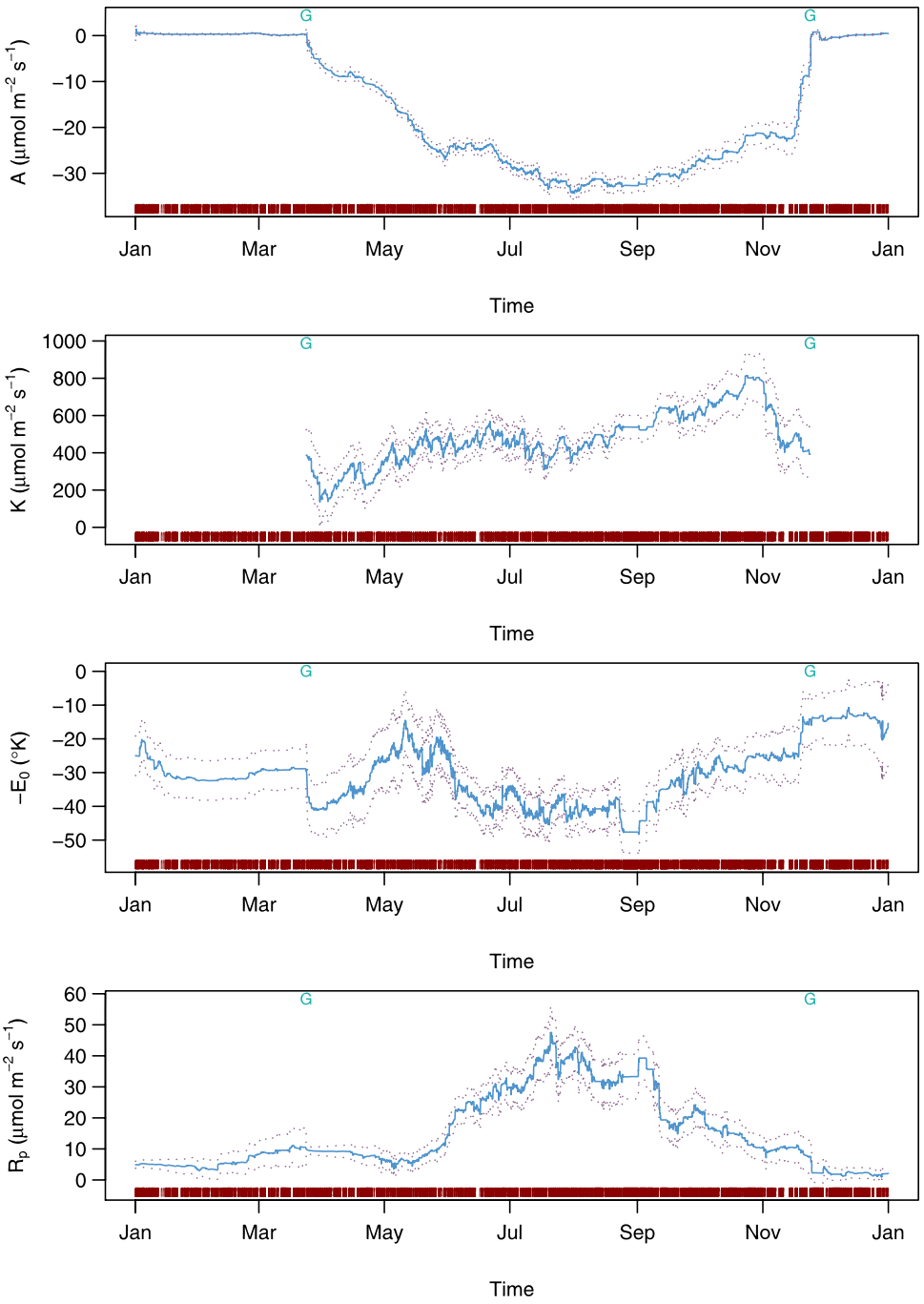


Figure 3. Kalman filter estimates for the four model parameters for the year 2000 data, with 95% confidence intervals (dashed) shown where appropriate. The vertical bars at the base of each graph show the missing values while the growing season is delineated at the top by letter G.

PPFD over the flux source (footprint) region will be less variable than the point measurement made at the flux tower. The estimates for respiration are clearly driven by temperature, the alignment shown is nearly perfect. Again, since T is the driving variable in our model, this result is also reasonable.

[60] Finally, the trajectory for NEE illustrates clearly the efficacy of the Kalman filter as a gap-filling method, even with our simple model. It is difficult to resolve the individual effects that PPFD and T have on the daytime NEE trajectory from the figures alone because of the correlation between PPFD and T (−0.54 for this period). However, correlations between NEE and PPFD for this period were −0.92, while those between NEE and T were −0.55, implying that PPFD is indeed the main driver of the system at this time of year under our model. However, where PPFD is zero at night, there is no photosynthesis and NEE is equivalent to nighttime respiration; therefore the sole driving variable at night is temperature. This effect can be discerned quite clearly on the night of 4 August where all NEE measurements are missing. Here, NEE peaks early in the twilight hours and decreases as T decreases into early

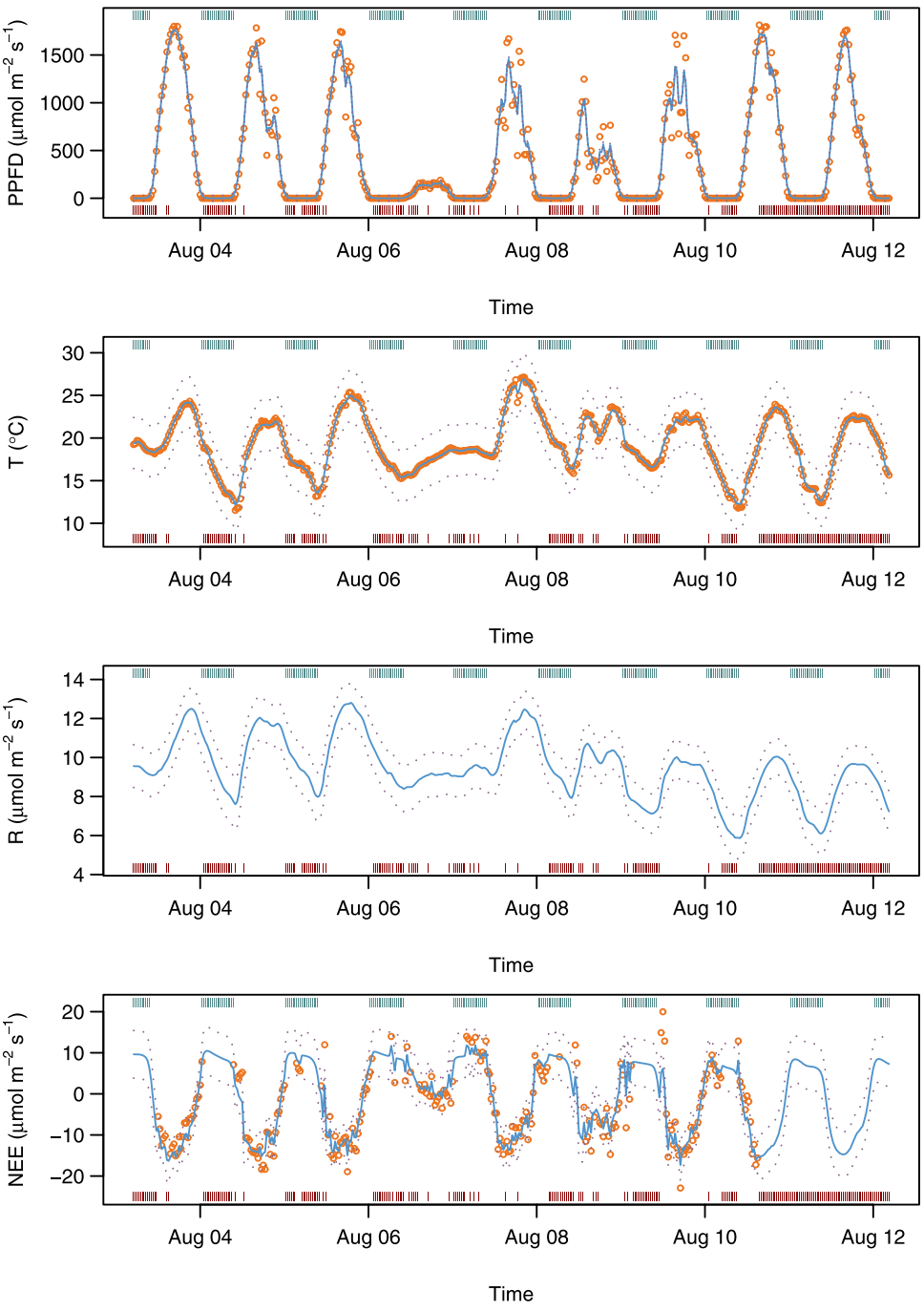


Figure 4. Kalman smoother estimates for the states PPFD, T, R and NEE for several days in August 2000, with 95% confidence intervals (dashed) and noisy observations (circles). The vertical bars at the base of each graph show the times where measurements of NEE are missing while those at the top delineate nighttime.

morning. On nights such as 5 August, where a few measurements interrupt the gaps, the smoother clearly adjusts from all model prediction to an optimal combination of model and data with corresponding downward adjustment in variance. In addition, the estimates for 7 August show how low light levels and cooler temperatures under heavy overcast, are accounted for in the estimates. Finally, there is a gap of several days at the end of this period (not all are shown) where the model predictions, driven by the complete measurement records for PPFD and T evidently do quite an adequate job at describing the trajectory for NEE during this period, when compared against similar days with complete data records. The conclusion to be drawn is that while our model is simple, it appears to be quite powerful in capturing a good degree of the system dynamics in relatively few state variables and associated few parameters.

[61] Incidentally, while the yearly trends in parameter estimates are interesting when viewed as a whole, for short

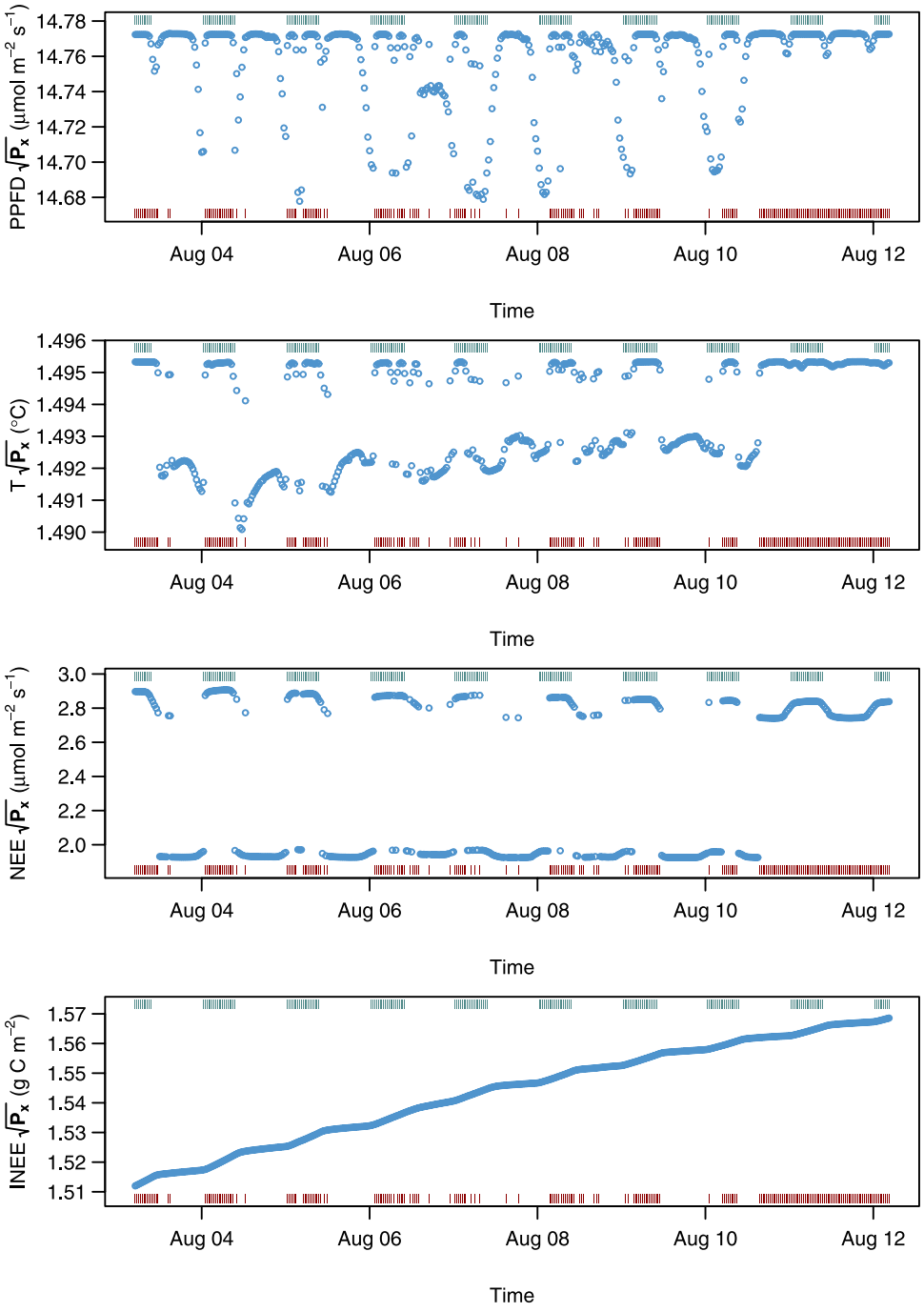


Figure 5. Kalman smoother standard deviation estimates for the states PPFD, T, NEE and INEE for several days in August 2000. The vertical bars at the base of each graph show the times where measurements of NEE are missing while those at the top delineate nighttime.

periods like our nine day illustration, they tend to be relatively constant and show only a mild degree of variation (see Figure 3). This is to be expected since the system itself is in a fairly steady state over such short periods. Large jumps in parameter values during such a period would be alerting to some exogenous inputs into the system that are not incorporated in the model.

[62] The confidence intervals on the yearly INEE trajectory and the 9-day PPFD trajectory are difficult to discern because the variances are small; their magnitude is shown in Figure 5 for the August example. Figure 5 clearly illustrates the increasing nature of the variance (as shown through the standard deviation) for INEE as mentioned earlier. In addition, it illustrates that there are essentially two distinct steady states that both the NEE and T variances tend to take, corresponding to whether NEE measurements are present or not; missing NEE values tend to raise the variances for these states to the higher level. In both cases, there is also a discernible difference in daytime and nighttime variance estimates for both NEE and T. PPFD is the exception, the

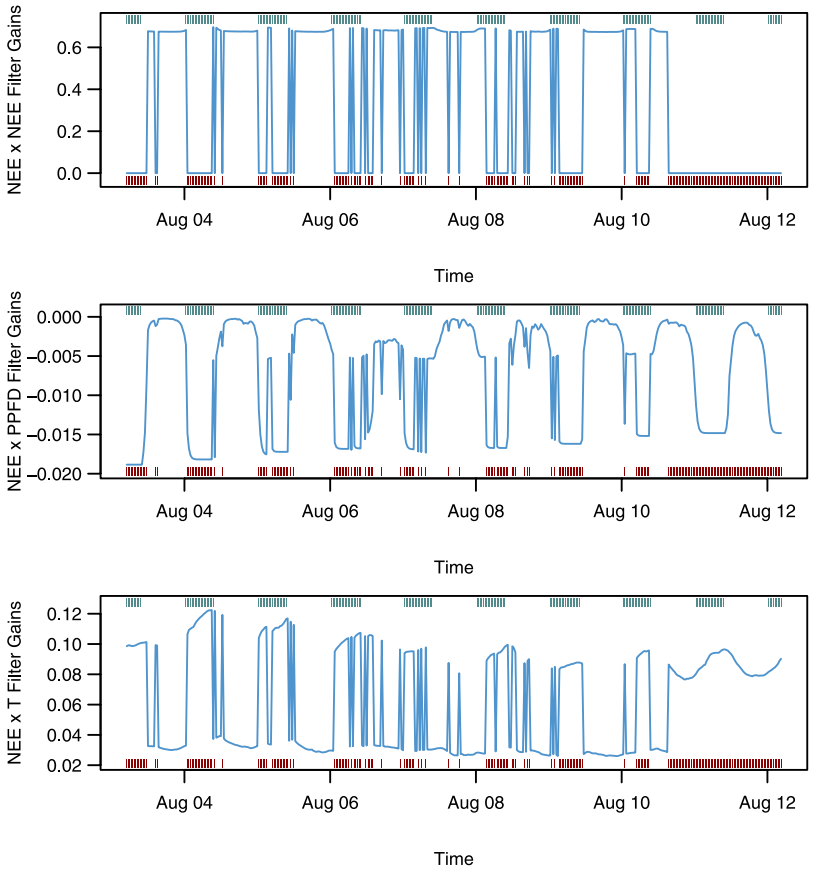


Figure 6. Forward Kalman gain estimates, $\mathbf{K}_k$, for the state NEE and measurements NEE, PPFD and T for several days in August 2000. The vertical bars at the base of each graph show the times where measurements of NEE are missing while those at the top delineate nighttime.

two levels seem to be missing and the variance values tend to fluctuate coincident with the steep downslope or upslope of the PPFD curve in the transition from day to night or vice versa. This may be at least partially an artifact of our setting all measurement values with PPFD < 5 μmol m^{-2} s^{-1} to zero in the preprocessing stage which affects the random walk model. Because the smooth estimates benefit from both forward and backward recursions over the data, this trend is established in both directions.

4. Sensitivity

[63] As mentioned earlier, one of the main assumptions leading to optimality of linear Kalman filters, or near optimality of nonlinear filters, is that the noise variances are known. We estimated these variances directly from the data. However, other methods for the estimation of these variances based on filter estimates have also been mentioned and include ML using the method of scoring on the prediction error decomposition form of the log-likelihood [*Harvey*, 1989, p. 140; *Gove and Houston*, 1996], and EM algorithms [*Shumway and Stoffer*, 1982], both of which make assumptions about the distribution of the error processes at a minimum. Because any discussion of sensitivity must be interpreted in light of the Kalman gains, which are influenced by these variances, we begin with a brief discussion of the role this matrix plays in the forward filter.

[64] The Kalman gain matrix, $\mathbf{K}_k$, is the optimal weight matrix for combining measurements and model predictions in the update step (21). It is composed of the cross covariance between the predictions (from resampled sigma points) and the measurements, $\mathbf{P}_{\mathbf{x}_k\mathbf{y}_k}$, weighted by the inverse of the measurement covariance $\mathbf{P}_{\mathbf{y}_k}$. Given our linear identity system transform, $\mathbf{h} = \mathbf{H}$, the gains are identically computed by (24). The gains can range from positive to negative depending on the cross covariances. Because of the structure of $\mathbf{H}$ in our formulation, the diagonal elements of $\mathbf{K}_k$ will be constrained to range between zero and one. When the gains are close to zero, preference is given to the model and the measurements are discounted; conversely, when the gains are near one, the measurements are preferred over the process model. In a simple univariate measurement system, this correspondence is exact; however, in a multivariate setting, the gain matrix has dimension $n_x \times n_y$ which means that even when there are missing values for one measurement variable, the gains will reflect, to some extent, the influence of the other measurements and state predictions in the resulting estimates. For example, when NEE is missing at time k, the first column of the gain matrix (as well as the residual for NEE) will be zero, but the rest of $\mathbf{K}_k$ will be nonzero. Therefore the update estimate, $\hat{\mathbf{x}}_k$, and its associated variance estimate, $\mathbf{P}_{\mathbf{x}_k}$, will be composed of the prediction step estimate plus a weighted linear combination of the other nonmissing innovations (or prediction covariances) at this time period. Similarly, negative gains act like

negative covariance terms, changing the sign of that innovation's contribution to the estimated state $\hat{\mathbf{x}}_k$ in (22).

[65] An example illustrating the influence of the gains on the estimates for NEE for our 9-day period in August can be seen in Figure 6. From Figure 6 it can be seen that when NEE is missing, the associated gains for PPFD and T are nonzero and therefore the prediction estimate for NEE will be adjusted by a weighted linear combination of these two innovation errors in the update step. When NEE is not missing, the NEE gains dominate those for PPFD and T so that most, but not all, of the updated estimate for NEE comes from its own innovation adjustment.

[66] Given this explanation, it should be clear that the Kalman update step is a highly integrated combination of information between states and measurements; in other words, there is a large degree of interaction between the variables of interest and one should not mistakenly infer that the states are estimated independently of one another. Because the Kalman filter propagates the joint density of the states through prediction and update steps, however, the cross link to the measurements through the gains can be easily overlooked.

[67] The influence of the Kalman gains are important because the state update estimates in the forward filter are the best estimates at any given time period in the forward recursions. A similar gain matrix also exists for the backward recursions, but not, per se for the smoothed estimates themselves. More importantly, the gains are influenced by the noise covariances $\mathbf{R}_k$ and $\mathbf{Q}_k$, and their affect on the gains can be assessed through sensitivity analysis in the process of filter tuning. The larger the number of states and measurements, the more difficult any sensitivity analysis is to undertake because of the interactions entering into the estimates that are partially manifest through the gains as just described. Any analysis must be undertaken conditionally, by modifying only one or two variance components at a time while holding the others constant. In addition, some criterion must be chosen for assessing the effects of the analysis. Because the final estimate of INEE and its variance are of major importance, we limit the discussion to the final forward filter estimate of these quantities.

[68] We ran numerous sensitivity trials in the tuning phase of the filter. In general, we noticed that the model we present is quite robust to changes in both initial conditions and individual (and sometimes joint) variance components. One interesting point to notice is that the standard deviation of our estimate for INEE is only 2.44 grams. Increasing the variance component for INEE in $\mathbf{Q}_k$ tenfold, produced no significant difference in either the state estimate or the estimated variance. Similarly, since it has been established that NEE (directly) and PPFD (via (3)) are the major contributors to INEE in our model, modifying their respective components of $\mathbf{R}_k$ and $\mathbf{Q}_k$ would be of interest. Increasing the measurement noise variance component for NEE in $\mathbf{R}_k$ tenfold during the dormant season and adding a fourfold increase in the measurement noise components for NEE during the growing season as well, resulted in no significant change to either the state or variance estimates for INEE. However, adding to these the same increases in the variance component for NEE in $\mathbf{R}_{\mathbf{w}_k}$ in the parameter filter reduced the overall state estimate of INEE by 24 grams and the associated estimated standard deviation by half a gram. These results suggests that the model is much less sensitive to increasing the measurement noise variance during the dormant season than during the growing season. In addition, it illustrates that the uncertainty of the estimate is quite robust to changes in NEE measurement noise variance. However, the estimate itself was decreased because of the tendency of the gains to weight more favorably in terms of the model, discounting the NEE measurements under the scenario of increased measurement noise forcing. Adding the increased variance to the parameter filter affected the growing season trajectory for E_0 while the other parameters remained largely unchanged. Similar results were found by increasing the process noise variance components for PPFD by 1000 in both the dormant and growing seasons.

[69] In order to observe any substantive change in estimated variance for the final INEE estimate, it was necessary to increase the measurement noise variance on PPFD fourfold for the growing season. This still only raised the final estimated standard deviation by 2 grams; however, it did also quite drastically increase the estimated INEE by close to 70 grams. Fortunately, this final scenario is so extreme, that it would be considered outside the possible variance domain for our model.

[70] One last point to bring out is the suggestion from the above runs that changing the variances in $\mathbf{R}_{\mathbf{w}_k}$ for the parameter filter exert a significant influence on the final estimate. However, this change in variance was in addition to parallel changes in $\mathbf{R}_k$ that had already been made. Adding these effects in a different order, it is easy to show that the change was really due to the increased variance component on NEE during the growing season for both filters. However, this raises the interesting question concerning the effects of the parameter variances on the final estimates when all initial conditions and variances are held to those in our original formulation. To help answer this question, $\mathbf{R}_{\mathbf{w}_k}$ was set equal to the identity matrix in a new filter run. The estimate for INEE was -295.94 grams with a standard deviation of 2.28 grams. All parameters effectively tracked the original estimates exactly during the dormant season and A, K and R_p were very close during the rest of the year. The main deviation was found in growing season estimates for E_0. Even then, however, the overall trajectory was very similar, just raised higher during the late spring and early summer and converging again in September. This exercise is important because it illustrates the robustness of the filter to changes in parameter measurement variance components. The parallel track taken by E_0 (with slight adjustments in R_p) could be due to equifinality, which has been observed in several other inverse modeling analyses of surface-atmosphere exchange [*Franks et al.*, 1997; *Schulz et al.*, 2001]. The main effect of decreasing the measurement variances in the parameter filter is to correspondingly decrease the estimated variances and subsequent confidence interval widths. Since the parameter filter is indeed a minimizer, this result makes a good deal of sense; it is discussed in greater detail by *van der Merwe* [2004].

5. Discussion

[71] In this study the dual approach to state and parameter estimation was used. However, as noted earlier, a related joint approach could also have been used. Theoretically, the

difference between the two is that the joint approach assumes explicit statistical dependence between states and parameters by allowing estimation of the cross state-parameter covariance terms, whereas in the dual approach these terms do not exist, effectively tantamount to assuming independence. However, even though the states and parameters were estimated in this quasi-independent manner under the dual filter (i.e., the cross covariances were implicitly zero), the correlations between states and parameters from our filter estimates (Table 1) show that the state-parameter switching at each time period does indeed introduce some implicit dependence. It would therefore be an interesting experiment, given this dependence, to see whether a joint modeling approach might be fruitful within our filter model formulation and whether the results obtained would differ significantly. This might be undertaken as part of a larger Monte Carlo simulation study. It is anticipated, however, that the estimate of INEE calculated via the joint approach would be similar to the present estimates.

[72] Two other Kalman-based methods have been developed recently that are very similar to the UKF. *Ito and Xiong* [2000] developed the central difference filter, while *Nørgaard et al.* [2000] independently developed the divided difference filter. Both filters use Sterling's interpolation methods to propagate the first two moments through the filter using a derivativeless deterministic sampling approach as in the UKF. Because the algorithms are so similar, these two filters can be considered the same for all practical purposes. In comparisons with the UKF, *van der Merwe* [2004] has found that they both yield very similar results.

[73] The literature on optimal estimation is vast, and numerous different filtering algorithms have been developed over the years to address different shortcomings to the Kalman filter when applied outside its original intent (i.e., linear propagation of the first two moments of the joint density). Methods for parameter estimation, nonlinearities, estimation of initial conditions, Bayesian extensions and many others have been developed; however, the basic set of recursions developed by Kalman still lay at the heart of most of these extensions. This is also true of two methods that have recently been applied to eddy flux data. In the first study, the state-dependent parameter (SDP) estimation approach [*Young*, 2000] was applied to eddy covariance data on an hourly time step [*Jarvis et al.*, 2004]. The SDP approach makes no a priori assumptions about the process models; rather it uses a multistep approach culminating in an empirical process formulation. In the SDP approach, the Kalman filter and smoother are run on nontemporally sorted data with time-varying parameters following a random walk model, facilitating simultaneous gap filling in this sorted state space. Final model identification was subsequently conducted from the final Kalman smoother estimates integrated over a daily time step to capture seasonal variation, producing site-specific empirical process models with constant parameter estimates.

[74] In a slightly more recent study, *Williams et al.* [2005] have employed a very powerful method known as the ensemble Kalman filter (EnKF), originally developed for high-dimensional modeling of ocean systems [*Evensen*, 2003]. The EnKF is a Markov Chain Monte Carlo approach that runs a parallel ensemble of filters together in time. The ensemble nature of this filter obviates the need to propagate forward high-dimensional covariance estimates as part of the filter algorithms, while also handling nonlinear dynamics without resorting to linearization. *Williams et al.* [2005] applied the EnKF to data that had been previously filled and integrated to a daily time step to accommodate their process models. Their application embedded the filter within a minimizer (sums of squared innovations) to estimate constant parameter values and initial conditions. *Harvey* [1989, p. 129] discusses the conditions under which such a criterion function is asymptotically equivalent to a maximum likelihood approach assuming linear Gaussian noise. This approach results in fixed initial conditions and constant parameter estimates for use in the process model dynamics in the filter.

[75] The SDP and EnKF approaches differ from our approach in several respects. First, we have made no assumption about the distributional properties of the noise terms, measurements, or states other than that the joint densities can be captured by their first two moments. Because Gaussianity may not be appropriate for our measurement data [*Hollinger and Richardson*, 2005], traditional Gaussian-based ML and EM approaches for hyperparameter and initial condition estimation, while robust, may not be entirely appropriate, especially with nonlinear system dynamics. Instead, we opted for estimating initial conditions and noise variances prior to filtering, while letting the process model parameter estimates evolve in time contemporaneously with the states, yielding a time-varying parameter trajectory that is correlated with the state estimates. This result is appealing for simple process models like ours, because the time-varying parameters may help to account for unmodeled system processes. This result is also in contrast to the other approaches, where the final parameter estimates are constants. Finally, in our approach gap-filling occurs coincident to estimation, whereas in the EnKF gaps were filled prior to estimation, or in the case of SDP estimation, while the data were sorted nontemporally.

[76] In light of the new applications of SDP and EnKF methods, eddy flux scientists now are beginning to have a number of choices for efficient data-model fusion methods for estimation of nonlinear systems that would include the UKF and its close variants. We have already suggested several more salient ways in which our approach could be improved, in the following we provide some discussion of the more subtle components of our model and analysis, with an eye toward future enhancements.

5.1. Time-Varying Parameters

[77] Several studies have recently explored the idea that carbon flux model parameters vary with time. We have already discussed the SDP approach taken by *Jarvis et al.* [2004], for example. In addition, *Raupach et al.* [2005], in discussing a joint filtering approach with state vector augmentation, note that allowing parameters to drift through time is a potential advantage of such techniques. However, such observations are not limited to filtering approaches. *Wang et al.* [2003] demonstrate how model parameters change seasonally and yearly using process models similar to ours, when fitted by traditional least squares. *Hollinger et al.* [2004] use ML to fit (3) and (4) to monthly 1996 eddy flux data from Howland. They observed monthly variation

in the parameter estimates with a high degree of correlation between A and K. These authors suggest that parameter variability may be due at least in part to the simplicity of the model, and conjectured that incorporating other factors might allow at least one of the parameters to stabilize.

[78] Our estimation approach yields time-varying model parameters (Figure 3), and while this appears to be useful in settings such as ours, there are still some unresolved questions. First, if parameter variability is due in part to overly simplistic models, one might propose using more complicated models that would include a more complete range of the state processes captured by the measurements. Indeed, one can imagine a complex model that almost completely accounts for ecosystem behavior (and would thus have at least some time-invariant parameters) that is far beyond our simple process model. Compared to other common models of ecosystem C exchange, we ignore temperature effects on photosynthesis, saturation deficit, soil water stress, ratio of direct to diffuse PPFD, seasonal phenological effects such as foliage production, and other factors. The influences of these other factors are, to some extent, swept into the four free parameters of the present model, and when their influence is strong, such as the impact of the first hard frost in the fall (i.e., several days prior to the switch), the effect on the model parameters can be severe. In addition, while more complicated process models can easily be implemented with the UKF approach, doing so introduces more states and parameters, increasing the dimensionality of the estimation problem. In such systems, correlations between parameters could become more of a problem as the limited information content in the flux measurements [*Schulz et al.*, 2001] does not change commensurate with the increased model dimensionality. Indeed the extent to which free parameters vary in a model-data fusion such as this could be diagnostic of the "completeness" of the model.

[79] Second, it is unclear at this time how the extramodel variation in the measurements is being partitioned between noise covariances and parameter variability in the dual approach. In the usual application of the Kalman filter to state estimation (parameters assumed known or possibly estimated by embedding the filter in likelihood iteration), the role of the noise covariance matrices are quite well understood in relation to their effects on the gains, and thus the optimal weighting of measurements and model predictions in the update step. For example, in univariate filters, the ratio of process to measurement noise variances is often used, e.g., to determine the rate of convergence to steady state conditions, while in multivariate settings it is the ratio of the eigenvalues of the noise covariance matrices [*Maybeck*, 1979, p. 224]. However, we noted in the sensitivity analysis to our approach that often fairly large (several-fold) changes could be made to components of the noise covariance matrices without changing the final state or variance estimates of INEE significantly. This can be corroborated by noting that increasing the process noise variances for the random walk components of the model tenfold, changed the final estimate of INEE by less than a gram, while the estimated variance remained virtually unchanged. This brings up the interesting, and unresolved question as to whether the ability of time-varying parameters in such systems to absorb extramodel noise in the observations mediates to some extent the classic roles of the noise covariance terms.

5.2. Switching Models

[80] The deterministic switch in our model can produce a fairly abrupt change in some estimated model parameter trajectories. This can be partially explained by noting that in the dormant season, NEE is composed mostly of respiration through R, but on warm days some photosynthesis can occur in A. While the simple dormant season models in (7) and (8) are certainly useful, they impart little knowledge of the process dynamics. At the winter-spring switching point, two related events happen. First, NEE is partitioned into R plus photosynthesis components, rather than being composed almost entirely of R. Second, a switch is made from models (7) and (8) to the process models for NEE and R. These two "shocks" to the system dynamics in a filter that has been in a fairly steady state, awaken it, as it were, to the growth that is beginning in the ecosystem. Thus any observed jumps in parameter trajectories are a manifestation of our models and the sharp switch between the seasons in spring. This same explanation can be used in the fall-winter season and applies to the drop in R_p that happens at the switch point. While R_p has been declining steadily with the decline in R, we effectively turn off its relationship to temperature and again force it to be equated to R and NEE. Simultaneously, E_0 is adjusted more heavily because of its larger variance, accommodating the adjustment required in R_p.

[81] In general, a hard deterministic switch is undoubtedly not the best method to use in the transition between dormant and growing seasons because in many places such transitions tend to be gradual. Other possibilities include precomputing running temperature averages or including soil temperature as a measurement and state to facilitate model refinements that would allow for multiple switch points during the transition period. In addition, the concepts used in stochastic switching models [*Hamilton*, 1993], might also be fruitfully employed.

5.3. Estimation Uncertainty

[82] The standard deviation in the UKF estimation of annual NEE (2.44 g m^{-2}) is less than 1% of estimated annual NEE. Compared to previous estimates of uncertainty in annual NEE [*Goulden et al.*, 1996b; *Falge et al.*, 2001a; *Griffis et al.*, 2003], this is a low value and warrants additional discussion. First, when NEE data are missing, the filter assumes that the model parameters are fixed, having no new information with which to update them, only adjusting parameters to new values when measured data again become available. With locally fixed parameters and only environmental data available, model uncertainty for these gaps will thus be low. On the basis of the observed parameter variability (Figure 3) this is reasonable for short gaps (a few days) but not over longer periods (weeks). In light of this, gaps are also more problematic in the beginning and end of the growing season when model parameters are changing rapidly.

[83] Second, our present results show that, while there is some change in the variance of the estimates for NEE where gaps occur, the variance does not increase substantially with time, even over many consecutive days of missing obser-

vations. This seems somewhat counterintuitive as one would expect there to be less certainty as we progress in time in the absence of measurements. However, it is well known that the linear Kalman filter covariances (both prediction and update) and gains, will reach a steady state in circumstances where the system matrices **F** and **H**, as well as the noise covariances **Q** and **R** are constant [*Gelb*, 1974, p. 142]. Indeed, under such conditions the steady state may be reached after only a few time steps [e.g., *Maybeck*, 1979, pp. 223–226]. This phenomenon occurs because the filter covariances and gains do not depend upon the measurements directly, even though the conditional means (state estimates) do. This is actually an appealing situation, since it allows filter designers to precompute the gain and filter variances offline, before any measurements are collected; these can then be used for optimizing filter design [*Maybeck*, 1979, p. 222]. In our model, **H**, **Q** and **R** are all constant matrices as in the linear case. In addition, our system model, **f**, is only mildly nonlinear. Since the UKF performs a statistical linearization at the current state, the mild nonlinearity evidently is discounted and the system becomes effectively linear.

[84] It can be seen from Figure 5 that the smoothed standard deviations for NEE are essentially constant at two levels corresponding to missing and nonmissing data (with slight diurnal variation). In addition, so are the corresponding NEE $\times$ NEE gains in Figure 6. Mathematically, the update variance contribution in our model for the nonmissing time periods is $(1 - k_{11})p_{11} - k_{12}p_{21} - k_{13}p_{31}$ (where the k_{ij} are elements of the gain matrix and the p_{ij} are from the prediction covariance matrix at some fixed time) and for those with missing data, $p_{11} - k_{12}p_{21} - k_{13}p_{31}$, since the gain is set to zero for NEE (i.e., $k_{11} = 0$). Thus, when the covariance matrices are in steady state (i.e., the p_{ij} and k_{ij} are constant), the variances simply fluctuate between these two different values depending upon whether NEE measurements are present or not. In more complicated models where the system or noise covariance matrices are allowed to vary in time, this interpretation would not necessarily hold.

[85] Lastly, flux data are also characterized by several systematic errors (biases) that are more difficult to quantify and were excluded from the present analysis. These errors include incorrectly specified spectral correction models to account for high- and low-frequency losses by real measurement systems using finite integration intervals, as well as possible errors in instrument calibration, data acquisition, and processing. Generally, it is believed that the largest systematic errors in CO_2 flux measurements are related to problems of insufficient nocturnal turbulence. The eddy covariance method depends upon scalar turbulent transport and if the nocturnal atmosphere becomes overly stable, transport between the ecosystem and overlying instrumentation can become temporarily uncoupled with the result that the measured flux is less than the true flux. Stable nocturnal conditions favor the development of katabatic flows that may carry CO_2 out of the ecosystem in surface flows that are never perceived by the flux instrumentation. To screen out artificially low nocturnal flux measurements, many researchers establish minimum turbulence criteria and only accept measurements if turbulent mixing exceeds the criteria threshold. *Goulden et al.* [1996a, 1996b] pioneered this approach and used a mean half-hourly friction velocity (u^*, the square root of the momentum flux) of 0.2 m s^{-1} as their threshold. The exact u^* cutoff is somewhat arbitrary and may depend upon the site. For the Howland data, *Hollinger et al.* [2004] used a u^* threshold of 0.25 m s^{-1} but also calculated NEE with u^* thresholds of 0.2 and 0.3 m s^{-1}. These values led to annual NEE estimates in 2000 that ranged between -287 and -251 g C m^{-2} y^{-1}. In the current analysis, we reran the filter with these same u^* thresholds of 0.2 and 0.3 m s^{-1}. The resulting estimates were -311 and -276 g C m^{-2} y^{-1} with estimated standard deviations of 2.4 and 2.5 grams, respectively. It is interesting to note that the range in both cases is almost identical.

[86] In addition, the u^* data are also noisy measurements; however, they have been used as true, uncorrected values without the benefit of filter estimation, as part of the prefiltering data step. With this in mind, the results of our analysis can be considered conditional on the u^* threshold chosen. Likewise, we note again that the results are also conditional on the values of the a priori estimates of unknown filter components such as the initial conditions and noise covariances. This considered, it is clear that the systematic NEE uncertainty is far larger than the random uncertainty derived from the data-model fusion. Therefore our resulting estimate of uncertainty should not be taken as reflecting the true uncertainty in ecosystem C exchange.

5.4. Gap Filling

[87] *Falge et al.* [2001a] highlight the need for consistent treatment of flux data, especially with regard to the modeling (filling) of gaps in the data record prior to synthesis efforts. Although several different methodologies were compared, including mean diurnal variation, look-up tables, and nonlinear regressions, no method was judged superior. Worryingly, annual NEE could differ by over 100 g C m^{-2} (30%), depending upon methodology. For each of these methods, the results of gap filling also depend upon an arbitrary time period. This may be a moving weeklong window for constructing a lookup table or monthly or seasonal periods for establishing nonlinear physiological model parameter values. In any case, altering the length, or beginning and ending dates of the time periods will change the gap filled values and hence the annual NEE estimate. Lack of consistency in choices will lead to variation in NEE calculated at different sites that has no ecological or environmental basis. Another potential problem is the recent finding of nonnormality in eddy flux measurement error [*Hollinger and Richardson*, 2005; *Richardson et al.*, 2006]. To correctly infer the parameters of models used in the gap filling process from flux data, the ML method should be used. If data errors are normally distributed with constant variance, least squares regression techniques yield ML parameter estimates; however, since flux data diverge from this standard, parameter values calculated by least squares methods are incorrect. These authors argue that flux data uncertainty is better represented by a double exponential probability density function and use the convenient ML estimation properties of this distribution to calculate flux model parameters [*Richardson and Hollinger*, 2005]. They show that filling in eddy flux gaps with models using parameters calculated in this manner can reduce estimated

respiration and consequently increases estimated INEE by 40 g C m^{-2} relative to least squares methods.

[88] We suggest that the UKF described here (or some similar variant) may be a suitable tool for providing consistency in filling gaps in eddy flux data time series. The UKF provides an approximate, close-to-optimal solution for the missing flux estimation problem (no nonlinear method can be truly optimal in general, because the optimal Bayesian solution requires propagating the full non-Gaussian joint probability density) and the autoregressive nature of the filter means that the prediction models are continuously updated, eliminating subjective decisions about time periods or dates that underlie other methods. The UKF is not limited to assumptions of normality in the data uncertainty. The filter requires information about the measurement and model uncertainty, which can be estimated in several ways for flux towers [e.g., *Hollinger and Richardson*, 2005] and by Monte Carlo methods for models [e.g., *Richardson and Hollinger*, 2005]. As presently formulated, the UKF also requires complete (gap free) time series of PPFD and T; however, alternative formulations may not impose this requirement. Once the work of coding the UKF has been completed, it can quickly run through data from a variety of years or sites.

[89] **Acknowledgment.** The authors would like to thank the three anonymous reviewers whose insightful suggestions greatly improved this manuscript.

References

Baldocchi, D. D. (2003), Assessing the eddy covariance technique for evaluating carbon dioxide exchange rates of ecosystems: Past, present, and future, *Global Change Biol.*, *9*(4), 479–492.

Bell, B. M., and F. W. Cathy (1993), The iterated Kalman filter update as a Gauss-Newton method, *IEEE Trans. Autom. Control*, *38*(2), 294–297.

Chen, Y., T. Huang, and Y. Rui (2002), Parametric contour tracking using unscented Kalman filter, paper presented at International Conference on Image Processing, Inst. of Electr. and Electron. Eng., Rochester, N. Y.

Crassidis, J. L., and F. L. Markley (2003), Unscented filtering for spacecraft attitude estimation, *J. Guidance Control Dyn.*, *26*(4), 536–542.

Davidson, E. A., A. D. Richardson, K. E. Savage, and D. Y. Hollinger (2006), A distinct seasonal pattern of the ratio of soil respiration to total ecosystem respiration in a spruce-dominated forest, *Global Change Biol.*, *12*, 230–239.

Evensen, G. (2003), The ensemble Kalman filter: Theoretical formulation and practical implementation, *Ocean Dyn.*, *53*, 343–367, doi:10.1007/s10236-003-0036-9.

Falge, E., et al. (2001a), Gap filling strategies for defensible annual sums of net ecosystem exchange, *Agric. For. Meteorol.*, *107*(1), 43–69.

Falge, E., et al. (2001b), Gap filling strategies for long term energy flux data sets, *Agric. For. Meteorol.*, *107*(1), 71–77.

Fitzgerald, R. J. (1971), Divergence of the Kalman filter, *IEEE Trans. Autom. Control*, *16*, 736–747.

Franks, S. W., K. J. Beven, P. F. Quinn, and I. R. Wright (1997), On the sensitivity of soil-vegetation-atmosphere transfer (SVAT) schemes: Equifinality and the problem of robust calibration, *Agric. For. Meteorol.*, *86*(1–2), 63–75.

Fraser, D. C., and J. E. Potter (1969), The optimum linear smoother as a combination of two optimum linear filters, *IEEE Trans. Autom. Control*, *7*(4), 387–390.

Gelb, A. E. (Ed.) (1974), *Applied Optimal Estimation*, MIT Press, Cambridge, Mass.

Goulden, M. L., J. W. Munger, S. M. Fan, B. C. Daube, and S. C. Wofsy (1996a), CO_2 exchange by a deciduous forest: Response to interannual climate variability, *Science*, *271*(5255), 1576–1578.

Goulden, M. L., J. W. Munger, S. M. Fan, B. C. Daube, and S. C. Wofsy (1996b), Measurements of carbon sequestration by long-term eddy covariance: Methods and a critical evaluation of accuracy, *Global Change Biol.*, *2*(3), 169–182.

Gove, J. H., and D. R. Houston (1996), Monitoring the growth of American beech affected by beech bark disease in Maine using the Kalman filter, *Environ. Ecol. Stat.*, *3*, 167–187.

Griffis, T. J., T. A. Black, K. Morgenstern, A. G. Barr, Z. Nesic, G. B. Drewitt, D. Gaumont-Guay, and J. H. McCaughey (2003), Ecophysiological controls on the carbon balances of three southern boreal forests, *Agric. For. Meteorol.*, *117*(1–2), 53–71.

Hamilton, J. D. (1993), Estimation, inference and forecasting of time series subject to changes in regime, in *Handbook of Statistics*, vol. 11, edited by G. S. Maddala, C. R. Rao, and H. D. Vinod, Elsevier, New York.

Harvey, A. C. (1989), *Forecasting, Structural Time Series Models and the Kalman Filter*, 1st ed., Cambridge Univ. Press, New York.

Hogberg, P., A. Nordgren, N. Buchmann, A. F. S. T. A. Ekblad, M. N. Hogberg, G. Nyberg, M. Ottosson-Lofvenius, and D. J. Read (2001), Large-scale forest girdling shows that current photosynthesis drives soil respiration, *Nature*, *6839*, 789–792.

Hollinger, D. Y., and A. D. Richardson (2005), Uncertainty in eddy covariance measurements and its application to physiological models, *Tree Physiol.*, *25*, 873–885.

Hollinger, D. Y., S. M. Goltz, E. A. Davidson, J. T. Lee, K. Tu, and H. T. Valentine (1999), Seasonal patterns and environmental control of carbon dioxide and water vapor exchange in an ecotonal boreal forest, *Global Change Biol.*, *5*(8), 891–902.

Hollinger, D. Y., et al. (2004), Spatial and temporal variability in forest-atmosphere CO_2 exchange, *Global Change Biol.*, *10*(10), 1689–1706.

Ito, K., and K. Xiong (2000), Gaussian filters for nonlinear filtering problems, *IEEE Trans. Autom. Control*, *45*(5), 910–927.

Jarvis, A. J., V. J. Stauch, K. Schulz, and P. Young (2004), The seasonal temperature dependency of photosynthesis and respiration in two deciduous forests, *Global Change Biol.*, *10*(6), 939–950.

Jazwinski, A. H. (1969), Adaptive filtering, *Automatica*, *5*, 475–485.

Jazwinski, A. H. (1970), *Stochastic Processes and Filtering Theory*, Elsevier, New York.

Julier, S. J., and J. K. Uhlmann (1997), A consistent, debiased method for converting between polar and cartesian coordinate systems, in *Proceedings SPIE AeroSense: Acquisition, Tracking and Pointing XI*, vol. 3086, pp. 110–121, Int. Soc. of Opt. Eng., Bellingham, Wash.

Julier, S. J., and J. K. Uhlmann (2002), The scaled unscented transformation, paper presented at American Control Conference, Am. Autom. Control Counc., Anchorage, Alaska.

Julier, S. J., and J. K. Uhlmann (2004), Unscented filtering and nonlinear estimation, *Proc. IEEE*, *92*(3), 410–422.

Kalman, R. E. (1960), A new approach to linear filtering and prediction problems, *J. Basic Eng., Ser. D*, *82*, 34–45.

Kitagawa, G. (1999), Processing of missing observations and outliers in time series, in *The Practice of Time Series Analysis*, edited by H. Akaike, pp. 353–365, Springer, New York.

Lewis, F. L. (1986), *Optimal Estimation With an Introduction to Stochastic Control*, 1st ed., John Wiley, Hoboken, N. J.

Lloyd, J., and J. A. Taylor (1994), On the temperature dependence of soil respiration, *Functional Ecol.*, *8*(3), 315–323.

Luo, Y., and J. F. Reynolds (1999), Validity of extrapolating field CO_2 experiments to predict carbon sequestration in natural ecosystems, *Ecology*, *80*(5), 1568–1583.

Massman, W. J., and X. Lee (2002), Eddy covariance flux corrections and uncertainties in long term studies of carbon and energy, *Agric. For. Meteorol.*, *113*(1–4), 121–144.

Maybeck, P. S. (1979), *Stochastic Models, Estimation, and Control*, vol. 1, *Mathematics in Science and Engineering*, vol. 141-1, Elsevier, New York.

Maybeck, P. S. (1982), *Stochastic Models, Estimation, and Control*, vol. 2, *Mathematics in Science and Engineering*, vol. 141-2, Elsevier, New York.

Nelson, A. T. (2000), Nonlinear estimation and modeling of noisy time-series by dual Kalman filtering methods, Ph.D. thesis, OGI Sch. of Sci. and Eng., Oreg. Health and Sci. Univ., Portland.

Nørgaard, M., N. Poulsen, and O. Ravn (2000), New developments in state estimation for nonlinear systems, *Automatica*, *36*(11), 1627–1638.

Plummer, E. S. (1995), Training neural networks using sequential-update forms of the extended Kalman filter, *Tech. Rep. LA-UR-95-422*, Los Alamos Natl. Lab., Los Alamos, N. M.

Raupach, M. R., P. J. Rayner, D. J. Barrett, R. S. Defries, M. Heimann, D. S. Ojima, S. Quegan, and C. C. Schmullius (2005), Model-data synthesis in terrestrial carbon observation: Methods, data requirements and data uncertainty specifications, *Global Change Biol.*, *11*, 378–397, doi:10.1111/j.1365-2486.2005.00917.x.

Reich, P. B., M. B. Walters, M. G. Tjoelker, D. Vanderklein, and C. Buschena (1998), Photosynthesis and respiration rates depend on leaf and root morphology and nitrogen concentration in nine boreal tree species differing in relative growth rate, *Functional Ecol.*, *12*, 395–405.

Reichstein, M., et al. (2003), Inverse modeling of seasonal drought effects on canopy CO_2/H_2O exchange in three Mediterranean ecosystems, *J. Geophys. Res.*, *108*(D23), 4726, doi:10.1029/2003JD003430.

Richardson, A. D., and D. Y. Hollinger (2005), Statistical modeling of ecosystem respiration using eddy covariance data: Maximum likelihood parameter estimation, and Monte Carlo simulation of model and parameter uncertainty, applied to three simple models, *Agric. For. Meteorol.*, *131*, 191–208.

Richardson, A. D., et al. (2006), A multi-site analysis of uncertainty in tower-based measurements of carbon and energy fluxes, *Agric. For. Meteorol.*, in press.

Saulson, B., and K. C. Chang (2004), Comparison of nonlinear estimation for ballistic missile tracking, *SPIE Opt. Eng. J.*, *43*(6), 1424–1438.

Schulz, K., A. Jarvis, K. Beven, and H. Soegaard (2001), The predictive uncertainty of land surface fluxes in response to increasing ambient carbon dioxide, *J. Clim.*, *14*(12), 2551–2562.

Shumway, R. H., and D. S. Stoffer (1982), An approach to time series smoothing and forecasting using the EM algorithm, *J. Time Ser. Anal.*, *3*(4), 253–264.

Shumway, R. H., and D. S. Stoffer (2000), *Time Series Analysis and its Applications*, Springer, New York.

Sitz, A., U. Schwarz, J. Kurths, and H. U. Voss (2002), Estimation of parameters and unobserved components for nonlinear systems from noisy time series, *Phys. Rev. E*, *66*, 016210, doi:10.1103/PhysRevE.66.016210.

Tenne, D., and T. Singh (2003), The higher order unscented filter, paper presented at American Control Conference, Am. Autom. Control Counc., Denver, Colo.

Teskey, R. O., D. W. Sheriff, D. Y. Hollinger, and R. B. Thomas (1995), External and internal factors regulating photosynthesis, in *Physiological Ecology of Coniferous Forests: A Contemporary Synthesis*, edited by W. K. Smith and T. M. Hinckley, pp. 105–140, Elsevier, New York.

Uhlmann, J. K. (1995), Dynamic map building and localization: New theoretical foundations, Ph.D. thesis, Dep. of Eng. Sci., Univ. of Oxford, Oxford, U. K.

van der Merwe, R. (2004), Sigma-point Kalman filters for probabilistic inference in dynamic state-space models, Ph.D. thesis, OGI Sch. of Sci. and Eng., Oreg. Health and Sci. Univ., Portland, Oreg.

van der Merwe, R., and E. A. Wan (2004), Sigma-point Kalman filters for integrated navigation, paper presented at 60th Annual Meeting, Inst. of Navig., Dayton, Ohio.

VanDyke, M. C., J. L. Schwartz, and C. D. Hall (2004), Unscented Kalman filtering for spacecraft attitude state and parameter estimation, in *Spaceflight Mechanics 2004: Proceedings of the 14th AAS/AIAA Space Flight Mechanics Meeting Held February 8–12, 2004, Maui, Hawaii*, vol. 119, edited by S. L. Coffey et al., pp. 217–228, Univelt, San Diego, Calif.

van Wijk, M. T., and W. Bouten (1999), Water and carbon fluxes above European coniferous forests modeled with artificial neural networks, *Ecol. Modell.*, *120*(2–3), 181–197.

van Wijk, M. T., and W. Bouten (2002), Simulating daily and half-hourly fluxes of forest carbon dioxide and water vapor exchange with a simple model of light and water use, *Ecosystems*, *5*(6), 597–610.

Wan, E. A., and A. T. Nelson (2001), Dual extended Kalman filter methods, in *Kalman Filtering and Neural Networks*, edited by S. Haykin, pp. 123–173, John Wiley, Hoboken, N. J.

Wan, E. A., and R. van der Merwe (2001), The unscented Kalman filter, in *Kalman Filtering and Neural Networks*, edited by S. Haykin, pp. 221–280, John Wiley, Hoboken, N. J.

Wang, Q., J. Tenhunen, E. Falge, C. Bernhofer, A. Granier, and T. Vesala (2003), Simulation and scaling of temporal variation in gross primary production for coniferous and deciduous temperate forests, *Global Change Biol.*, *10*, 37–51, doi:10.1046/j.1529-8817.2003.00716.x.

Williams, M., P. A. Schwarz, B. E. Law, J. Irvine, and M. R. Kurpius (2005), An improved analysis of forest carbon dynamics using data assimilation, *Global Change Biol.*, *11*(1), 89–105, doi:10.1111/j.1365-2486.2004.00891.x.

Young, P. C. (2000), Stochastic, dynamic modelling and signal processing: Time variable and state dependent parameter estimation, in *Nonlinear and Nonstationary Signal Processing*, edited by W. J. Fitzgerald et al., pp. 74–114, Cambridge Univ. Press, New York.

J. H. Gove and D. Y. Hollinger, Northeastern Research Station, USDA Forest Service, 271 Mast Road, P.O. Box 640, Durham, NH 03824, USA. (jgove@fs.fed.us; dhollinger@fs.fed.us)

Monte Carlo Methods

The general class of computational methods based on various forms of random sampling were called Monte Carlo methods, in honor of the famous principality whose raison d'être is gambling[1]. The method is called "deterministic" if the computation being performed is not inherently random: numerical quadrature schemes, based on evaluating a function at random abscissas; or solution of the Laplace equation with complicated boundary conditions; are deterministic problems. Conversely, if the problem has inherent random elements—for example, determining the distribution of particles that strike a screen after emerging from a magnetic spectrometer; or estimating the variance of the value of a stock portfolio in the face of market fluctuations—we call it "probabilistic".

We have already discussed numerical quadrature by Monte Carlo methods, as well as function minimization by simulated annealing. Both of these are deterministic problems. In this chapter we shall focus on problems for which the labor of simulating the underlying physics is no greater—and in some instances, much less—than that needed for an analytic solution. Our examples will be fairly simple—students who wish to delve further into this subject are directed to the classic work by Hammersley and Handscomb[2].

1. Pseudo-random numbers

The best contemporary discussion of methods for generating pseuo-random numbers on a digital computer is Knuth[3]. The liveliness of the subject is indicated by voluminous recent correspondence in computing journals[4].

We begin with a brief discussion of random numbers. A random sequence is a set of integers $\{a_k\}$ in the range [M, N] that exhibit no correlation from one to the next—in the sense that given that a_k has a certain value, the next number in the sequence, a_{k+1} is more likely to have one value than another (over the range of possible values). The absence of serial correlation can be generalized to an absence of multiple serial correlation: that is, instead of testing a_{k+1} we test a_{k+r} where $r > 1$. Such random sequences might be generated by perfectly fair roulette wheels, for example.

Of course, the techniques for generating "random" numbers on a digital computer are actually deterministic in nature. Thus given the same initial value a_0 , a function for generating "random" numbers will produce exactly the same sequence of numbers

1. When Von Neumann and Metropolis invented the method, Las Vegas, Nevada had not yet become the gambling Mecca it is today.
2. J.M. Hammersley and D.C. Handscomb, *Monte Carlo Methods* (Methuen & Co., Ltd., London, 1964).
3. Donald Knuth, Seminumerical Algorithms: The Art of Computer Programming, v. 2 (Addison-Wesley Publishing Co., reading, MA, 1981).
4. See, *e.g.* the letter by G. Marsaglia, *Commun. ACM* **36**, 7 (July 1993) 105-108, and various responses.

$$a_{n+1} = R(a_n)$$

every time such a sequence is generated. Moreover, by their very nature, even the least correlated random number generating functions will exhibit some degree of serial correlation. This is why such functions are called *pseudo-random number generators* (`prng`'s) in the literature of computation.

Another point to be aware of is that on computers whose word-size is 32 bits, many `prng`'s in common use produce integers in the range $[1, 2^{31} - 1] \equiv [1, 2147483647]$ so that after about 2×10^9 calls to the function the sequence will repeat itself. This may affect the simulation adversely[5].

The standard algorithm for generating prn's is the *linear congruential method* first proposed by Lehmer[6]. This is based on

$$x_{k+1} = (a\,x_k + c) \mod M\ .$$

With proper choices of a, c and M this can be a very good generator. Poor choices will lead to short cycles and lots of correlation. One good choice is $a = 16807$, $c = 0$ and $M = 2^{31} - 1$. There are other choices that are as good or better, but somewhat harder to implement.

Here is a program in Forth that implements this algorithm and produces a uniformly distributed floating point number in the range (0, 1) :

```
\ Pseudo random number generator in ANS Forth
\
\       Leaves a pseudo random number in the range (0,1)
\           on fp stack.
\
\ -------------------------------------------------
\     (c) Copyright 1998  Julian V. Noble.          \
\       Permission is granted by the author to      \
\       use this software for any application pro-  \
\       vided this copyright notice is preserved.   \
\ -------------------------------------------------
\
\   Based on GGUBS algorithm:  s' = 16807*s mod (2^31-1)
\   P. Bratley, B.L. Fox and L.E. Schrage, A guide to simulation
\   (Springer, Berlin, 1983).
\
\   To simplify transport to 16-bit machines the 32-bit
\   modular division is performed by synthetic division:
\
\   Let  b = d * m1 + m2 (= 2^31 - 1)
\
\   so that ( [n] means "largest integer <= n" )
\
```

5. For example, a system-supplied `prng` I once used had a cycle length of only 2^{15}-1= 32767, which led to very unreliable results.
6. D.H. Lehmer, *Ann. Comp. Lab. Harvard Univ.* **26** (1951) 141-146; M. Greenberger, *Journal of the ACM* **8** (1961) 163-167.

```
\       s' = (s*m1) MOD b = s*m1 - [s*m1/b]*b
\                         = m1 * (s - [s/d]*d) - m2 * [s/d]
\
\   Environmental dependence:
\
\       1. assumes at least 32-bit DOUBLEs
\       2. needs FLOATING and DOUBLE wordsets
\       3. assumes separate floating point stack
\

\ MARKER -rand

2VARIABLE     seed

21474.83647   D>F   FCONSTANT  bigdiv            \ 2^31-1
1277.73       D>F   FCONSTANT  divis
16807.        D>F   FCONSTANT  m1
2836.         D>F   FCONSTANT  m2

: (rand)    ( adr --)   ( f: -- seed')
         DUP  2@  D>F  divis FOVER FOVER        ( f: s d s d)
         F/  F>D  2DUP   D>F        ( [s/d])    ( f: s d [s/d])
         F*   F-                    ( f: s-d*[s/d])
         m1   F*                    ( f: m1*[s mod d])
         D>F  m2  F*  F-  FDUP  F>D  ( adr d)  ROT 2! ;

: prng      ( f: -- random#)
    seed  (rand)  bigdiv            ( f: -- seed 2**31-1)
    FSWAP  FDUP  F0<        ( -- f) ( f: -- 2**31-1 seed)
    IF   FOVER  F+   THEN   FSWAP  F/  ;

: test  0.1  seed 2!  1000 0 DO  prng  FDROP  LOOP  seed 2@ D. ;
\ TEST 522329230  ok
```

The reader should convince himself of the identities

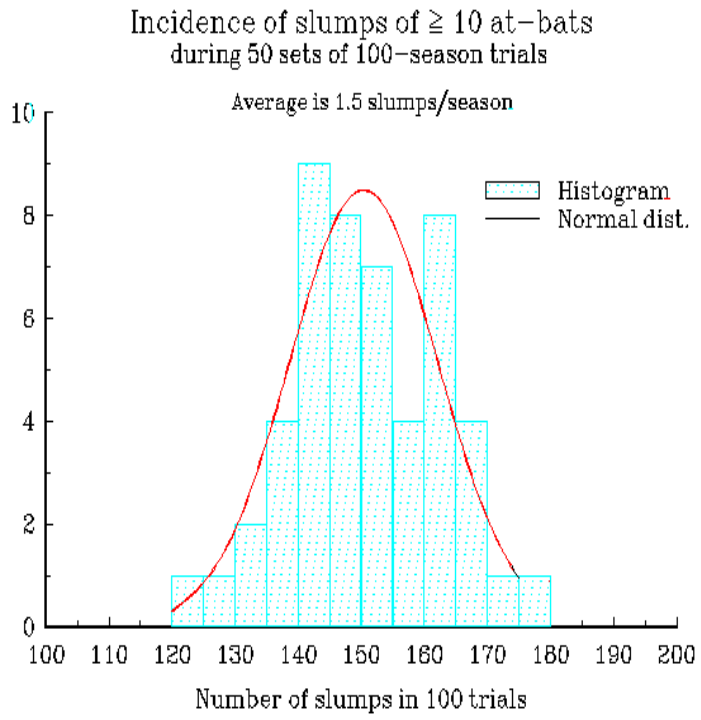
$$s' = (s \cdot m_1) \bmod b \equiv m_1 \cdot (s \bmod d) - m_2 \cdot \left[\frac{s}{d}\right]$$

$$\equiv m_1 \cdot \left(s - \left[\frac{s}{d}\right] \cdot d\right) - m_2 \cdot \left[\frac{s}{d}\right] .$$

where the square brackets mean "largest integer less than", *i.e.* round fractions down.

2. Batting practice

Having constructed a `prng` we are now in a position to apply it to some simulations. Consider a baseball player who plays N games per season. Assume his batting average is 0.300 (it could be anything.) If his chance of getting a hit, whenever he is at bat, is independent of any other time at bat, chance will lead to strings of hits or strikeouts—that is, to streaks or slumps. We night like to know what is the incidence of batting slumps—defined as 10 or more times at bat without a hit—for such a batter. This is a rather difficult problem to analyze using probability theory, but it is perfectly straightforward to simulate.



Basically, each time at bat is represented by generating a random number ξ. If ξ is between 0.0 and 0.3, a hit is recorded. If ξ is greater than 0.3, the trial is recorded as a strikeout. Next we need to check the strikeout counter to see if it has exceeded 10 strikeouts without an intervening hit. If so the event is recorded as a slump. The results from this simulation are shown to the left. The program that generated them is given below:

```
\ Simulation of batting slumps
\
\ ------------------------------------------------
\     (c) Copyright 1999  Julian V. Noble.          \
\       Permission is granted by the author to      \
\       use this software for any application pro-  \
\       vided this copyright notice is preserved.   \
\ ------------------------------------------------
\
\ This is an ANS Forth program requiring the
\   FLOAT, FLOAT EXT, FILE and TOOLS EXT wordsets.
\
\ Environmental dependencies:
\       Assumes independent floating point stack
```

```
MARKER -slump

INCLUDE prng.f      \ load the random number generator

\ ------------------------------------------- data structures
0.3 FCONSTANT  p0   \ batting average
250 VALUE N         \ # times @ bat/season

0 VALUE slump_len   \ place to hold the current

10 VALUE max_len    \ a slump is at least max_len strikeouts

0 VALUE #slumps

: initialize   ( 0.1  seed 2!)
               0 TO slump_len
               0 TO #slumps   ;

: got_a_hit?      ( -- f)
    prng  ( f: -- xi)   p0  F<=   ; \ its a hit if xi <= p0

: is_slump?   ( -- f)
    max_len  slump_len  <  ;

: end_slump!   is_slump?        \ was there a slump going?
          IF   #slumps  1+  TO #slumps   THEN
          0 TO slump_len  ;

: measure   ( -- )
    N 0  DO     got_a_hit?
                IF     end_slump!
                ELSE   slump_len  1+  TO  slump_len  THEN
    LOOP  ;

: stats     ( ncases --)
    initialize  DUP
    0 DO    measure   LOOP
    CR           . ."  trials of "        \ print results
        N        . ."  at-bats, giving "
        #slumps  . ."  slumps."
    ;
```

3. Non-uniformly distributed random variates

It often happens that we wish to simulate a process for which the random events are not representible by uniformly distributed real numbers in the interval (0, 1). For example we might wish to represent normally distributed random variables with mean $\bar{x}$ and standard deviation σ. There are several well-known algorithms for this. The Box-Muller algorithm uses the fact that

$$\int dx \int dy\, e^{-(x^2+y^2)/2} = \int d\theta \int d\left(e^{-\rho^2/2}\right)$$

Hence if we choose θ uniformly distributed on $[0, 2\pi]$ and $\eta = e^{-\rho^2/2}$ uniformly distributed on $(0, 1)$ we find

$$a = \sqrt{-2 \ln \eta}\ \cos\theta$$

and

$$b = \sqrt{-2 \ln \eta}\ \sin\theta$$

to be normally distributed random variables.

The problem with the algorithm in this form is that it requires that we evaluate three transcendental functions (`ln`, `sin` and `cos`) and a square root. However, we note that when x and y are independent random variables, uniformly distributed on the interval $(-1, 1)$, the random variable

$$u = \left(x^2 + y^2\right)\theta(1 - x^2 + y^2)$$

is a random variable that is uniformly distributed on $(0, 1)$. This allows us to implement the following procedure:

- pick two random numbers using the uniform `prng`;
- transform them to the interval $(-1, 1)$;
- compute u as above, rejecting it and starting over if it exceeds unity;
- compute the two random variables a and b given above, using the fact that

  $$a = x\sqrt{(-2 \ln u)/u} \qquad \text{and} \qquad b = y\sqrt{(-2 \ln u)/u}$$

  are independent normally distributed random numbers (on the interval $(-\infty, \infty)$) with mean 0 and standard deviation $\sigma = 1$.

Note that if we compute both numbers when the subroutine is called, we can save one of them so that we need only evaluate a logarithm and a square root every other time.

A standard FORTRAN function for the Box-Muller transformation is[7]

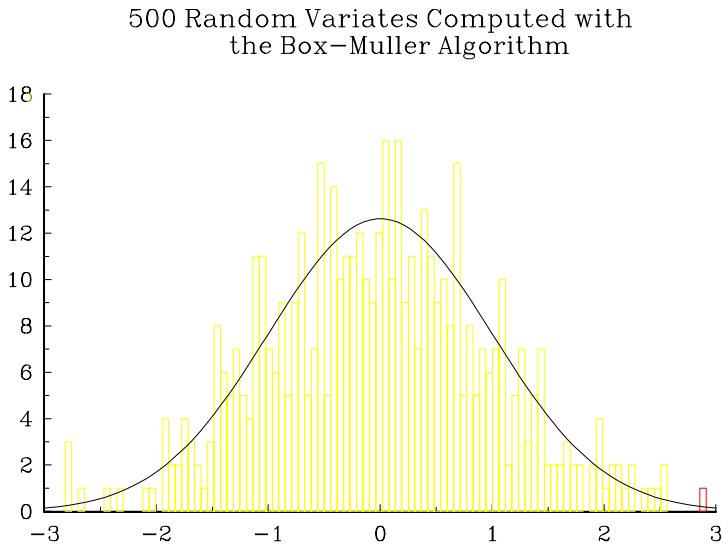
```
      FUNCTION GASDEV(IDUM)
      DATA ISET/0/
      IF (ISET.EQ.0) THEN
1       V1=2.*RAN1(IDUM)-1.
        V2=2.*RAN1(IDUM)-1.
        R=V1**2+V2**2
        IF(R.GE.1.)GO TO 1
        FAC=SQRT(-2.*LOG(R)/R)
        GSET=V1*FAC
        GASDEV=V2*FAC
        ISET=1
      ELSE
        GASDEV=GSET
        ISET=0
      ENDIF
      RETURN
      END
```

A Forth version appears below, to illustrate the difference in style between the two languages.

7. Press, *et al.*, *Numerical Recipes* (Cambridge University Press, New York, 1986).

```
\ Box-Muller algorithm for normal deviates
\ with mean 0 and sigma = 1.

\ This is an ANS Forth program requiring the
\   CORE EXT, FLOAT, FLOAT EXT, FILE and TOOLS EXT wordsets.
\
\ Environmental dependencies:
\       Assumes independent floating point stack

\ ---------------------- conditional compilation of non-Standard words
: undefined    BL WORD  FIND  NIP  0=  ;
undefined F2*       [IF]  : F2*    2.e0  F*  ;          [THEN]
undefined f^2       [IF]  : f^2   FDUP  F*   ;          [THEN]
undefined prng      [IF]  include  prng.f               [THEN]

\ -------------------- data structures
FVARIABLE   xi
0 VALUE  use_last

: get_x   prng  f2*  1e0  f-   ( f: -- x)  ;

: u_test    ( -- f) ( f: x y -- x y u)    \ true if u > 1
    FOVER  F^2  FOVER  F^2  F+  FDUP  1e0  F>  ;

: box_mul   ( f: -- a)
    use_last  IF   xi F@    FALSE  TO use_last   EXIT   THEN
        BEGIN   get_x  get_x   u_test
        WHILE   FDROP  FDROP  FDROP     \ if u > 1 start over
        REPEAT
    FDUP   FLN  FSWAP  F/   F2*         ( f: x y  -2*ln u / u )
    FABS   FSQRT                        ( f: x y sqrt[-2*ln u /u])
    FDUP   FROT  F*  xi  F!   F*   ;    ( f: -- a)
```

Output from this program is shown below in histogram form—the normal distribution corresponding to the number of samples is also shown.



Next let us consider how we would sample from an arbitrary probability distribution. We note that if the cumulative distributions for two sets of random variables are

$$P(x) = \int_{-\infty}^{x} dx' \, p(x')$$

$$Q(\xi) = \int_{-\infty}^{\xi} d\xi' \, q(\xi')$$

and we set

$$P(x) = Q(\xi)$$

then our problem is to find the function $x(\xi)$. If we suppose this is a known function and differentiate both sides with respect to ξ we find

$$p(x) \frac{dx}{d\xi} = q(\xi) \, .$$

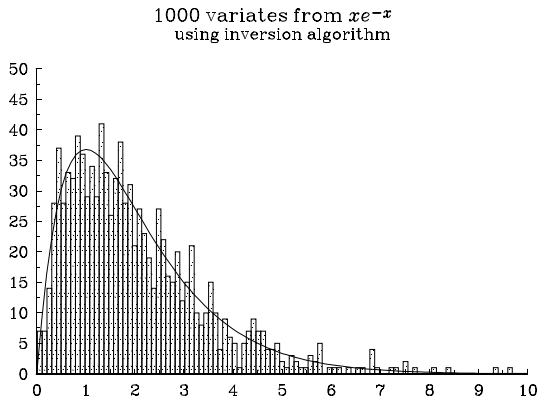
In general, since it is easy to generate uniformly distributed random variables we set

$$q(\xi) = \theta(\xi)\, \theta(1 - \xi)$$

and invert the relation

$$\int_{-\infty}^{x(\xi)} dx' \, p(x') = \xi \, .$$

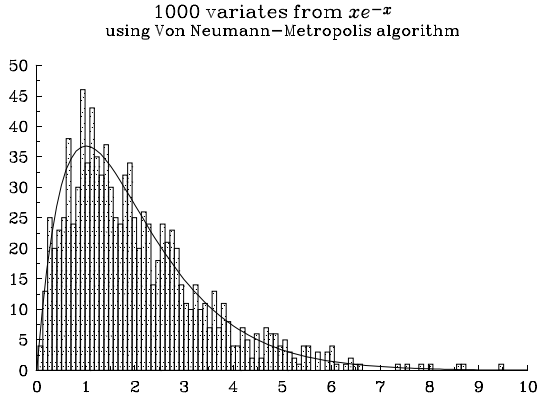
For some distributions this is difficult or impossible to accomplish in closed form, hence choosing random variates requires solving a transcendental equation. The figure to the right is a histogram of 1,000 variates chosen in this way from the distribution $p(x) = xe^{-x}$.



Alternatively, one may sample from the distribution $p(x)$ using the rejection method of Von Neumann: choose a random variable η from a uniform `prng` and also (uniformly) choose a value of x in the domain of $p(x)$ by scaling from $(0, 1)$. Then if

$$\eta \cdot \max p \leq p(x)$$

we keep the point, otherwise we reject it. A histogram of 1,000 such variates is shown to the right.



4. Young's two-slit experiment

Thomas Young is known for many contributions to physics but principally for having demonstrated the interference of light in 1800. He allowed coherent light to fall upon parallel slits in an opaque sheet, producing interference fringes on a screen. As is well known, the two-slit interference pattern is approximated by the intensity distribution

$$I(x) \propto 4\cos^2\left(\frac{a\pi x}{L\lambda}\right)$$

where a is the separation between the slits, L the distance to the viewing screen, and x the distance along the screen measured from the central maximum.

Young's result stood for more than a century as proof that light consisted of waves; in 1905, however, Albert Einstein introduced the idea that light consisted of energy quanta. Sir J.J. Thomson immediately recognized that this would imply the statistical nature of the preceding intensity distribution[8]. That is, one would expect the diffraction pattern to build up from individual photons striking the screen, the minima being places where photons had zero probability of arriving. One of Thomson's assistants, G.I. Taylor, thence attempted to photograph diffraction patterns in exceedingly dim light, hoping to see individual photons[9]. This proved impossible, however, with the available technology so the (negative) outcome led only to an upper bound on Planck's constant (which was, of course, much greater than the value predicted either by Einstein's photoelectric formula or by the blackbody radiation spectrum).

To simulate the buildup of the diffraction pattern on the screen, we choose points whose distribution function is

$$p(x) = \frac{\cos^2 x}{\int_{-X}^{X} du\, \cos^2 u}$$

on the rectangle

$$x \in (-X, X)$$

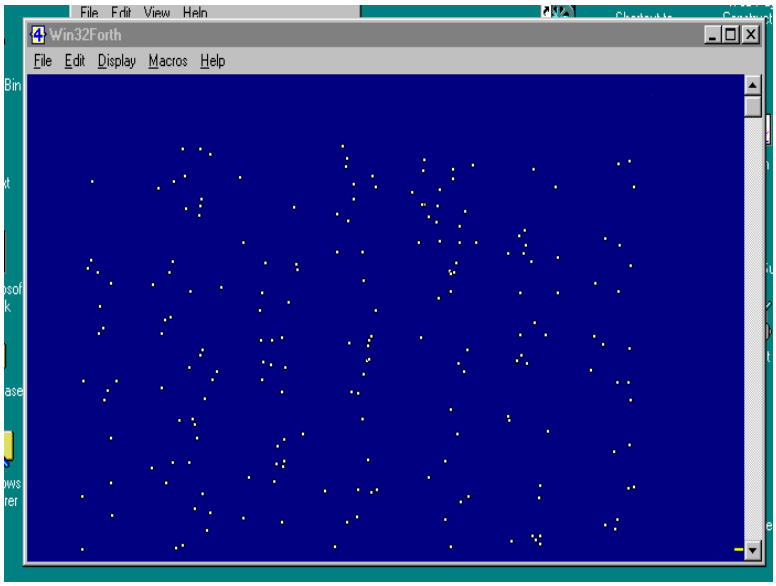
$$y \in (-Y, Y) .$$

A Forth program that does this by inverting the distribution (using the equation solver from Chapter 1) is shown on the next page, together with a screen shot of the output.

8. J.J. Thomson, *Proc. Camb. Phil. Soc.* **xiv** (1907) 417.
9. G.I. Taylor, *Proc. Camb. Phil. Soc.* **xvi** (1909) 415.

```
\ Simulation of Young 2-slit experiment with
\   individual photons
\
\ ************************************************
\     (c) Copyright 1998  Julian V. Noble.          *
\       Permission is granted by the author to      *
\       use this software for any application pro-  *
\       vided this copyright notice is preserved.   *
\ ************************************************
\
\ This is an ANS Forth program requiring the
\   FLOAT, FLOAT EXT, FILE and TOOLS EXT wordsets.
\
\ Environmental dependencies:
\       Assumes independent floating point stack
\       Assumes non-Standard graphics words from Win32Forth

MARKER -2slit
include prng.f
include ansfalsi.f
    0.1 seed 2!

    10e FCONSTANT x1
     5e FCONSTANT y1

   300e FCONSTANT y_lim
   500e FCONSTANT x_lim

    x1 F2*  FSIN  F2/  x1 F+  FCONSTANT  c

    FVARIABLE zeta

\ non-Standard Win32Forth graphics words
    WinDC theDC

: set_plot  ( – )
    CONDC PutHandle: theDC          \ initialize DC to the console
    0 0 800 600 BLUE  FillArea: theDC
    WHITE           BrushColor: theDC   ;

: offset    ( x y n – x+n y+n)
     TUCK  +  R  +  R  ;

: scale   ( f: x y – )  ( – x' y')
    y1  F/  1e  F-  F2/   y_lim  F*   FNEGATE   F>S
    x1  F/  1e  F+  F2/   x_lim  F*             F>S  SWAP  ;

: fplot    ( f: x y –)     \ plot a point
    scale  50 offset  2     FillCircle: theDC ;

\ continued on next page
```

```
\ Simulation
: new_zeta   prng  F2*  1e0  F-   c  F*  zeta  F!  ;

: new_y      prng  F2*  1e0  F-   y1 F*  ;

: f1    ( f: x - -  f1 = x+sin[2x]/2 - zeta )
        FDUP   F2*  FSIN  F2/  F+   zeta  F@  F-  ;

: new_x     ( f: - - x)      \ invert cos^2 distribution
            new_zeta            use( f1 x1 fnegate x1 1.0e-4 )falsi   ;

: 2slits    ( Npoints –)
        0.1 seed 2!                 \ initialize prng
        500 500 gotoxy              \ move cursor away
        set_plot                    \ make background
        0 DO
            new_x  new_y  fplot     \ plot a point
        LOOP
        BEGIN  KEY?  UNTIL          \ loop until key pressed
        KEY DROP    CLS  ;          \ clean up
```



Graphical output from 2slit.f -- 200 photons

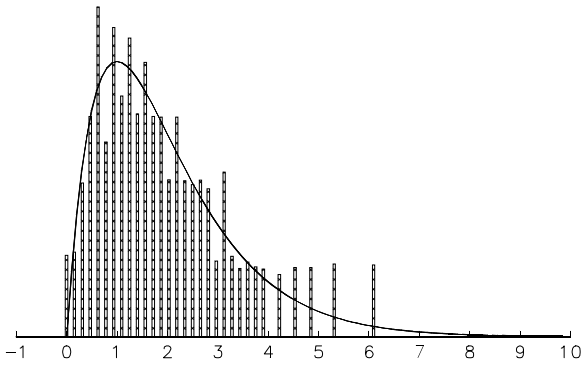
5. Random look-up tables

In our example above, the bottleneck was solving the transcendental equation for the random variables distributed according to the $\cos^2 x$ law. It is easy to see that if the integral representing the cumulative distribution cannot be expressed in closed form, in terms of well-known functions, the time to compute each point will become unacceptably large. Under such conditions even the Von Neumann rejection algorithm may be too slow.

There exists a remedy for this problem, based on the observation that for many simulations a histogram representation of the non-uniform distribution is perfectly adequate. Depending on the available storage, the histogram may be made sufficiently fine-grained that no loss of "physical significance" will be incurred. To construct such a representation it is enough to create a table of N variates by solving the equation

$$\int_{-\infty}^{x_k} dx\, p(x) = \frac{k}{N}, \quad k = 0, \ldots, N-1$$

Now, in what sense does such a table represent the desired random process? If we sample it using a pseudo-random number generator to generate random integers in the range [0, N–1], the resulting variates will generate a histogram that approximates the original distribution. An example of this is shown to the right, for the distribution $p(x) = x\,e^{-x}$, using a table of only 64 elements. If one employs a uniform random number generator with cycle length K, the table will be traversed in approximately N/K different orders, before repeating. This should be adequate for most uses.



The advantage of the random look-up table is its speed. All the computation is performed when the table is generated. Thus the run-time speed is simply the time to generate a random integer and fetch the appropriate variate from the table. In languages like FORTRAN or C we must create each such table and invoke it "by hand".

One advantage of Forth and other extensible languages[10] such as Lisp or Scheme, in this context, is that they permit us to define a generic constructor—a subroutine that creates packaged subroutines—of random lookup tables. Thus a large simulation that needs several such tables, each with its own distribution function and random number generator, can be much briefer to program. In the Forth program on the next pages, the subroutine `)DIST:` is the generic constructor.

10. "Object-oriented" languages like C++ and Smalltalk are similarly advantageous.

```
\ Forth tools for random lookup tables
\
\ ------------------------------------------------
\     (c) Copyright 1998  Julian V. Noble.          \
\       Permission is granted by the author to      \
\       use this software for any application pro-  \
\       vided this copyright notice is preserved.   \
\ ------------------------------------------------
\
\ This is an ANS Forth program requiring the
\   FLOAT, FLOAT EXT, FILE and TOOLS EXT wordsets.
\
\ Environmental dependencies:
\       Assumes independent floating point stack

FALSE [IF]

 Algorithm:
    Random lookup tables are described in
    Noble, "Scientific Forth", pp.

    If we need to sample a given random distribution, but have
    only a prng producing numbers y[k] evenly distributed on [0,1],
    we must solve the (generally transcendental) equation

        P(x<X[k]) = y[k]

    where P(x<X) is the cumulative distribution. In some cases this is
    not known in closed form and must be obtained by numerical quad-
    rature. Thus the inversion process can become a serious bottleneck
    in a large simulation, where many samples are required.

    The defining word DIST: creates a table with N entries

        x[k] = P^{-1} ( k/N ), k=0, 1, ... , N-1

    The lookup code defined by DOES> employs a prng with a long
    cycle to traverse the table in random order. This is equivalent
    to approximating the distribution function p(x) by a set of step
    functions (i.e. a histogram) of uneven width.


    IMPORTANT NOTE:
        Make sure the version of prng.f that you load is the version of
        November 24th, 1998 - 21:21. Earlier versions WILL NOT WORK!

[THEN]

MARKER -dist

: undefined    BL WORD  FIND  NIP  0=  ;

undefined s>f   [IF] : s>f   S>D  D>F  ;    [THEN]
undefined f>s   [IF] : f>s   F>D  D>S  ;    [THEN]
```

```
\ Conditionally define vectoring: for using function names as arguments
undefined  use( [IF]
: use(      ’       \ state-smart ’ for syntactic sugar
    STATE @  IF  POSTPONE LITERAL  THEN  ;  IMMEDIATE
’ NOOP  CONSTANT  ’noop
: v:   CREATE  ’noop  ,  DOES> PERFORM  ;   \ create dummy def’n
: ’dfa   ’ >BODY  ;                         ( -- data field address)
: defines    ’dfa   STATE @
             IF   POSTPONE  LITERAL    POSTPONE  !
             ELSE   !   THEN  ;  IMMEDIATE
\ end vectoring
[THEN]

include prng.f
include ansfalsi.f

\ program begins here

: SF,   \ "comma in" an IEEE 32-bit number
    HERE  SF!  1 SFLOATS  ALLOT  ;

: random_seed   ( -- d)
    TIME&DATE                   ( sec min hour day month year)
    2DROP 2DROP  60 *  +  S>D  ;

: random_ndx    ( adr -- index)
    DUP  CELL+                      ( -- adr adr+cell)
    (rand) bigdiv                   ( f: -- seed 2**31-1)
    FSWAP  FDUP  F0<        ( -- f) ( f: -- 2**31-1 seed)
    IF   FOVER  F+   THEN
    FSWAP  F/               ( adr)  ( f: prng)
    @  s>f  F*  f>s  ;      ( index)

: )DIST:    ( xt n --)              \ dist_name converts xi to X(xi)
    CREATE                          \ make entry under dist_name
        TUCK  ,                     \ save size
        random_seed  , ,            \ initialize seed
        SWAP  DUP  s>f              ( -- xt n)  ( f: -- n)
        0e  SF,                     \ first entry is 0.0e
        1 DO                        \ table will have n entries
            I s>f   FOVER   F/      ( xt)  ( f: -- n I/n)
            DUP  EXECUTE            ( xt)  ( f: -- n X[I/n])
            SF,                     \ store in table
        LOOP   DROP  FDROP          ( --)  ( f: --)
    DOES>   DUP  random_ndx  ( dup >R )
            SFLOATS  +  3 CELLS +   SF@   ( R> .  f.)  ;

\ Say:   use( inverse_dist_name  Table_size )DIST:  dist_name
\ auxiliary words with distribution
```

```
\ Example: inverse Poisson distribution

FVARIABLE xi

\ Want to solve P(X) = 1 - (1+X)*e^(-X) = xi ;  let u = e^(-X) or X = -ln(u)
: f1   ( f: u -- 1-[1-ln[u]]*u - xi )
    1e0  FOVER  FLN  F-  F*  FNEGATE  1e0  F+   xi F@  F-   ;

: invP      ( f: xi -- X[xi])   xi F!
    use( f1  1e-10  1e0  1e-6 )falsi    ( f: -- e^{-X})
    FLN  FNEGATE  ;
```

6. Stochastic differential equations

We now consider what happens when coefficients or driving forces of a system of differential equations are random or *stochastic*. An example is the damped harmonic oscillator bombarded by molecules at non-zero temperature:

$$m\ddot{x} + \gamma\dot{x} + m\omega^2 x = Q(t) .$$

With random driving forces the behavior of a system becomes unpredictable. There is no point in our considering the temporal behavior of a particular set of observations, since if we repeat the experiment next week we will get an entirely different set of observations. Random processes limit the useful information we can derive from a system to the distribution function of the solutions, or to ensemble averages of interesting quantities.

First let us consider what is an ensemble average. We imagine that a given experiment is run many different times, always with the same initial conditions; and that averages of coordinates $x(t)$ or velocities $\dot{x}(t)$—or of powers thereof—can be computed in the usual fashion.

If $x_k(t)$ is the value of position measured in the k'th experiment, at time t after the experiment began, then

$$\langle x(t) \rangle \overset{df}{=} \frac{1}{N} \sum_{k=1}^{N} x_k(t) ,$$

$$\mathrm{Var}\Big(x(t)\Big) \overset{df}{=} \Big\langle \Big(x(t) - \langle x(t) \rangle \Big)^2 \Big\rangle = \frac{1}{N} \sum_{k=1}^{N} \Big(x_k(t) - \langle x(t) \rangle \Big)^2 ,$$

etc. Assuming there is no steady component of the force $Q(t)$, we may say

$$\langle Q(t) \rangle \equiv 0 .$$

Since for any driving force we may express the solution as

$$x(t) = \frac{1}{m} \int_0^t ds\, K(t-s)\, Q(s) + x_0(t) ,$$

where

$$K(\tau) \stackrel{df}{=} \frac{1}{\Omega} e^{-\gamma\tau/2m} \sin(\Omega\,\tau)$$

and

$$\Omega^2 \stackrel{df}{=} \omega^2 - \frac{\gamma^2}{4m^2},$$

the ensemble average $\langle x(t)\rangle$ is just the homogeneous solution, $x_0(t)$.

To determine the mean-square position, however, we need to know something more about the random force than its (ensemble) mean of zero. From the above definitions we see that the variance,

$$\mathrm{Var}\Big[x(t)\Big] \equiv \Big\langle \Big[x(t) - \langle x(t)\rangle\Big]^2 \Big\rangle ,$$

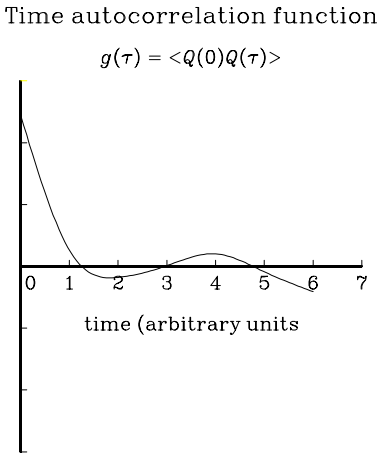
may be written as a double integral

$$\mathrm{Var}\,[x(t)] = \int_0^t du \int_0^t ds\;\; K(t-u)\,K(t-s)\,\langle Q(u)\,Q(s)\rangle .$$

The function

$$g(u-s) = \langle Q(u)\,Q(s)\rangle$$

is called the *time autocorrelation function* of the random process $Q(t)$. We have written it as a function of the difference between the two times, u and s because it is manifestly time-translation invariant.



The figure to the right displays the autocorrelation function of a pseudorandom number generator. We notice $g(u-s)$ is strongly peaked at 0, and falls off rapidly thereafter. Since the natural time scales, $1/\Omega$ and m/γ , for a noise–driven damped harmonic oscillator, are likely to be long compared to the time in which the autocorrelation function drops to zero, it is not unreasonable to approximate the latter by a δ–function:

$$g(\tau) \approx \sigma^2\delta(\tau) .$$

The variance of position is

$$\mathrm{Var}\Big[x(t)\Big] = \frac{\sigma^2}{m^2}\int_0^t ds\,\Big[K(s)\Big]^2$$

where we have used the obvious identity

$$\int_0^t ds\, f(t-s) \equiv \int_0^t ds\, f(s) .$$

Similarly, the velocity is given by the time derivative of $x(t)$,

$$\dot{x}(t) = \frac{1}{m}\int_0^t ds\,\Lambda(t-s)Q(s) + \dot{x}_0(t) - \frac{\gamma}{2m}x(t) ,$$

where

$$\Lambda(\tau) \stackrel{df}{=} e^{-\gamma\tau/2m}\cos(\Omega\tau) .$$

Therefore the mean velocity is

$$\langle \dot{x}(t) \rangle = \dot{x}_0(t) - \frac{\gamma}{2m} x_0(t) ,$$

and the corresponding variance is

$$\mathrm{Var}\left[\dot{x}(t)\right] = \sigma^2 \int_0^t ds \left[\Lambda(s)\right]^2 + \frac{\gamma^2\sigma^2}{4m^2}\int_0^t ds \left[K(s)\right]^2 + \frac{\gamma\sigma^2}{2m}\int_0^t ds\, K(s)\Lambda(s) .$$

We are interested in the behavior of the particle once it has reached equilibrium and forgotten its initial conditions. For large times,

$$\int_0^t ds \left[K(s)\right]^2 \to \frac{m}{2\gamma\omega^2}$$

$$\int_0^t ds \left[\Lambda(s)\right]^2 \to \frac{\Omega^2}{2\gamma\omega^2}$$

and

$$\int_0^t ds\, K(s)\Lambda(s) \to \frac{\gamma}{4m\omega^2} .$$

Combining these we find that after a long time the mean energy,

$$\langle H \rangle = \frac{m}{2}\left[\langle \dot{x}(t)^2 \rangle + \omega^2 \langle x(t)^2 \rangle\right] ,$$

has the limit

$$\langle H \rangle \to \frac{m^2\sigma^2}{2\gamma} .$$

That is, in equilibrium, an under-damped harmonic oscillator subject to a random driving force has mean energy independent of the oscillator frequency[11].

Now, suppose the random force represents collisions of molecules in a liquid or gas with the oscillator. (Of course, in that case the damping constant also arises from molecular collisions, *i.e.* viscosity.) Following Einstein, we can relate the constants σ^2 and γ to the temperature of the bath. A free particle in such a bath will, by the equipartition theorem, have average kinetic energy $\frac{1}{2}k_B T$ per degree of

11. Obviously any (damped) solution of the homogeneous equation contribute to the energy a term proportional to $e^{-\gamma t/m}$ that vanishes for large times.

freedom. It is easy to see (for example, take the limit of the over-damped solution of the harmonic oscillator) that in the limit $\omega^2 \to 0$, the average kinetic energy of a (damped) free particle is

$$\frac{1}{2} m\langle \dot{x}(t)^2 \rangle = \frac{m\sigma^2}{4\gamma} \equiv \frac{1}{2} k_B T .$$

In Brownian motion—the diffusion of small objects suspended in a liquid at (absolute) temperature T—there are no restoring forces, harmonic or otherwise. If one represents the density of objects as $\rho(x,t)$ then, as Einstein showed, ρ obeys the diffusion equation

$$\frac{\partial}{\partial t} \rho(x,t) = D \frac{\partial^2}{\partial x^2} \rho(x,t) .$$

where D is the *diffusion constant*. Since ρ represents the probability for finding a particle between x and $x+dx$ at time t, the mean position of the particles is

$$\langle x(t) \rangle = \int_{-\infty}^{\infty} dx \, x \, \rho(x,t) ;$$

it is then easy to see (integrate by parts) that

$$\langle x(t) \rangle = \text{constant} .$$

Similarly, the variance of the position is

$$\text{Var}\left[x(t)\right] = \int_{-\infty}^{\infty} dx \left(x^2 - \langle x(t) \rangle^2 \right) \rho(x,t) = 2D\,t$$

Comparing this with the variance (in position) of a free particle in a liquid or gas (the limit $\omega^2 \to 0$ of the damped harmonic oscillator with a random driving force),

$$\text{Var}\left[x(t)\right] = \frac{2k_B T}{m\gamma} t ,$$

we obtain Einstein's expression

$$D = \frac{k_B T}{6\pi\eta a} ,$$

for spherical particles of radius a, in a liquid of viscosity η, obeying Stokes' Law:

$$\gamma = 6\pi\eta a .$$

That is, measurement of the diffusion constant of a suspension of spheres allows direct measurement of the Boltzmann constant—or equivalently, of Avogadro's number. Perrin won the 1928 Nobel Prize in Physics for this and related measurements.

Returning to the one-dimensional, damped harmonic oscillator in a thermal bath, we see its average energy is $k_B T$. This was a key result used in deriving the Planck black body radiation formula before the modern formulation of quantum mechanics.