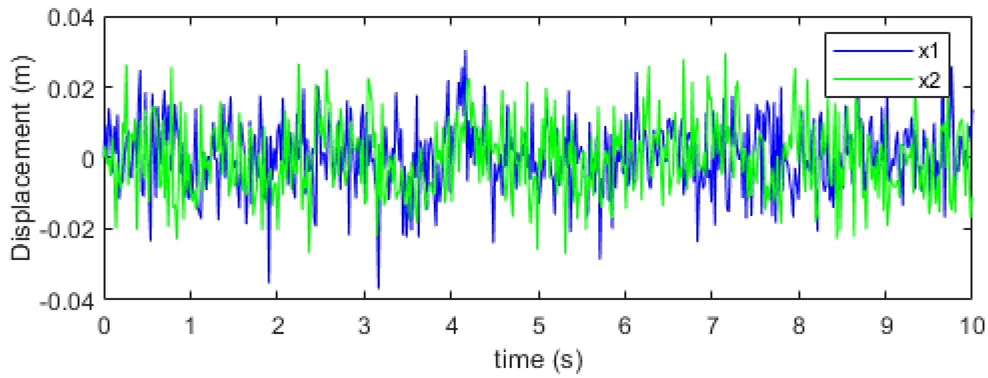
Systematic review of SysID - Expectation-Maximization algorithm

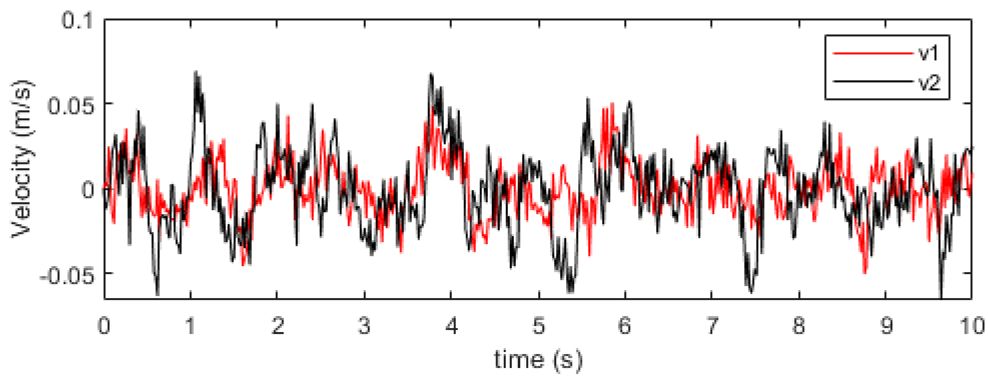
```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Jingxiao Liu
% Inferlab, CEE, CMU
% June, 2019
% Matlab live script for introducing expectation-maximization algorithm
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

5.1 State space model

$$x[k+1] = A\,x[k] + B\,u[k] + \epsilon[k]$$
$$y[k+1] = C\,x[k+1] + \eta[k+1]$$
$$\epsilon[k] \sim N(0, Q)$$
$$\eta[k] \sim N(0, R)$$

```
clc;
clear;
close all;
addpath ../functions/
T = 10; %length of time duration
nt = 500; %number of time stamps
Fs = nt/T;
F = csvread(['../data/ambient.csv']); %load data
X = csvread(['../data/2dof_noP.csv']); %load data
tspan=linspace(0,T,nt);
figure(1)
plot_dv(tspan,X')
```






```
m1=5; %mass 1 [kg]
m2=2; %mass 2 [kg]
k1=200; %spring 1 [N/m]
k2=100; %spring 2 [N/m]
c1=10; % damping coeff 1
c2=10; % damping coeff 2

% system matrices
M=[m1 0;0 m2]; %mass matrix
K=[k1+k2 -k2; -k2 k2]; %stiffness matrix
Damp=[c1+c2 -c2; -c2 c2]; % damping matrix
dt = tspan(2)-tspan(1); % delta time
C = [1,0,0,0;0,0,1,0];
Ac = [0,0,1,0;
     0,0,0,1;
     -K(1,1)/M(1,1),-K(1,2)/M(1,1),-Damp(1,1)/M(1,1),-Damp(1,2)/M(1,1);
     -K(2,1)/M(2,2),-K(2,2)/M(2,2),-Damp(2,1)/M(2,2),-Damp(2,2)/M(2,2)];
A = Ac*dt;
A = expm(A);
Bc = [0,0,0,0;
    0,0,0,0;
    0,0,1/M(1,1),0;
    0,0,0,1/M(2,2)];
B = inv(Ac)*(A-eye(4))*Bc;
```

Partial observation

```
Y = C*X;
```

5.2 Kalman filter

$$
\begin{aligned}
&\widehat{x}[k+1|k] = A\,\widehat{x}[k|k] + B\,u[k]\\
&\Sigma[k+1|k] = A\Sigma[k|k]A^T + Q\\
&K[k+1] = \Sigma[k+1|k]C^T(C\Sigma[k+1|k]C^T + R)^{-1}\\
&\widehat{x}[k+1|k+1] = \widehat{x}[k+1|k] + K[k+1](y[k+1] - C\widehat{x}[k+1|k])\\
&\Sigma[k+1|k+1] = \Sigma[k+1|k] - K[k+1]C\Sigma[k+1|k]
\end{aligned}
$$

5.2.1 Deriving the posteriori estimate covariance matrix

$$
\begin{aligned}
&\Sigma[k|k] = \operatorname{cov}(x[k] - \widehat{x}[k|k]) = \operatorname{cov}[x[k] - (\widehat{x}[k|k-1] + K[k](y[k+1] - C\widehat{x}[k+1|k]))]\\
&= \operatorname{cov}[x[k] - (\widehat{x}[k|k-1] + K[k](C\,x[k] + \eta[k] - C\widehat{x}[k+1|k]))]\\
&= \operatorname{cov}[(I - K[k]C)(x[k] - \widehat{x}[k|k-1]) - K[k]\eta[k]]\\
&= (I - K[k]C)\Sigma[k|k-1](I - K[k]C)^T + K[k]R\,K[k]^T
\end{aligned}
$$

5.2.2 Deriving Kalman gain

We seek to minimize the mean-square error estimator $E\left[\|x[k] - \widehat{x}[k|k]\|^2\right]$, which is quivalent to minimize

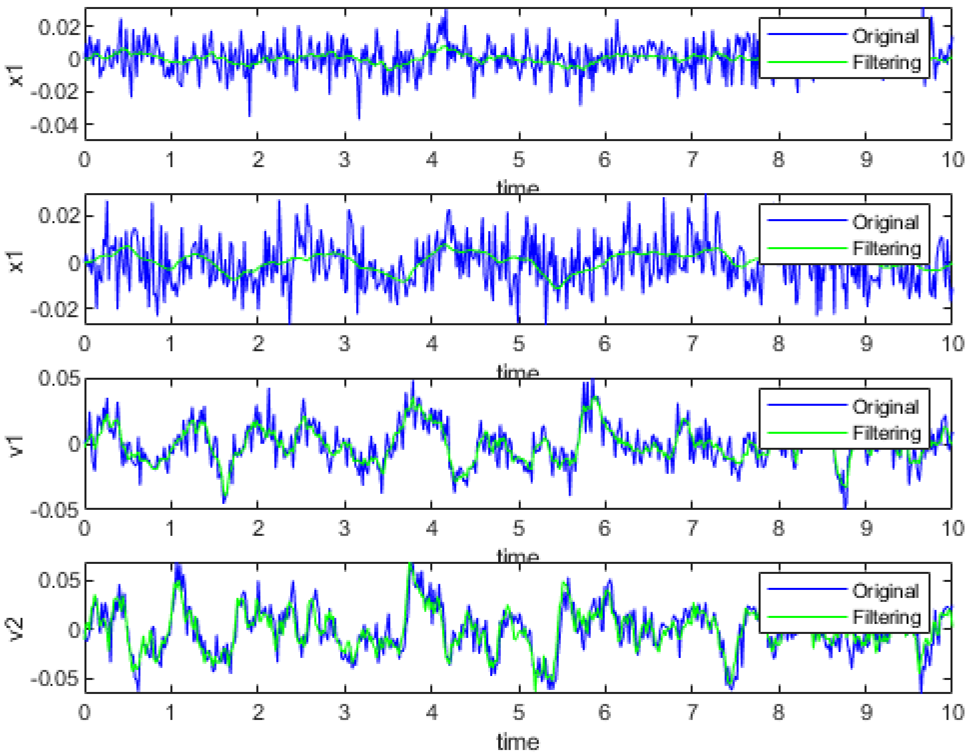
$$
\begin{aligned}
&\Sigma[k|k] = (I - K[k]C)\Sigma[k|k-1](I - K[k]C)^T + K[k]R\,K[k]^T\\
&\frac{\partial}{\partial K[k]}\operatorname{tr}(\Sigma[k|k]) = -2(C\,\Sigma[k|k-1])^T + 2K[k](C\Sigma[k+1|k]C^T + R)
\end{aligned}
$$

Then,

$$
K[k+1] = \Sigma[k+1|k]C^T(C\Sigma[k+1|k]C^T + R)^{-1}
$$

```
Q = diag([1e-6,1e-6,1e-6,1e-6]); % initialize process noise covariance
R = diag([1e-4,1e-4]); % initialize observation noise covariance
xkk = X(:,1);
pkk = B*Q*B';
Xk1k = zeros(4, size(Y,2));
Xkk = zeros(4, size(Y,2));
Pkk = zeros(4, 4, size(Y,2));
Pk1k = zeros(4, 4, size(Y,2));
for i = 1:size(Y,2)
    xk1k = A * xkk + B*[0;0;F(1,i);F(2,i)];
    pk1k = A * pkk * A' + Q;
    K = pk1k*C'*inv(R + C*pk1k*C');
    xkk = xk1k + K * (Y(:,i)-C*xk1k);
    pkk = pk1k - K*C*pk1k;
    Xk1k(:,i) = xk1k;
    Xkk(:,i) = xkk;
    Pk1k(:,:,i) = pk1k;
    Pkk(:,:,i) = pkk;
end
ylabels = {'x1','x1','v1','v2'};
figure(1)
```

```
plot_com(tspan,X,Xkk,ylabels,'Original','Filtering')
```



5.3 Kalman smoothing

5.3.1 Forward pass: Kalman filter computes $(x[k]|y[0:k])$, which is real-time,

5.3.2 Backward pass: computing $(x[k]|y[0:T])$, which is post-processing.

$$\begin{bmatrix} x[k|k] \\ x[k+1|k] \end{bmatrix} = N\left(\begin{bmatrix} E[x[k|k]] \\ E[x[k+1|k]] \end{bmatrix}, \begin{bmatrix} \mathrm{var}(x[k|k]) & \mathrm{cov}(x[k|k], x[k+1|k]) \\ \mathrm{cov}[x[k+1|k], x[k|k]] & \mathrm{var}(x[k+1|k]) \end{bmatrix}\right)$$
$$= N\left(\begin{bmatrix} \hat{x}[k|k] \\ \hat{x}[k+1|k] \end{bmatrix}, \begin{bmatrix} \Sigma[k|k] & \Sigma[k+1|k]A^T \\ A\Sigma[k+1|k] & \Sigma[k+1|k] \end{bmatrix}\right)$$
$$\mathrm{cov}(x[k+1|k], x[k|k]) = A\,\mathrm{cov}(x[k|k], x[k|k]) + \mathrm{cov}(B\,u[k+1], x[k|k]) = A\Sigma[k+1|k]$$

Then,

$$\left(x[k|k] | x[k+1|k] = \tilde{x}[k+1]\right)$$
$$= N\left(\hat{x}[k|k] + \Sigma[k|k]A^T\Sigma[k+1|k]^T\left(\tilde{x}[k+1] - \hat{x}[k+1|k]\right), \Sigma[k|k] - \Sigma[k|k]A^T\Sigma[k+1|k]^{-1}A\Sigma[k|k]\right)$$

We do not know $\tilde{x}[k+1]$, but $\tilde{x}[k+1] \sim x[k+1|0:T]$, thus,

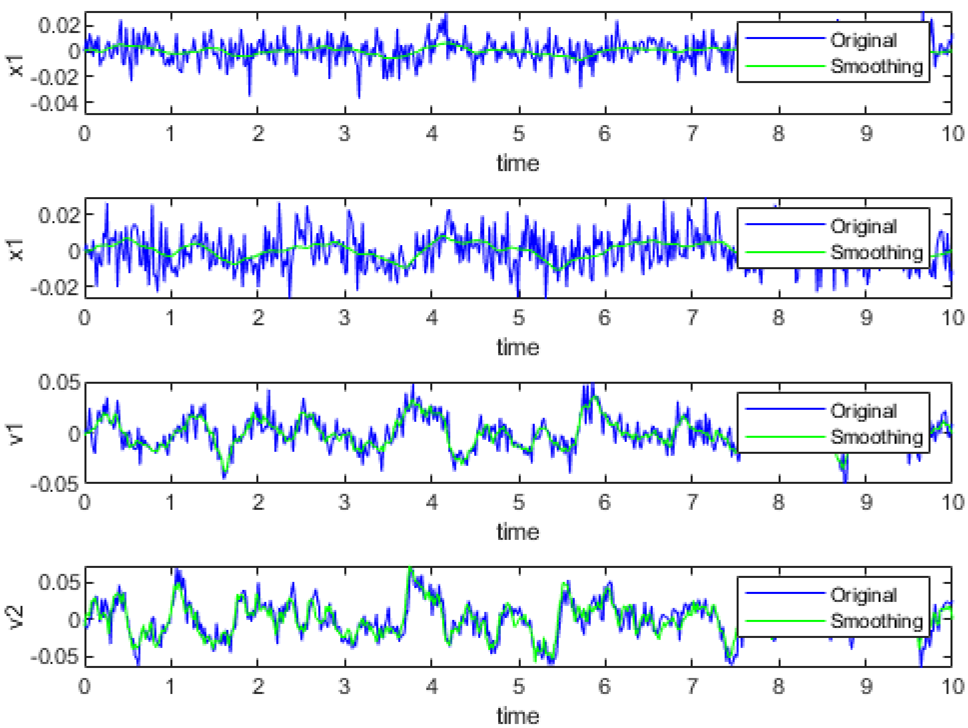
(Law of total expectation)

$$\hat{x}[k|0:T] = E[x[k|0:T]] = E_{x[k+1|0:T]}[E(x[k|k] | x[k+1|k] = x[k+1|0:T])]$$
$$= \hat{x}[k|k] + \Sigma[k|k]A^T\Sigma[k+1|k]^T(\hat{x}[k+1|0:T] - \hat{x}[k+1|k])$$

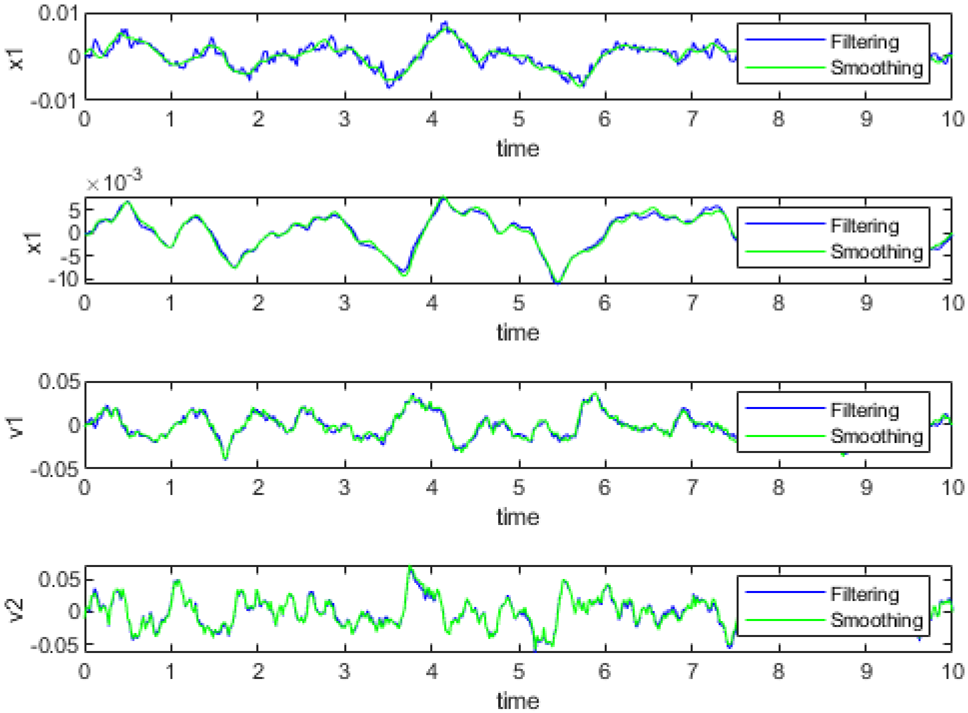
(law of total variance)

$$\Sigma[k|0:T] = \mathrm{Var}[x[k|0:T]]$$
$$= E_{x[k+1|0:T]}[\mathrm{Var}(x[k|k]|x[k+1|k] = x[k+1|0:T])] + \mathrm{Var}_{x[k+1|0:T]}[E(x[k|k]|x[k+1|k] = x[k+1|0:T])]$$
$$= \Sigma[k|k] - \Sigma[k|k]A^T\Sigma[k+1|k]^{-1}A\Sigma[k|k] + \Sigma[k|k]A^T\Sigma[k+1|k]^T\Sigma[k+1|0:T](\Sigma[k|k]A^T\Sigma[k+1|k]^{-1})^T$$

```
XT = zeros(4, size(Y,2));
PT = zeros(4, 4, size(Y,2));
for k=(size(Xk1k,2)-1):-1:1
    L(:,:,k) = Pkk(:,:,k) * A' * inv(Pk1k(:,:,k+1)+diag(ones(1,4)*1e-7));
    XT(:,k) = Xkk(:,k) + L(:,:,k) * ...
        (XT(:,k+1) - Xk1k(:,k+1));
    PT(:,:,k) = Pkk(:,:,k) + L(:,:,k) * ...
        (PT(:,:,k+1) - Pk1k(:,:,k+1)) * L(:,:,k)';
end
figure(3)
plot_com(tspan,X,XT,ylabels,'Original','Smoothing')
```




```
figure(4)
plot_com(tspan,Xkk,XT,ylabels,'Filtering','Smoothing')
```



5.4 Log-likelihood function

$$L(A,B,C,u[0:T],Q,R|x[0:T],y[0:T]) = \log p(x[0:T],y[0:T]|A,B,C,u[0:T],Q,R)$$

$$= \log \prod_{k=0}^{T} p(x[k]|x[k-1])p(y[k]|x[k])$$

$$= \sum_{k=0}^{T-1} \log p(x[k+1]|x[k]) + \sum_{k=0}^{T} \log p(y[k]|x[k])$$

$$= \frac{T}{2}\log\left|Q^{-1}\right| - \frac{1}{2}\sum_{k=0}^{T-1} \mathrm{tr}[(x[k+1]-\mathrm{Ax}[k]-\mathrm{Bu}[k])^T Q^{-1}(x[k+1]-\mathrm{Ax}[k]-\mathrm{Bu}[k])]$$

$$+\frac{T+1}{2}\log\left|R^{-1}\right| - \frac{1}{2}\sum_{k=0}^{T} \mathrm{tr}[(y[k]-\mathrm{Cx}[k])^T R^{-1}(y[k]-\mathrm{Cx}[k])] + \mathrm{const}$$

5.5 Maximize the expected log-likelihood function

$$\frac{\partial}{\partial A} E[L(A, B, C, u[0:T], Q, R | x[0:T], y[0:T])] \equiv 0$$

$$A = \left(\sum_{k=0}^{T-1} [E[x[k]x[k+1]^T | y[0:T]] - E[B\,u[k]x[k]^T | y[0:T]]] \right) \left(\sum_{k=0}^{T-1} E[x[k]x[k]^T | y[0:T]] \right)^{-1}$$

$$\frac{\partial}{\partial C} E[L(A, B, C, u[0:T], Q, R | x[0:T], y[0:T])] \equiv 0$$

$$C = \left(\sum_{k=0}^{T} [E[y[k]x[k]^T | y[0:T]] \right) \left(\sum_{k=0}^{T} E[x[k]x[k]^T | y[0:T]] \right)^{-1}$$

$$\frac{\partial}{\partial Q^{-1}} E[L(A, B, C, u[0:T], Q, R | x[0:T], y[0:T])] \equiv 0$$

$$Q = \frac{1}{T} \left\{ \sum_{k=0}^{T-1} E[(x[k+1] - \mathrm{Ax}[k] - \mathrm{Bu}[k])^T (x[k+1] - \mathrm{Ax}[k] - \mathrm{Bu}[k]) | y[0:T]] \right\}$$

$$\frac{\partial}{\partial R^{-1}} E[L(A, B, C, u[0:T], Q, R | x[0:T], y[0:T])] \equiv 0$$

$$R = \frac{1}{T+1} \left\{ \sum_{k=0}^{T} E[(y[k] - \mathrm{Cx}[k])^T (y[k] - \mathrm{Cx}[k]) | y[0:T]] \right\}$$

where

$$E(x[k] | y[0:T]) = \hat{x}[k|0:T]$$
$$E(x[k]x[k]^T | y[0:T]) = \Sigma[k|0:T] + \hat{x}[k|0:T]\hat{x}[k|0:T]^T$$
$$E(x[k]x[k+1]^T | y[0:T]) = \hat{x}[k|0:T]\hat{x}[k+1|0:T]^T$$
$$+\Sigma[k|k]A^T\Sigma[k+1|k]^T(\Sigma[k+1|0:T] + (\hat{x}[k+1|0:T] - \hat{x}[k+1|k])\hat{x}[k+1|0:T]^T)$$

```
[A,C,Q,R,llhs] = ss_em(A,C,Q,R,Y,X(:,1),Q);
figure(5)
plot(llhs)
xlabel('Iteration')
ylabel('Log-likelihood')
% [XT, PT, Exx, Ex1x, llh] = kalman_smoother(A,C,Q,R,Y,X(:,1),Q);
% figure(5)
% plot_com(tspan,X,XT,ylabels,'Original','Smoothing')
```