DATA ANALYSIS METHODS IN PHYSICAL OCEANOGRAPHY

THIRD EDITION

RICHARD E. THOMSON
Fisheries and Oceans Canada
Institute of Ocean Sciences
Sidney, British Columbia
Canada

and

WILLIAM J. EMERY
University of Colorado
Aerospace Engineering Sciences Department
Boulder, CO
USA

ELSEVIER

AMSTERDAM • BOSTON • HEIDELBERG • LONDON • NEW YORK • OXFORD
PARIS • SAN DIEGO • SAN FRANCISCO • SINGAPORE • SYDNEY • TOKYO

MAURY OCEANOGRAPHIC LIBRARY
Stennis Space Center, MS 39522-5001

rec'd 5/2015 ($150.00)

dissimilar neighbor. In this way, the layer begins to grow in breadth; τ_1 is therefore called the "breadth controlling parameter". If τ_1 is set to zero, the map would grow to infinity, so the idea is to choose a reasonable value for τ_1. The default value in the MatLab toolbox is 0.3, which indicates that the mean or cumulative QE must be 70% greater than that for the preceding QE for the map to stop growing. Mihály and Thomson (unpublished) found that a higher value (0.6) improved the analysis.

Step 3

If the inequality Eqn (4.123) is still not satisfied, the strategy is to pick map units, which might be expanded further into a new hierarchical layer. A second inequality is used:

$$QE_i > \tau_2 \langle QE \rangle_0 \tag{4.124}$$

where τ_2 is smaller than τ_1 ($0 < \tau_2 \leq \tau_1 < 1$); here, τ_2 is called the "depth controlling parameter" and has a MatLab default value 0.03 (Mihály and Thomson used a lower value of 0.005). In this process, the quantization error of each individual map is compared to the quantization error of the "layer 0" map unit. As a consequence, each single map unit that satisfies the above inequality is expanded in a subsequent layer. In general, decreasing the parameter τ_1, grows SOM arrays with greater breadth, while decreasing the parameter τ_2, grows SOMs with more layers in the GHSOM hierarchy. Choosing the appropriate parameters re-introduces a strong degree of subjectivity back into the analysis. Using GHSOM for a single layer illustrates the hierarchical aspect of the process while preserving the objectiveness of the approach.

4.10 KALMAN FILTERS

In 1960, Rudolf E. Kalman published his now famous paper describing a recursive solution to the discrete-data linear filtering problem of trying to estimate the state y, which is governed by the linear stochastic difference equation,

$$y_k = Ay_{k-1} + Bu_{k-1} + \text{random (white) noise} \tag{4.125}$$

The matrix A relates the state at the present step, k, with that at the previous step, $k-1$, and the matrix B relates the optional control input (u, such as the Navier–Stokes equations of motion) to the state y. Kalman's original interest was in determining the orbits of planets from limited Earth observations but the filter soon found extensive use in autonomous and assisted navigation, with both civilian and military applications. The Kalman Filter is an efficient optimal estimator (a set of mathematical equations) that provides a recursive computational methodology for estimating the state of a discrete-data controlled process from measurements that are typically noisy, while providing an estimate of the uncertainty of the estimates. Because the filter is recursive, new measurements can be processed as they are received from external sensors. Unlike recursive filters, the method doesn't need to store all previous measurements nor reprocess all data at each time step. If the noise associated with all processes and data contributing to the state have Gaussian (normal) distributions, then the Kalman filter obtains solutions by minimizing the mean square error of the estimated parameters. If the Probability Density Functions of the variables are not known, and only the mean and standard deviation of the noise are available, the Kalman filter is still the best linear estimator; some non-linear estimators may be better than the linear estimators.

Kalman filters have a number of advantages. Specifically, the filter provides reliable practical results due to its optimality, it is designed for real time processing, and equations involving the measurements do not need to be inverted as part of the solution. Moreover, the method is

easy to formulate a
understanding of t
As with other filters
filter is to determin
parameters, Y, fr
Kalman filter not o
parameters within
surement uncertain
count the error of t
the error of the pre
filters fuse predicti
on a weighted diffe
surement and a m
method proceeds f
of the state y_k at ste
tion of the process
minus sign supers
mate of the state, $\hat{y}$
newly acquired m
Bishop, 2006) Speci

$$\hat{y}_k = \hat{y}_k^- +$$

where the new m
terms of the actual

$$z_k$$

plus some uncer
random (white r
a Gaussian probat
a matrix linking th
matrix K is *gain* or
ence $z_k - H\hat{y}_k^-$ is
innovation. The erro
ori estimates are th

$$e_k^-$$

$$e_k$$

with correspondin
(traditionally writt
we did earlier)

$$P_k^-$$

$$P$$

easy to formulate and implement, given a basic understanding of the underlying mathematics. As with other filters, the purpose of the Kalman filter is to determine the "best" estimate of state parameters, Y, from noisy input data. The Kalman filter not only determines values of the parameters within the constraints of the measurement uncertainty but it also takes into account the error of the measurements relative to the error of the predictions. In essence, Kalman filters fuse prediction and measurement based on a weighted difference between an actual measurement and a measurement prediction. The method proceeds from an *a priori* estimate, $\widehat{y}_k^-$, of the state y_k at step k that is based on information of the process prior to step k (hence the minus sign superscript), to an *a posteriori* estimate of the state, $\widehat{y}_k$, at step k that incorporates newly acquired measurements, z_k (Welch and Bishop, 2006) Specifically,

$$\widehat{y}_k = \widehat{y}_k^- + K\left(z_k - H\widehat{y}_k^-\right) \tag{4.126}$$

where the new measurement, z_k, is given in terms of the actual state of the process, y_k,

$$z_k = Hy_k + v_k \tag{4.127}$$

plus some uncertainty, characterized by a random (white noise) variable, v_k, having a Gaussian probability distribution. Here, H is a matrix linking the state to the measurement, matrix K is *gain* or *blending factor*, and the difference $z_k - H\widehat{y}_k^-$ is the measurement *residual* or *innovation*. The errors of the *a priori* and *a posteriori* estimates are then, respectively,

$$e_k^- = y_k - \widehat{y}_k^- \tag{4.128a}$$

$$e_k = y_k - \widehat{y}_k \tag{4.128b}$$

with corresponding error covariance functions (traditionally written as P rather than Cov as we did earlier)

$$P_k^- = E\left[e_k^- e_k^{-T}\right] \tag{4.129a}$$

$$P_k = E\left[e_k e_k^T\right] \tag{4.129b}$$

The key to solving Eqn (4.126) is to find that matrix K that minimizes, in the usual least squares sense, the a posteriori error covariance in Eqn (4.129b). This is accomplished by substituting Eqn (4.126) into Eqn (4.128b), which is then substituted into Eqn (4.129b), deriving the indicated expected values, taking the derivative of the trace of the result with respect to K, setting the result to zero (corresponding to minimization), and then solving for K. One expression for K that minimizes Eqn (4.129b) is (Welch and Bishop, 2006)

$$K_k = \frac{P_k^- H^T}{\left(HP_k^- H^T + R\right)} \tag{4.130}$$

where R is the measurement error covariance. Note that as R approaches zero, $K_k \to 1/H$, so the gain K weights the residual between the observation and the initial estimate, $K(z_k - H\widehat{y}_k^-) \to (y_k - \widehat{y}_k^-)$, more favorably. In contrast, as the *a priori* estimate error covariance P_k^- approaches zero, $K_k \to 0$ so that the gain K weights the new estimate more favorably than the data.

As an example, consider the problem of determining the precise location of a sailboat participating in a trans-oceanic race. We assume that the boat is equipped with a Global Positioning System (GPS) unit that provides an estimate of the vessel's position within a few meters but that the calculated positions are noisy, with values that "dance" around while remaining within a few meters of the actual position. (Early in our careers, we depended on Loran-C navigation for positioning in offshore regions, for which fixed positions could jump around by as much as several "cables" over periods of several minutes; a cable is 1/10 of a nautical mile.) Because the heading and speed of the boat are known, the boat's location can be computed from the GPS information provided the influence of the surface current and wind on the boat are known or can be estimated.

This "dead-reckoning" method—which, in the not-so-distant past, was one of the only ways sailing ships could determine their position at sea—yields smooth estimates of the vessel's position but which drift over time. As in other applications, the Kalman filter can be thought of as operating in two distinct phases of dead-reckoning: *predict* and *update*. During the prediction phase, the boat's position can be modified by the external forces acting on the boat (the dynamic or "state transition" model). In addition to calculating a new estimate of the vessel's position, a new covariance function can also calculated. Next, in the update phase, a measurement of the boat's position is obtained from the GPS, a measurement that, once again, has a degree of uncertainty. The covariance between the newly predicted position and that of the prediction from the previous phase determines how much the new measurement will affect the updated prediction. Ideally, if the dead-reckoning estimates tend to drift away from the real position, the GPS measurement will help "nudge" the position estimate back toward the real position, while not perturbing the system to the point that the predictions become rapidly changing and noisy. We have summarized the above procedure in Figure 4.45 where the first prediction phase is based on previously available data and precedes the collection of new data in the measurement phase. The variances of the two steps are related.

In Figure 4.45, the conditions for the state and its variance at the previous data step are $\hat{y}_{k-1}$ and σ_{k-1}, respectively; the prediction for the state and the variance at the next step are $\hat{y}_k^-$ and σ_k^-.

We use the dynamical model (i.e., physical system), together with the initial conditions, to make this prediction. We then take the measurement z_k and compute the corrected state, $\hat{y}_k$ (with variance σ_k), by blending the prediction and the residual, which is always a case of merging two Gaussian variables. The result is an optimal estimate with a smaller variance.

The blending factor, K, is controlled by knowledge gained from the measurements or from the prediction. As noted earlier, if the prediction is more certain, then the prediction error covariance P_k decreases to zero, K also approaches zero and the filter weights the prediction more heavily than the residual, $z_k - H\hat{y}_k^-$. In contrast, if the measurements are more certain that the prediction, then the measurement covariance (R) approaches zero as K approaches H^{-1} and the filter weights the residual more heavily than the prediction. The computational basis for the Kalman filter equations are outlined in Figure 4.46, which shows both the prediction and measurement phases of the Kalman filter.

In Figure 4.46, the recursive nature of the filter equations is indicated by the arrows at the top and bottom of the boxes. The boxes represent the prediction and measurement steps of the Kalman filter. The basic assumptions behind the filter are:

1. The model used to predict the "state" needs to be a linear function of the measurements;
2. The model error and the measurement error (measurement noise) must both be Gaussian with zero means.

FIGURE 4.45 The sequence of the Kalman equations for estimation of the variable y.

The Kalman equations

Make prediction based on previous data: $\hat{y}^-, \sigma^-$

↓

Take measurement: z_k, σ_z

↓

Optimal estimate ($\hat{y}$) = Prediction + (Kalman gain) * (Measurement − Prediction)

Variance of estimate = Variance of prediction * (1 − Kalman gain)

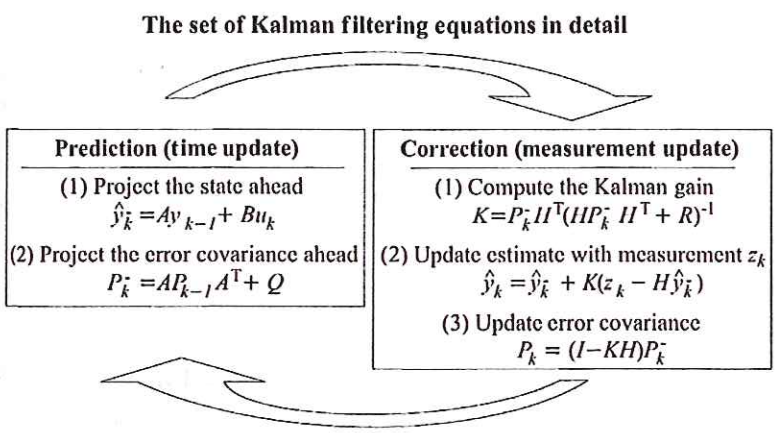

FIGURE 4.46 The Kalman filter equations (see text for details and functions).

Even if the noise is not Gaussian, and we know only the mean and standard deviation of the noise and not its probability distribution, the Kalman filter is still the best linear estimator. For highly non-Gaussian distributions, nonlinear estimators may be preferable.

It worth emphasizing that the Kalman filter combines a system's dynamics (i.e., the physical laws of motion) known to control the inputs to that system with multiple sequential measurements to form an estimate of the systems varying quantities (its state) that is better than the estimate obtained by using any one measurement alone. Noisy sensor data, approximations to the physical equations that describe how a system changes with time, and unaccounted for external factors, introduce uncertainty in the system's state. The Kalman filter averages a prediction of the system's state with a newly acquired measurement whose contribution is determined by a series of weights. These weights, which also take into account the "trustworthiness" of the data relative to the prediction, are calculated from the covariance function, a measure of the estimated uncertainty of the prediction of the system's state. As a result, the new state estimate lies between the predicted and the measured state, and has a smaller estimated uncertainty than either of the two alone. This process is repeated for every time step, with the new estimate and its covariance forming the prediction used in the subsequent iteration. Hence, the recursive character of the Kalman filter, which requires only the last "best guess" rather, then the entire history of a system's state to calculate the new state.

Since the certainty of the measurements is often difficult to specify precisely, it is common to discuss the filter's behavior in terms of its gain. The Kalman gain depends on the relative uncertainty of the measurements and the current state estimate, and it can be adjusted to achieve a particular desired performance of the filter. For a high gain, the filter places more weight on the measurements and follows them more closely; for a low gain, the filter conforms more to the model predictions smoothing out noise and decreasing the responsiveness of the system. At the extremes, a gain of zero causes the measurements to be ignored completely while a gain of unity causes the estimate of the state to be ignored entirely.

Implementation of the Kalman filter is often difficult to achieve in practice due to the requirement for a good estimate of the noise covariance matrices. A study by Furrer and Bengtsson (2007) used Monte Carlo methods to estimate the Kalman filter variants. Another promising

approach is the Autocovariance Least Squares (ALS) technique that uses the autocovariance of the data to estimate the noise covariance. A study by Odelson et al. (2005) demonstrates that the noise covariances estimated in this way are unbiased and converge to the true values with increasing sample size. They also add positive semi definiteness constraints to these covariances. Abdel-Hafez (2008) uses the same technique to estimate the Global Positioning System (GPS) measurement noise-covariance matrix.

The Kalman filter is known to be optimal since: (a) the model perfectly matches the real system; (b) the measurement noise is white; and (c) the covariances of the noise are exactly known. We have already discussed methods to estimate the error covariance's. Once these are known, it is useful to estimate the performance of the Kalman filter itself; i.e., to determine whether it is possible to further improve the state estimation quality. We also know that if the Kalman filter works optimally, the output prediction error is also white noise. This white noise character properly reflects the state estimation quality. To evaluate the performance of the Kalman filter, it is necessary to inspect the "whiteness" of the predictions.

An interesting example of Kalman filter application is given by www.cs.cornell.edu/Courses/cs4758/2012sp/.../MI63slides.pdf. This example concerns tracking the fluid level in a tank being filled. For this problem, we can write the analysis sequence as:

Predict:

$$\widehat{y}_{t|t-1} = A_t \widehat{y}_{t-1|t-1} + B_t u_t \qquad (4.131a)$$

$$P_{t|t-1} = A_t P_{t-1|t-1} A_t^{\mathrm{T}} + Q_t \qquad (4.131b)$$

where: $\widehat{y}$ is the estimated state; A is the state transition matrix (i.e., the matrix describing the transition between states); u represents the optional control variables on the state; B is the control matrix (i.e., the matrix mapping the control variables to the state variables); P is the state error covariance matrix (i.e., error of the estimation); and Q is the process covariance error matrix (i.e., the error due to the process).

Subscripts are: $t|t$ = the current time, $t-1|t-1$ = the previous time, and $t|t-1$ is time at intermediate steps. The Kalman filter removes noise from the system by assuming a pre-defined model of the system. This model should be defined by the following:

1. Understand the situation: Examine the problem and break it down in to the mathematical basics.
2. Model the state process: Start with a basic model, which may not be perfect at first, but can be refined later.
3. Model the measurement process: Analyze how to measure the process. The measurement space may not be in the same space as that of the state (e.g., using an electrical diode to measure weight, an electrical reading does not directly translate to weight).
4. Model the noise: This needs to be done both for the state and the measurements processes. The basic Kalman filter assumes Gaussian white noise so that the variance and the covariance (error) functions are meaningful (i.e., make sure that the error you model is suitable for the situation).
5. Test the filter: This step is often overlooked. Use synthetic data if needed. See if the filter is behaving as it should.
6. Refine filter: Try to change the noise parameters (filter), as these are the easiest to change. If necessary go back further and rethink the situation.

As an example, consider the water in a tank (Figure 4.47) and assume there is a measurement parameter that gives the water level of the tank as it fills or empties. The goal is to estimate the unknown level of water in this tank. In this example, the water level measurements are provided by a float in the tank. The tank could be: (a) Filling, emptying or static (level is increasing, decreasing or constant); or (b) sloshing around or

FIGURE 4.47
provided by the

static (level cl
and constant)
model in whic
izontal) and th
Eqn (4.10.7), v
scalar so that
of L. Since we
there is no tin
$A = 1$ for any i

We now n
process, z, wh
precisely the
we need to
simplicity, we
measurement,
Eqn (4.130) for
a scalar so we
process is not
so that $Q = q$.
(4.131) whereb

$p_{t|t-1}$

The update i

$y_{t|t} = y_{t|}$

$K_t = p$

$p_{t|t} =$

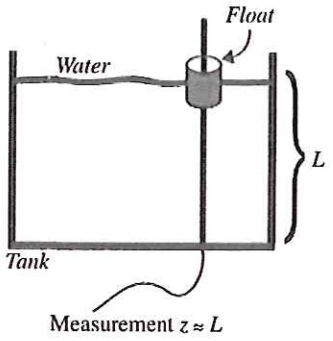

FIGURE 4.47 Water level in a tank. Measurements are provided by the float.

static (level changes from side to side or is flat and constant). We first consider the most basic model in which the fluid in the tank is level (horizontal) and the surface is constant ($L = c$). Using Eqn (4.10.7), we can write the state variable as a scalar so that $\hat{y} = y$, where y is now the estimate of L. Since we are assuming a constant model, there is no time variation and $y_{t+1} = y_t$, so that $A = 1$ for any $t \geq 0$. Both B and $u = 0$.

We now need to model the measurement process, z, which for simplicity we assume is precisely the water level in the tank. Finally, we need to model the noise. Again, for simplicity, we assume that there is noise in the measurement, and that $R = r$ in the expression Eqn (4.130) for the Kalman gain. The process is a scalar so we can assume that $P = p$; since the process is not well defined, we adjust the noise so that $Q = q$. To test the filter, we apply Eqn (4.131) whereby

$$y_{t|t-1} = y_{t-1|t-1} \tag{4.132a}$$

$$p_{t|t-1} = p_{t-1|t-1} + q_t \tag{4.132b}$$

The update is

$$y_{t|t} = y_{t|t-1} + K_t(z_t - y_{t|t-1}) \tag{4.133a}$$

$$K_t = p_{t|t-1}(p_{t|t-1} + r)^{-1} \tag{4.133b}$$

$$p_{t|t} = (1 - K_t)p_{t|t-1} \tag{4.133c}$$

where, the new data value is z_t. The filter is now completely defined. To put some numbers into this model, we first assume that the true level is $L = 1$. We initialize the state with an arbitrary number, with an extremely high variance, as it is completely unknown. Specifically, $y_0 = 0$ and $p_0 = 1000$. If one initializes with a more meaningful value, the filter solution will convergence faster. We choose the system noise as $q = 0.0001$ since we believe that we have a realistic and accurate model. Thus,

Predict 1:

$$y_{1|0} = 0$$

$$p_{1|0} = 1000 + 0.0001$$

The hypothetical measurement we obtain from the float is $z_t = z_1 = 0.9$, which differs from the true water level value of $L = 1.0$ because of noise. Assuming a measurement noise of $r = 0.1$, we can write:

Update 1:

$$K_1 = (1000 \times 0.0001)(1000 \times 0.0001)^{-1}$$

$$= 0.9999$$

$$y_{1|1} = 0 + 0.9999(0.9 - 0) = 0.8999$$

$$p_{1|1} = (1 - 0.9999)(1000 \times 0.0001) = 0.1$$

As indicted by this first update step, the initialization value $y_{1|0} = 0$ has been brought close to the true value, $L = 1$, of the system. In addition, the error variance, $p_{1|1}$, has diminished to a more reasonable value.

If we do another step, we have:

Predict 2:

$$y_{2|1} = 0.8999$$

$$p_{2|1} = 0.100 + 0.001 = 0.1001$$

And, assuming a second measurement $z_2 = 0.8$ (again differing from unity because of noise), we find:

Update 2:

$$K_2 = 0.1001(0.1001 + 0.1)^{-1} = 0.5002$$

$$y_{2|2} = 0.8999 + 0.5002(0.8 - 0.8999) = 0.8499$$

$$p_{2|2} = (1 - 0.5002)0.1001 = 0.0500$$

If we continue this process, we eventually obtain the results as shown in Table 4.11.

From this table, we can see that the model works successfully. After stabilization at about time step 4, the estimated state is within 0.05 of the true value, while the specified measurements are between 0.8 and 1.2 (only within 0.2 of the true value). The results are plotted in Figure 4.48 where the purple line represents the values estimated by the Kalman filter; the yellow line is the true value (a constant) and the dark blue line are the input measurements.

As shown, the Kalman filter provides an optimal estimate of the true values even when the measurements are very noisy (a 20% errors in measurements resulted in only a 5% inaccuracy in the Kalman filter estimate). Hence, the Kalman filter has achieved its purpose.

We now examine a more realistic case in which the tank fills at a constant rate, f, such that the water level $L_t = L_{t-1} + f$. We assume that $f = 0.1$ per unit time and start with $L_0 = 0$. We will also assume that the measurement and the process noise are constant with time (i.e., $q_t = 0.001$ and $r_t = 0.1$). Table 4.11 now takes the form shown in Table 4.12.

We see that, over time, the estimated state stabilizes (i.e., the variance becomes very small). While the estimate reduces the noise, it dramatically underestimates the true value, L, which is much closer to the measured values, z (Figure 4.49).

It is clear from Figure 4.49 that the estimates systematically underestimate the true and even the measured values. This is not a very attractive result even though the noise level has been greatly reduced. There are two possible causes for this problem: (a) the model we have chosen; and/or (b) the reliability of our process model (our chosen q value). The easiest corrective approach is to change the q value. A valid question is "Why did we chose $q = 0.0001$ to begin with?". The answer is that we thought our model was a good estimation of the true process and that the error level would be quite small. Apparently, our model was not as good as we had anticipated so we need to relax this requirement. Specifically, we now assume there

TABLE 4.11 Kalman Filter Applied to the Water Tank Problem (Figure 4.47)

	Predict		Update			
Time (t)	$y_{t\|t-1}$	$p_{t\|t-1}$	z_t	K_t	$y_{t\|t}$	$p_{t\|t}$
3	0.8499	0.0501	1.1	0.3339	0.9334	0.0334
4	0.9334	0.0335	1	0.2509	0.9501	0.0251
5	0.9501	0.0252	0.95	0.2012	0.9501	0.0201
6	0.9501	0.0202	1.05	0.1682	0.9669	0.0168
7	0.9669	0.0169	1.2	0.1447	1.0006	0.0145
8	1.0006	0.0146	0.9	0.1272	0.9878	0.0127
9	0.9878	0.0128	0.85	0.1136	0.9722	0.0114
10	0.9722	0.0115	1.15	0.1028	0.9905	0.0103

The Predict and Update Parameters are Explained in the Text.

FIGURE 4.48 K
input to the filter

TABLE 4

Time (t)
0
1
2
3
4
5
6

The Last Colu

is a greater erro
$q = 0.01$. Applic
previous model
Figure 4.50.

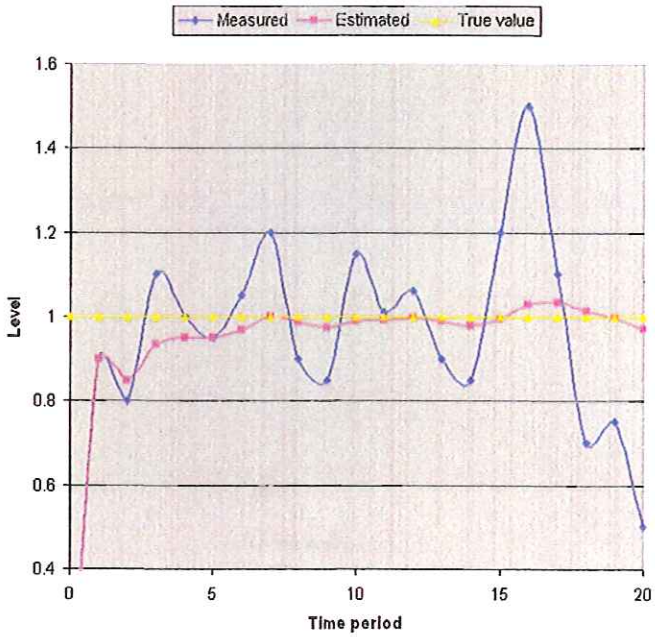

FIGURE 4.48 Kalman filter estimates (purple line) of a constant water level (yellow line) in a tank. The measured values (z) input to the filter are shown in dark blue.

TABLE 4.12 Kalman Filter Applied to the Water Tank problem (Figure 4.47) for The Specific Values Presented in the Text (Specifically, $f = 0.1$ per Unit Time and Initial Water Level $L_0 = 0$)

	Predict		Measurement and Update				Actual
Time (t)	$y_{t\|t-1}$	$p_{t\|t-1}$	z_t	K_t	$y_{t\|t}$	$p_{t\|t}$	L
0	—	—	—	—	0	1000	0
1	0.000	1000 × 0.0001	0.11	0.9999	0.1175	0.100	0.1
2	0.1175	0.1001	0.29	0.5002	0.2048	0.0500	0.2
3	0.2048	0.0501	0.32	0.3339	0.2452	0.0334	0.3
4	0.2452	0.0335	0.50	0.2509	0.3096	0.0251	0.4
5	0.3096	0.0252	0.58	0.2012	0.3642	0.0201	0.5
6	0.3642	0.0202	0.54	0.1682	0.3945	0.0168	0.6

The Last Column Gives the Actual Water Level.

is a greater error with our process model and set $q = 0.01$. Application of the new q-value to the previous model results in the values plotted in Figure 4.50.

As indicated by Figure 4.50, changing q greatly improved the accuracy of the estimate, although the estimates are still significantly below the true and measured values. If we

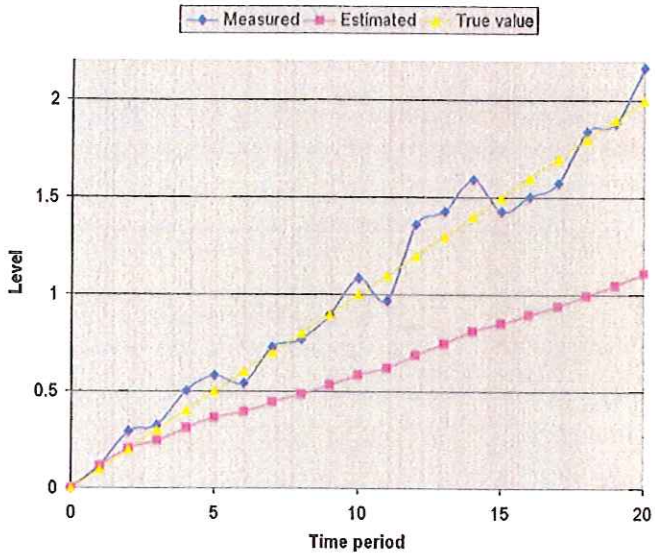

FIGURE 4.49 Kalman filter estimates (purple line) of a water tank filling at a constant rate $f = 0.1$ per unit time with constant error $q = 0.0001$ per unit time. The yellow line shows the true water level, L, and the dark blue line the water level measured by the float.

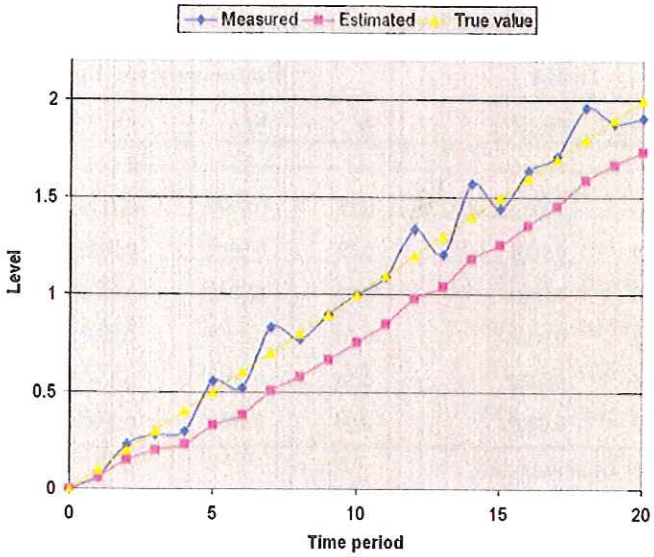

FIGURE 4.50 As with Figure 4.49, but with a much smaller constant error $q = 0.01$ for the process model.

Level

FIGURE 4.51 As with I

increase q once again I obtain the results plottec shows an even greater i man filter estimate begins surements, an intentional q, but still has a bit less n points. The filter estima true value more effectivel ing smaller noise values.

It is clear that further i noise level will cause th increasingly match the m no benefit to using the Ka here is that a poorly defin vide a good filter estimat the estimated error will a estimate to rely more values, while still allowin

The primary use of Kal ical oceanography is fo oceanographic data int (Pham et al., 1998). This Extended Kalman Filter

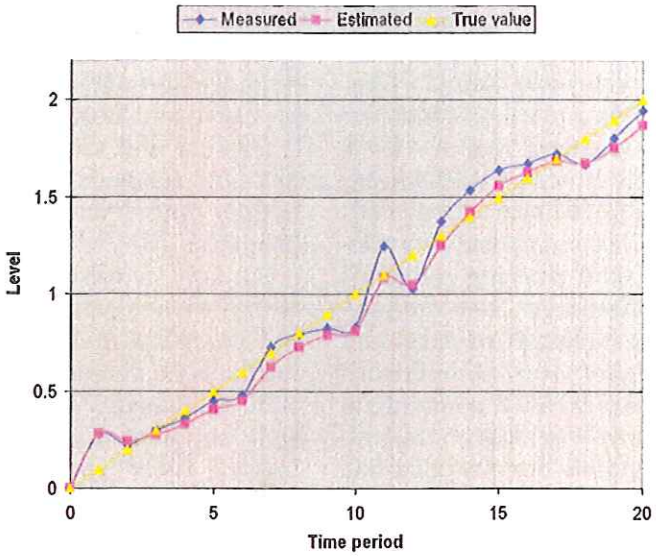

FIGURE 4.51 As with Figure 4.49, but with an even much smaller constant error $q = 0.1$ for the process model.

increase q once again by setting $q = 0.1$, we obtain the results plotted in Figure 4.51, which shows an even greater improvement. The Kalman filter estimate begins to track the noisy measurements, an intentional outcome of increasing q, but still has a bit less noise than the measured points. The filter estimates now approach the true value more effectively than the estimates using smaller noise values.

It is clear that further increases in the process noise level will cause the estimated values to increasingly match the measured values, giving no benefit to using the Kalman filter. The lesson here is that a poorly defined model will not provide a good filter estimate. However, increasing the estimated error will allow the Kalman filter estimate to rely more on the measurement values, while still allowing some noise removal.

The primary use of Kalman filtering in physical oceanography is for the assimilation of oceanographic data into numerical models (Pham et al., 1998). This application uses the Extended Kalman Filter (EKF), which applies to non-linear state systems or measurement equations. Here, a Kalman filter is a linearized version of these equations, which continues to yield an optimal solution. The key to this form of data assimilation is to approximate the error covariance of the data to be assimilated into the numerical model. This procedure amounts to making no correction in those directions for which the error is the most attenuated by the system. This has the added benefit of improving the filter stability. These "directions of correction" evolve with time according to the model evolution, which is a primary feature of this filter that distinguishes it from other sequential assimilation methods. Pham et al. (1998) suggest a method for initializing the filter based on EOFs discussed earlier in this chapter. They examine assimilation of wind stress forcing into a simple quasi-geostrophic (QG) model for a square ocean domain. Although this is an unrealistic test case, the result of this assimilation method are very encouraging.

In a study of coastal ocean problems, Chen et al. (2009) compared the reduced rank Kalman filter (RRKF) with the ensemble Kalman filter (EnKF) and the ensemble square-root Kalman filter (ENSKF) in three idealized regimes: (a) A flat bottom circular shelf driven by tidal forcing at the open boundary; (b) a linear slope continental shelf with river discharge; and (c) a rectangular estuary with tidal flushing intertidal zones and freshwater discharge. They used the unstructured grid Finite-Volume Coastal Ocean Model (FVCOM). Model run comparisons showed that the success of the data assimilation method depends on sampling location, assimilation methods (univariate or multivariate covariance approaches), and the nature of the dynamical system. In general, for these applications, the EnKF and EnSKF work better than RRKF, particularly for time dependent cases with large perturbations. In EnKF and ENSKF, multivariate covariance methods should be used to avoid the appearance of unrealistic numerical oscillations. Since the coastal ocean features multiscale dynamics in both time and space, an individual case-by-case approach should be used to determine the most efficient and reliable data assimilation approach for different dynamical systems.

4.11 MIXED LAYER DEPTH ESTIMATION

Much of this chapter has focused on methods for constructing smoothed oceanic fields from large-scale spatial and temporal data. Some methods take into account the dynamics of the physical system, while others take a strictly statistical approach. Little has been said about methods designed to delineate physical "breaks" in oceanic distributions. Here, we gave a brief overview of techniques used to define the depth of the surface mixed layer—the top layer of the ocean characterized by uniform to nearly uniform vertical water property structure—which is of interest to a broad variety of oceanic studies including upper ocean productivity, air-sea exchange processes, and climate variability (cf. Curry and Roy, 1989; Robinson et al., 19[illegible] Wijesekera and Gregg, 1996; and Kara et al. 2000a,b). Methods for determining temporal "regime shifts" in the ocean are presented in Chapter 5.

Surface windstress, convective cooling, breaking waves, current shear, and other turbulent processes in the upper ocean generate a surface layer characterized by uniform to near-uniform density, active vertical mixing, and high turbulent dissipation. The depth of this "mixing layer" is ultimately determined by a balance between the destabilizing effects of mechanical mixing and the stabilizing effects of surface buoyancy flux. As a result of temporal variations in these opposing affects, the mixing layer may be imbedded in a deeper "mixed layer" of almost identical density to the mixing layer and representing the time-integrated response to previous mixing events. Mixed layer depth (MLD) can vary by tens of meters over a diurnal cycle and by over 100 m over an annual cycle (cf. Large et al., 1994).

In the absence of direct turbulent dissipation measurements, mixed layer depth is derived from oceanic profile data using a variety of proxy variables. Methods for estimating the MLD from CTDs and other profiling instrumentation data fall into four broad categories: (1) Threshold methods, which find a predefined step in the surface profile (Price et al. 1986; Lukas and Lindstrom 1991; Peters et al. 1988) or find a critical gradient for the upper layer (Lukas and Lindstrom, 1991; Holte and Talley, 2009); (2) least-squares regression methods, which fit two or more line segments to near-surface profiles (Papadakis, 1981, 1985); (3) integral methods, which calculate a depth-scale for the upper layer based on integral properties of the water column such as conservation of mass (Ladd and Stabeno, 2012; Freeland et al., 1997, 2013); and (4) a split-and-merge

algorithm that fits st
on a specified erro
and Fine, 2003, 2009)
mentally different ap
four consider the mi
distinct entity whose
from the observed c
dently of any integra
In contrast, method tl
as the upper compon
mation to a continuou
vation must satisf
requirements such as

4.11.1 Threshold

In air-sea interactio
and Gregg, 1996; Sm
depth of the surface r
as that depth, z, at wl
difference $\Delta\sigma_\theta(z) = \sigma_\theta$
ocean exceeds a specif
cally 0.01 kg/m^3 (Figu
reference depth (gene
the ocean surface, to
$\rho_\theta(z) - 1000$ kg/m^3 is
measured potential d
threshold method is
depth estimates can be
analytical computatic
threshold difference c
yields diurnal depth es
from turbulent dissipa
threshold difference me
0.01 kg/m^3 has becom
for many mixed layer
climate studies, where f
in MLD averaged over
longer, threshold value
m^3 are more commonly
data (cf. Table 1 in Kara
back with the 0.01 kg/r
that it often neglects th
near-identical density er
rophyll, nutrient, and