Teaching Kalman Filters To Undergraduate Students

Andrew Love
Johns Hopkins University Applied Physics Laboratory

Maurice Aburdene, Rami William Zarrouk
Bucknell University

Abstract

The Kalman filter algorithm is one of the most common estimation techniques used today, yet generally engineers do not encounter it until they have begun their graduate or professional careers, even though the concepts necessary to understand it are introduced to sophomore engineering students. This paper presents an approach intended to take undergraduate students from the concept of an average to Kalman filters in a half dozen small steps. Using MATLAB or MathCAD provides some advantages in teaching the Kalman algorithm since in those languages the algorithm can fit on two pages even with extensive commenting, and the syntax of these languages does not obscure the structure of the algorithm.

I. Introduction

Kalman filtering[1] is a widely used technique for process control and evaluation of mechanical, electrical, chemical, and medical systems[2-12], but this technique is not often taught at an undergraduate level. This is unfortunate because issues such as instrument noise, system noise, and systems models are very important to working engineers, and Kalman filters provide a convenient framework for discussing each of these issues. Furthermore, undergraduate engineering students learn all the mathematics necessary to use the Kalman filter equations (matrix differential equations and the concept of uncertainty), if not to derive them, by their sophomore or junior year, so there is no reason not to give them the opportunity to apply this knowledge. We present an approach that was used to introduce the Kalman filter concept to first semester electrical engineering juniors in a linear systems course. Rather than simply introducing the Kalman filter equations as given in the texts[2-8, 10-12], however, we have found it effective to begin with the idea of an average, and add features until the general Kalman filter is developed.

II. Averaging

The lesson begins with the introduction of a problem to be solved: determining resistance of a circuit to high accuracy with a measurement device that is not accurate enough. First, we show the 15 measurements that have been taken, which show that the measurement device has errors of about 500 ohms, about 4 times too inaccurate for the (assumed) purpose at hand. At this

point the students eagerly suggest an average (using the "batch average" equation below) as a solution

$$\bar{x} = \sum_{i=1}^{M} \frac{x_i}{M} \qquad (1)$$

where x_i is the i-th measurement of some quantity x (resistor value in our case), and M is the number of measurements, but they are often unclear as to how an average helps the situation. At this point we introduce (or remind them of) the equation for the uncertainty of an average[13]:

$$\sigma_{\bar{x}} = Average's\ Uncertainty\ = \frac{Measurement\ Uncertainty}{\sqrt{Number\ of\ measurements}} = \frac{r}{\sqrt{M}} \qquad (2)$$

It is sometimes worth taking the time to sketch out a proof of equation (2). Given that each measurement equals the true value of x plus some error, e_i. The mean of the measurements is equal to the true value of x plus the average value of the errors ($\frac{1}{M}\sum_{i=1}^{M} e_i$). The variability in the mean is entirely due to the errors (since the true value of x is constant). Therefore, since we are given that the standard deviation of the errors is r, then the standard deviation of the mean is $r/\sqrt{M}$. This is the natural point at which to introduce the assumptions behind averaging:

- The true value of x is constant.
- The measurement uncertainty value, r, is known and identical for all samples.
- Noise values are not correlated in time

After establishing that these assumptions are reasonable for the case of a resistance measurement, the students are told that the assumptions will be pared down as the lesson continues and averaging evolves into Kalman filtering.

III. Recursive Averaging

The first refinement of averaging to introduce is the recursive equations for an average[4,5]:

$$\begin{aligned}
\bar{x}_1 &= x_1 \\
\sigma_{\bar{x}_1} &= r \\
\bar{x}_{i+1} &= \frac{i}{i+1}\bar{x}_i + \frac{1}{i+1}x_{i+1} = \bar{x}_i + \frac{1}{i+1}(x_{i+1} - \bar{x}_i) \\
\sigma_{\bar{x}_{i+1}}^2 &= \left(\frac{i}{i+1}\right)\sigma_{\bar{x}_i}^2
\end{aligned} \qquad (3)$$

which can be justified to the students simply by pointing out that it is entirely equivalent to the ordinary equation for an average. We also emphasize that equation (3) does not require storage of all measurements. The students should note at this point that successive measurements have less and less impact on the average, as $i/(i+1)$ approaches one and $1/(i+1)$ approaches zero.

This point can be emphasized with a plot of the estimate of the resistance as a function of measurement number as shown in Figure 1.
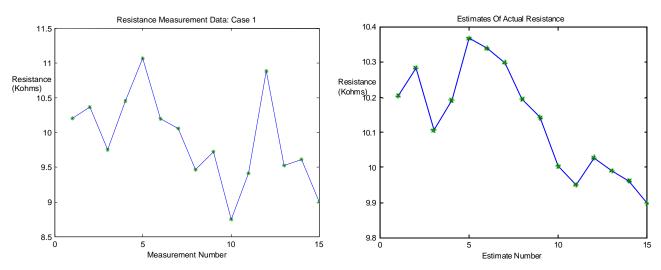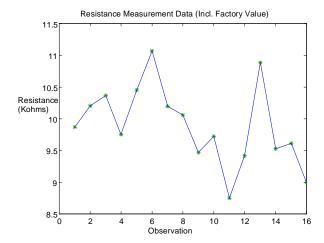

Figure 1: Plot of Estimated Value of a 10-kohm Resistor vs. Measurement Number

IV. Reformulating the Recursion

The next step is to introduce the students to a more general form of the recursive averaging formula by first introducing them to the need for something different. We do this by showing the same resistance measurements as before, but with an initial measurement described as a "factory value" with smaller uncertainty than the other measurements (0.1 kohm vs. 0.5 kohm). Using equations (3) to compute the average produces results that do not make sense: although the final estimate is close to the factory value, the final uncertainty is greater than the uncertainty of the factory value in Figure 2.

Using the equations (3) the final estimate is 9.8962 kohms and the final estimate uncertainty is 0.1250 kohms. The factory value was 9.86 kohm with an uncertainty of 0.1 kohm

Figure 2: Plot of Resistance Value vs. Measurement Number

Once the students are shown the need for new equations, the equations are presented and discussed in some detail.

$$b_i = \frac{\sigma^2_{\bar{x}_i}}{\sigma^2_{\bar{x}_i} + r^2}$$

$$\bar{x}_{i+1} = \bar{x}_i + b_i\left(x_i - \bar{x}_i\right) \tag{4}$$

$$\sigma^2_{\bar{x}_{i+1}} = \left(1 - b_i\right)\sigma^2_{\bar{x}_i}$$

With these equations the student is introduced to the filter gain (b_i), which is a function of the current estimate uncertainty and the uncertainty of the newest measurement. It is valuable to demonstrate that when all the measurements have the same uncertainty, equation (4) reduces to equation (3), and also to discuss how the equations deal with extreme cases: very large or very small measurement uncertainties. The students will see that in both extreme cases, the equations produce results that make sense (either rejection of the new (bad) measurement in favor of the old estimate or the reverse).

Also, at this point, the idea of "covariance simulation" can be introduced; the uncertainty of the final estimate can be predicted before the measurements are taken, using only knowledge of the accuracies of the measurements. A plot, Figure 3, of the steadily declining estimation uncertainty helps make all these concepts clearer.
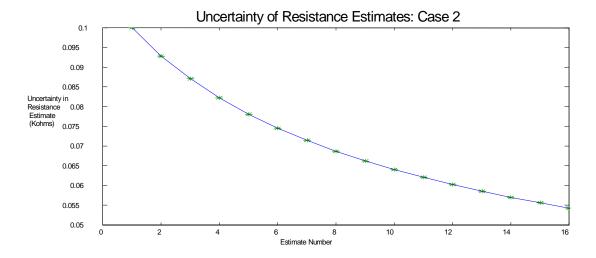

Figure 3: Plot of Uncertainty of Resistance Estimate vs. Measurement Number

V. Introducing Dynamics

Next the student is introduced to estimation of a dynamic system. A simple one-variable example such as an RC circuit is used, and the student is shown what happens when equation (4) is used to estimate the current vs. time in a circuit due to an initial charge on the capacitor. The obviously bad results of Figure 4 that arise provide motivation for the introduction of new equations, in which the dynamics of the system are "built-in."
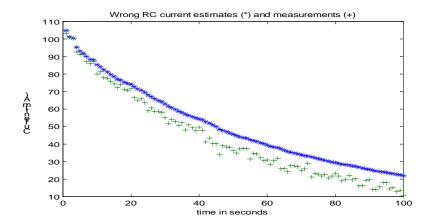
Figure 4: Estimates of Current in an RC Circuit vs. Time in Seconds

Although complicated, it's not hard to see how the non-dynamic equations are transformed into dynamic ones.

$$\dot{x} = f\, x, \quad x_{(o)} = x_o \qquad \text{Continuous-time system equations}$$

$$x_i^+ = \phi x_i^- \qquad \phi = e^{f\Delta t} \qquad \text{Discrete-time system equations}$$

$$\sigma_{x_i}^+ = \phi \sigma_{x_i}^- \phi \qquad \Delta t \text{ is the time between measurements}$$

$$(5)$$

$$b_i = \frac{\sigma^{2\,+}_{x_i}}{\sigma^{2\,+}_{x_i} + r^2}$$

$$\overline{x}_{i+1}^- = \overline{x}_i^+ + b_i\left(\overline{x}_i - \overline{x}_i^+\right)$$

$$\sigma^{2}_{xi}{}^- = \left(1 - b_i\right)\sigma^{2}_{\overline{x}_i}{}^+$$

When these equations are introduced, the instructor can then introduce the term "innovation" for the quantity $\left(\overline{x}_i - \overline{x}_i^+\right)$, describing it as the difference between what the filter expects the next measurement to be, based on past measurements and known dynamics, and what the next measurement actually is. MATLAB code for demonstrating how the filter works using both the wrong and the right formulation of the filter is included in the Appendix. Also, the students are reminded of the original averaging assumptions; two of them (constancy of the underlying phenomenon being estimated and constancy of the measurement accuracy) have now been discarded. Figure 4 shows the estimates of the currents vs. time using the dynamic equations (5).
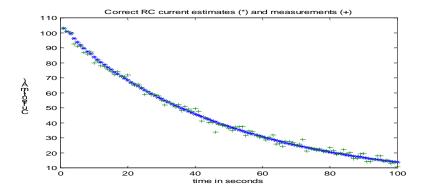
Figure 5: Estimates of Current in RC Circuit vs. Time in Seconds

VI. System Noise

Here, the students are again shown a case for which the previous equations fail to produce reasonable results. This time, it's the same RC circuit, with random noise driving the system. By introducing measurement noise earlier and system noise now, a clear distinction is made between system noise, which results in changes to the behavior of the system, and measurement noise, which does not affect the system itself, but only the observations of the system. This distinction can be emphasized by showing the students a plot of the simulated RC circuit outputs without the simulated "measurement noise". The system noise causes the current in the circuit not to decay exponentially, but rather to wander randomly both above and below zero. When the Kalman equations above are used to estimate this system, the filter estimates stop tracking the measurements as soon as the filter's uncertainty in its estimates drops to zero as shown in Figure 6.
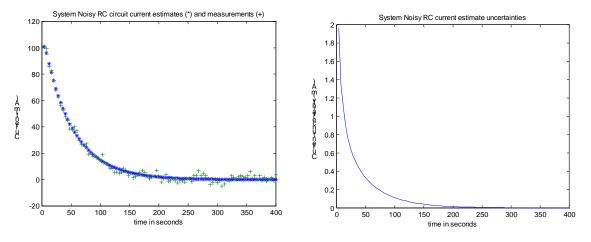

Figure 6: Estimates of RC Circuits Current Using Wrong Equations

At this point the students can guess that the some change to the uncertainty equation is needed to prevent the filter from "locking in" on a single answer that is really continuously changing, so they are ready to be introduced to these discrete equations:

$$\sigma_{\bar{x}_i}^{+} = \phi \sigma_{\bar{x}_i}^{-} \phi + q$$

$$q = \int_{0}^{\Delta t} e^{f\tau} n\, e^{f\tau}\, d\tau$$

Since the students have seen what happens when the filter's uncertainty about the system's state grows too small, they are ready to understand **n** (in the equation above) as the average rate of change of the system state due to system noise. Showing the performance of the filter when system noise is taken into account demonstrates that in a system with system noise, estimate uncertainty will not generally approach zero, and that the estimates now track the measurements as shown in Figure 7.
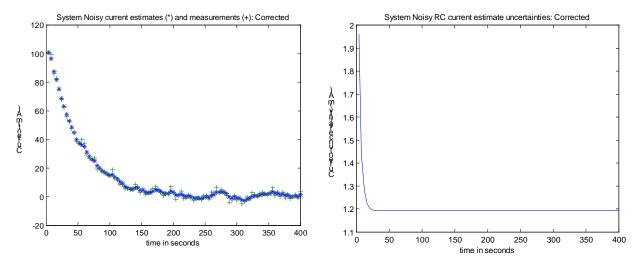

Figure 7: Estimates Using the Correct Equations for the Same Data

VII. Multiple States

Finally, the students are introduced to the idea that systems with multiple states can be estimated with Kalman filters[4,5,8,10]. The "measurement matrix," H, which maps the system state vector to the measurement state vector is the new concept introduced at this stage. Therefore it's helpful to pick as an example a system with more states than are directly measured, and possibly one for which the states are expressed in different units than the measurements.

We chose a ballistic example: an object is shot upwards at an unknown angle. The five states in the system are horizontal position, horizontal velocity, vertical position, vertical velocity, and vertical acceleration; the system matrix is given by

$$F = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

There are frequent measurements of vertical position (with measurement errors of about 10 feet), horizontal velocity (with measurement errors of about 100 knots), and horizontal position (with measurement errors of about 1000 feet); the other states are unmeasured. Therefore the measurement matrix is:

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0.6818 & 0 & 0 \end{bmatrix}$$

Note that the measurement matrix converts the measurement units (knots) to the state transition matrix units (feet/sec). The measurement uncertainty matrix is

$$R^2 = \begin{bmatrix} 1000^2 & 0 & 0 \\ 0 & 100^2 & 0 \\ 0 & 0 & 10^2 \end{bmatrix}$$

To demonstrate the effectiveness of the Kalman filter we also add large system noise due to winds, and huge errors in the initial state estimates (MATLAB code for this example is in the Appendix). The measurements are shown in Figure 7.
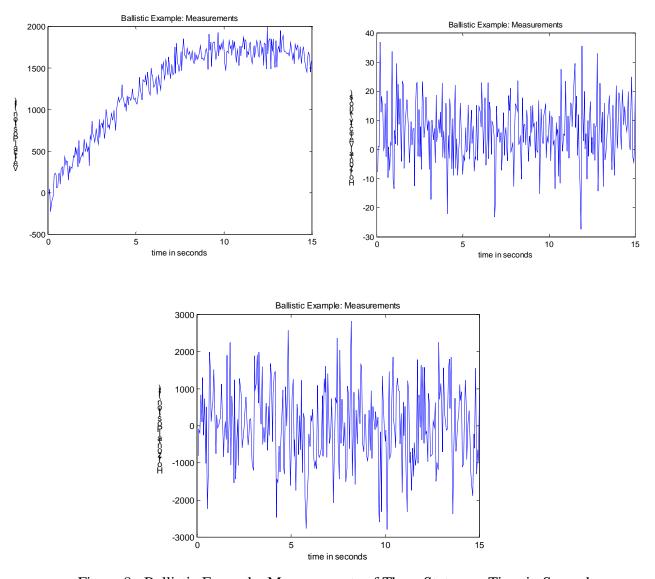
Figure 8: Ballistic Example: Measurements of Three States vs. Time in Seconds

All the Kalman equations need to be converted to matrix form as shown in equation (6) and since each equation is analogous to the single state equivalent the students should not be overwhelmed. As a reminder that all the scalar quantities have become matrices, all the lower case symbols are changed to uppercase.

$$\dot{X} = FX \qquad\qquad \text{Continuous-time system equation}$$

$$X_i^+ = \Phi X_i^- \quad \Phi = e^{F\Delta t} \qquad \text{Discrete-time system equations}$$

$$B_i = \Sigma_{\bar{x}_i}^+ H^T (H\Sigma_{\bar{x}_i}^+ H^T + R^2)^{-1}$$

$$X_{i+1}^- = X_i^+ + B_i(Y_i - HX_i^+) \tag{6}$$

$$Q = \int_0^{\Delta t} e^{F\tau} N \; e^{F\tau} \; d\tau$$

$$\Sigma_{\bar{x}_i}^+ = \Phi \sigma_{\bar{x}_i}^- \Phi^T + Q$$

$$\Sigma_{\bar{x}_{i+1}}^- = (I - B_i)\Sigma_{\bar{x}_i}^+$$

The superscript T indicates the transpose of the matrix.

Students in general are impressed at how well the filter can estimate unmeasured states (see Figure 9) even in the presence of poor initial estimates and very high measurement noise.
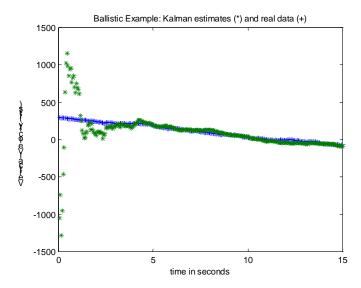

Figure 9: Ballistic Example: Estimates of Vertical Velocity (an unmeasured state) vs. Time in Seconds

Figure 10 shows the student the whole Kalman loop[4,5,10], and emphasizes how much elaboration the original "averaging" concept has received.
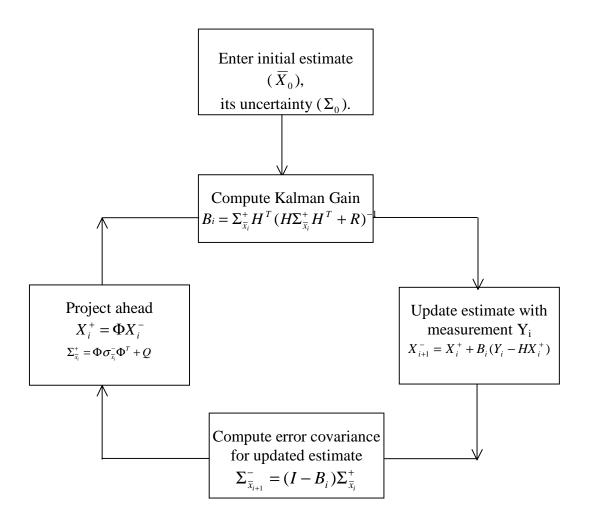
Figure 10: Kalman Filtering Loop

VIII. Summary

Kalman filters provide a convenient framework for introducing students to the concepts of instrument noise, system noise, and system models. The approach develops the filter equations by the use of examples of increasing complexity. The use of "real world" examples captures the interest of the students. This approach has been used in a linear systems course for first semester junior electrical engineering students. The lesson and computer code has motivated students to investigate the use of Kalman filters for estimation of systems that are familiar to them.

Bibliography

1. Kalman, R.E, "A New Approach to Linear Filterings and Prediction Problems", Journal of Basic Engineering, The Transactions of the American Society of Mechanical Engineers, Vol. 83, No. 1, March 1960.
2. Balakrishnan, A.V. "Kalman Filtering Theory", Optimization Software, Inc., 1984.
3. Bozic, S.M. "Digital and Kalman Filtering: An Introduction to Discrete Time Filering and Optimum Linear Estimation", Halsted, 1979.
4. Brown, R.G. "Introduction to Random Signal Analysis and Kalman Filtering", Wiley, 1983.
5. Brown, R.G. & Hwang, P.Y.C. "Introduction to Random Signals and Applied Kalman Filterings (with MATLAB exercises and solutions)", 3rd Edition, Wiley, 1997.
6. Gelb, A. "Applied Optimal Estimation", The MIT Press, Cambridge, Massachusetts 1974.
7. Grewal, M. and Andrews, A. "Kalman Filtering: Theory and Practice, 1st Edition, Prentice Hall, 1993.
8. Helstrom, C. "Probability and Stochasitic Processes for Engineers", 2nd Edition, MacMillan Publishing Company, 1991.
9. Levy, L. "Applied Kalman Filtering", Navtech Seminars, Alexandria VA, July 1997.
10. Picinbono, B. "Random Signals and Systems", Prentice Hall Signal Processing Series, 1993.
11. Sorenson, H.W. "Kalman Filtering: Theory and Application", IEEE Press, 1985.
12. Viniotis, Y. "Probability and Random Processes for Electrical Engineers", McGraw-Hill, 1998.
13. Cramer, H. "Mathematical Methods of Statistics", Princeton University Press, 1946.

Appendix

MATLAB Code for all examples used in this paper

```
clear
randn('seed',130);

ii=1:15;
x=10+randn(1,15)*.5;
disp('15 samples')
disp(x);pause
axis([1 15 9 11]);plot(ii,x,ii,x,'*');title('Resistance Measurement Data: Case 1')
ylabel('Resistance (ohms)')
xlabel('Measurement Number'); ;pause

disp('mean is')
disp(mean(x));pause
disp('std is')
disp(std(x)) ;pause
disp('uncertainty of estimate is')
```

```
disp(.5/sqrt(15))
axis;
hist(x,3) ;title('Histogram of Measurement Data')
ylabel('Resistance (ohms)');pause

p=.5^2;xest(1)=x(1);
for i=2:15
    B=p(i-1)/((.5^2)+p(i-1));
    xest(i)=xest(i-1)+B*(x(i)-xest(i-1))  ;
     p(i)=(1-B)*p(i-1);
end
xest
axis([1 15 9 11]);
plot(ii,xest,ii,xest,'*');
title('Estimates Of Actual Resistance')
ylabel('Resistance (ohms)');
xlabel('Estimate Number');pause

axis([1 15 0 .5]);
plot(ii,sqrt(p),ii,sqrt(p),'*');
title('Uncertainty of Resistance Estimates')
ylabel('Uncertainty in Resistance Estimate (ohms)')
xlabel('Estimate Number');
pause

ii=1:16;
x2=x;
x2=[10+randn*.1 x];
disp('new data');
disp(x2)        ;pause
axis([1 16 9 11]);plot(ii,x2,ii,x2,'*');title('Resistance Measurement Data (Incl. Factory Value)')
ylabel('Resistance (ohms)')
xlabel('Observation Number') ;pause

disp('the wrong mean is')
mean(x2)
disp('the wrong estimate uncertainty is')
.5/sqrt(16)
disp('?')
pause;

R=[.5*ones(1,15)].^2;
p2=.1^2;xest2(1)=x2(1);
for i=2:16
    xest2(i)=xest2(i-1)+p2(i-1)/(R(i-1)+p2(i-1))*(x2(i)-xest2(i-1))         ;
```

```matlab
    p2(i)=((R(i-1)^2)*p2(i-1))/((R(i-1)^2)+p2(i-1));
end
axis([1 16 9 11]);plot(ii,xest2,ii,xest2,'*');title('Estimates Of Actual Resistance: Case 2')
ylabel('Resistance (ohms)')
xlabel('Estimate Number');
pause

axis([1 16 0 .5]);
plot(ii,sqrt(p2),ii,sqrt(p2),'*');
title('Uncertainty of Resistance Estimates: Case 2')
ylabel('Uncertainty in Resistance Estimate (ohms)')
xlabel('Estimate Number');;pause
disp(['the right final estimate is ' num2str(xest2(16))]);
disp(['the right final estimate uncertainty is ' num2str(sqrt(p2(16)))])

x3=zeros(1,100);
x3(1)=100;
F=-.02;
dt=1;R=2^2;
phi=exp(F*dt);
for i=1:99
    x3(i+1)=phi*x3(i);
end
y=x3+sqrt(R)*randn(size(x3));
axis([0 100 0 100]);
plot(y);
title('RC circuit current measurements');
ylabel('Current (mA)');
xlabel('time in seconds');
pause

R=[2*ones(1,100)].^2;
p3w=10^2;xest3w(1)=105;
for i=2:100
    xest3w(i)=xest3w(i-1)+p3w(i-1)/(R(i-1)+p3w(i-1))*(y(i)-xest3w(i-1))        ;
    p3w(i)=((R(i-1)^2)*p3w(i-1))/((R(i-1)^2)+p3w(i-1));
end
plot(1:100,xest3w,'*',1:100,y,'+');
title('Wrong RC current estimates (*) and measurements (+)')
ylabel('Current (mA)')
xlabel('time in seconds');
pause

axis([0 100 0 10]);
plot(1:100,sqrt(p3w));
```

```
title('Wrong RC current estimate uncertainties');
xlabel('time in seconds');
ylabel('Current uncertainty (mA)');
pause

R1=2^2;
X0=105;
P0=10^2;
Q=0;
n=100;
R2=2^2;
yprime=[1:100;ones(1,100);y];
[xx,pp,inn,res]=kalman4(F,dt,n,1,P0,X0,R1,R2,yprime,0,Q,0);
axis([0 100 0 100]);plot(1:100,xx,'*',1:100,y,'+');
title('Correct RC current estimates (*) and measurements (+)')
ylabel('Current (mA)')
xlabel('time in seconds'); ;pause

axis([0 100 0 3]);
plot(1:100,sqrt(pp));
title('Correct RC current estimate uncertainties');
xlabel('time in seconds');
ylabel('Current uncertainty (mA)');
pause

x4=zeros(1,100);
x4(1)=100;Q=1^2;
F=-.02;
dt=4;R=2^2;t=4:4:400;
phi=exp(F*dt);
for i=1:99
    x4(i+1)=phi*x4(i)+randn*sqrt(Q);
end
y=x4+sqrt(R)*randn(size(x4));
axis([0 400 -10 100]);
plot(t,y);
title('RC circuit current measurements: With System Noise');
ylabel('Current (mA)');
xlabel('time in seconds');
pause
yprime=[1:100;ones(1,100);y];
[xx2w,pp2w,inn2w,res2w]=kalman4(F,dt,n,1,P0,X0,R1,R2,yprime,0,0,0);
axis([0 400 -10 100]);
plot(t,xx2w,'*',t,y,'+');
title('System Noisy RC circuit current estimates (*) and measurements (+)');
```

```
ylabel('Current (mA)');
xlabel('time in seconds');
pause

axis([0 400 0 3]);plot(t,sqrt(pp2w));title('System Noisy RC current estimate uncertainties');
xlabel('time in seconds');
ylabel('Current uncertainty (mA)');
pause
[xx2c,pp2c,inn2c,res2c]=kalman (F,dt,n,1,P0,X0,R1,R2,yprime,0,Q,0);

axis([0 400 -10 100]);
plot(t,xx2c,'*',t,y,'+');
title('System Noisy current estimates (*) and measurements (+): Corrected');
ylabel('Current (mA)');xlabel('time in seconds');
pause

axis([0 400 0 3]);
plot(t,sqrt(pp2c));
title('System Noisy RC current estimate uncertainties: Corrected');
xlabel('time in seconds');
ylabel('Current uncertainty (mA)');
pause

F=zeros(5,5);F(1,3)=1;F(2,4)=1;F(4,5)=1;
dt=.05;t=.05:.05:300*.05;
x=zeros(5,300);
phi=expm(F*dt);
R=diag([1000 100 10])^2;
Q=zeros(5,5);Q(3,3)=.1^2;Q(4,4)=3^2;
h=[1 0 0 0 0; 0 1 0 0 0;
   0 0 3600/5280 0 0];
x(:,1)=[0 0 10 300 -32.2]';
for i=1:299
    x(:,i+1)=phi*x(:,i)+randn(5,1).*diag(sqrt(Q));
end
y=h*x;
y(1,:)=y(1,:)+sqrt(R(1,1))*randn(size(y(1,:)));
y(2,:)=y(2,:)+sqrt(R(2,2))*randn(size(y(2,:)));
y(3,:)=y(3,:)+sqrt(R(3,3))*randn(size(y(2,:)));
plot(t,y(2,:));title('Ballistic Example: Measurements')
ylabel('Vertical Position (ft)');
xlabel('time in seconds');
pause

plot(t,y(3,:));
```

```
title('Ballistic Example: Measurements')
ylabel('Horizontal Velocity (knots)');
xlabel('time in seconds');
pause

plot(t,y(1,:));
title('Ballistic Example: Measurements')
ylabel('Horizontal Position (ft)');
xlabel('time in seconds');
pause

yprime=[1:300;ones(1,300);y];
X0=1000*randn(5,1)+x(:,1);P0=diag([1000*ones(1,5)])^2;
[xxballk,ppballk,innballk,resballk]=kalman(F,dt,300,h,P0,X0,R,yprime,0,Q,0);

plot(t,x(1,:),'+',t,xxballk(1,:),'*');
title('Ballistic Example: Kalman estimates (*) and real data (+)')
ylabel('Horizontal Position (ft)')
xlabel('time in seconds');
pause;

plot(t,x(2,:),'+',t,xxballk(2,:),'*');
title('Ballistic Example: Kalman estimates (*) and real data (+)')
ylabel('Vertical Position (ft)')
xlabel('time in seconds');
pause;

plot(t,x(3,:),'+',t,xxballk(3,:),'*');
title('Ballistic Example: Kalman estimates (*) and real data (+)')
ylabel('Horizontal Velocity (ft/s)')
xlabel('time in seconds');
pause;

plot(t,x(4,:),'+',t,xxballk(4,:),'*');
title('Ballistic Example: Kalman estimates (*) and real data (+)')
ylabel('Vertical Velocity (ft/s)')
xlabel('time in seconds');
pause;

plot(t,x(5,:),'+',t,xxballk(5,:),'*');
title('Ballistic Example: Kalman estimates (*) and real data (+)')
ylabel('Vertical Acceleration (ft/s/s)')
xlabel('time in seconds');
```

MATLAB function code for Kalman filter

```
function [x,sigmatime,innov,resid]=kalman(F,dt,N,h,sigma0,X0,R,y,Q);
% General Kalman filter function.
% Inputs:
% phi is the discrete state transition matrix.
% h is the measurement matrix.  h must contain one row for each measured quantity
%      and one column for each state.
% sigma0 is the initial estimate error covariance matrix.
% X0 is the initial estimate.
% R is the measurement noise covariance matrix.
% y is the time series of measurements.
% Q is the system noise covariance matrix.
% Outputs:
% x will hold the time series of filtered estimates.
% sigmatime will hold the time series of estimate variances
% resid contains the estimate residuals.
% innov contains the pre-estimate innovations

phi=expm(F*dt);
% make sure that X0 is a column vector not a row vector.
[i j]=size(X0);
if i<j
  X0=X0';
end
Mmeas=length(R);
% how many states are there in the system?
m=max(size(F));
% initialize the matrix that will hold the time sequence of state variances.
sigmatime=zeros(m,N);
% initialize the matrix that will hold the time sequence of state estimates.
x=zeros(m,N);
innov=zeros(Mmeas,N);
resid=innov;
% the Kalman filter itself
xest=X0;
sigma=sigma0;
for i=1:N
```

```
B=sigma*h'*inv(h*sigma*h'+R);    % Kalman Gain
innov(:,i)=y(:,i)-(h*xest);      % innovation
x(:,i)=xest+B*innov(:,i);        % estimation
xest=phi*x(:,i);                 % update estimate for dynamics
sigma=(eye(max(size(sigma)))-(B*h))*sigma; % update uncertainty
sigmatime(:,i)=diag(sigma);
sigma=phi*sigma*phi'+Q;          % add system noise to the estimate uncertainty.
end
resid=y-h*x;
```

MAURICE F. ABURDENE
Maurice F. Aburdene is the T. Jefferson Miers Professor of Electrical Engineering and Professor of Computer Science at Bucknell University. He has also taught at Swarthmore College, State University of New York at Oswego, and the University of Connecticut. Professor Aburdene was a project engineer and project manager at the Bristol Company, a visiting research scientist at MIT's Laboratory for Information and Decision Systems, and held various summer fellowships with NASA and the Naval Research Laboratory.

ANDREW LOVE
Andrew Love graduated from Bucknell in 1987 with a bachelor's degree in Electrical Engineering and then again in 1989 with a Master's degree. He began work as an Engineer/Analyst at the Johns Hopkins University Applied Physics Laboratory later that year, where he evaluates the performance of submarine inertial navigation systems for the United States Navy (often using Kalman filters). He married fellow Bucknell graduate Pamela Gibson in 1990, and earned another Master's Degree at the Johns Hopkins University evening school.

RAMI ZARROUK
Rami William Zarrouk is a graduate student in Electrical Engineering at Bucknell University. He has a bachelor's degree in Electrical Engineering from Christian Brothers University in Memphis, Tennessee. He is a member of Tau Beta Pi and IEEE. Rami was a recipient of the 1998 Phillip-Morgan Award for best engineering design. He is currently working as a teacher assistant for an introduction to digital systems class.