# Pink Noise and Noise Cancellation

## NDSU ECE 463/663

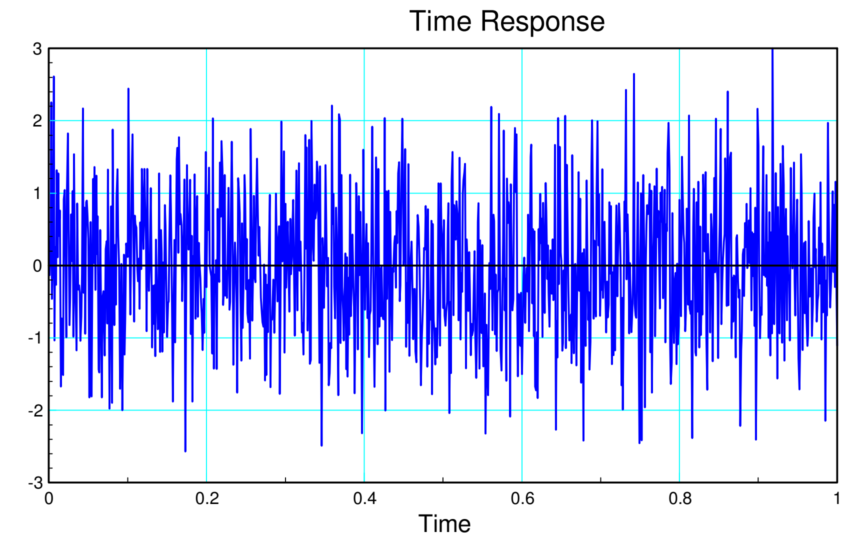
## Lecture #33


## Inst: Jake Glower


Please visit Bison Academy for corresponding lecture notes, homework sets, and solutions

## Problem:

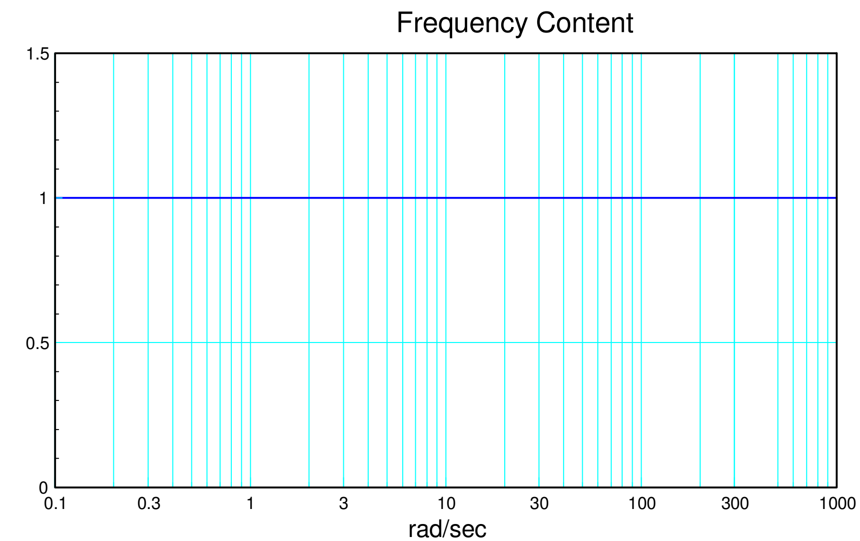
A system has an input disturbance with a distinct spectra. Can you take this information into account?




## White Noise:

White noise is

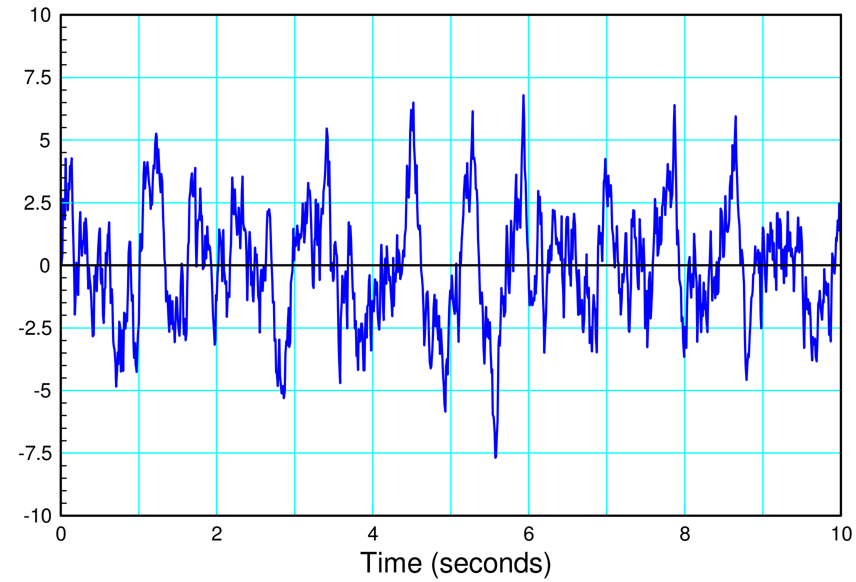
- Gaussian: $x(t) \sim N(\mu, \sigma^2)$
- Uncorrelated
- The energy content is the same everywhere

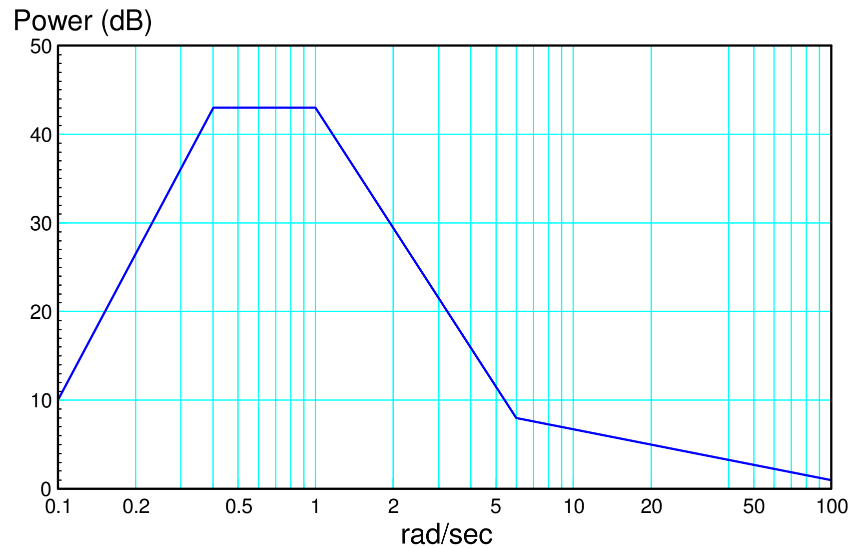


## Pink Noise:

Some disturbances, such as wind, solar heating, etc. are not uncorrelated white noise like this. Instead, their spectrum has a distinct shape.

- The Dryden Spectrum is the approximate power of wind buffeting an airplane
- *from Control Systems Design* by Bernard Friedland



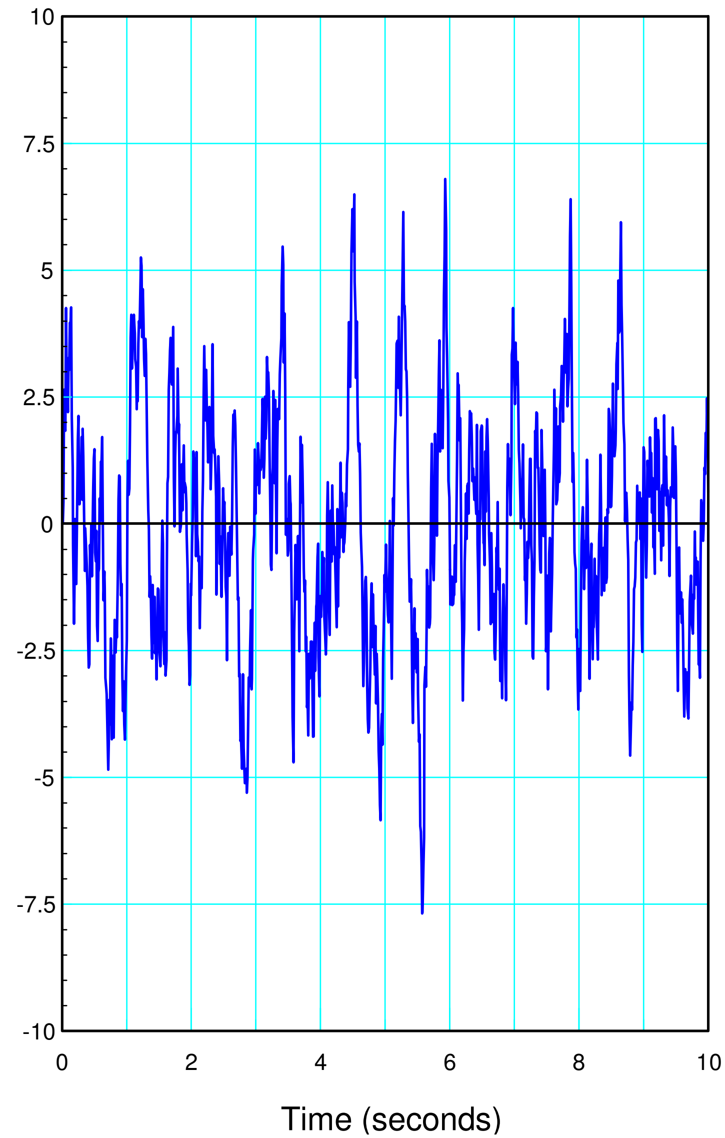
You can approximate this as white noise passed through a band-pass filter:

$$d \approx \left(\frac{100(s+0.1)}{s^2+10s+25}\right)\eta \qquad \eta \sim N(0,1)$$

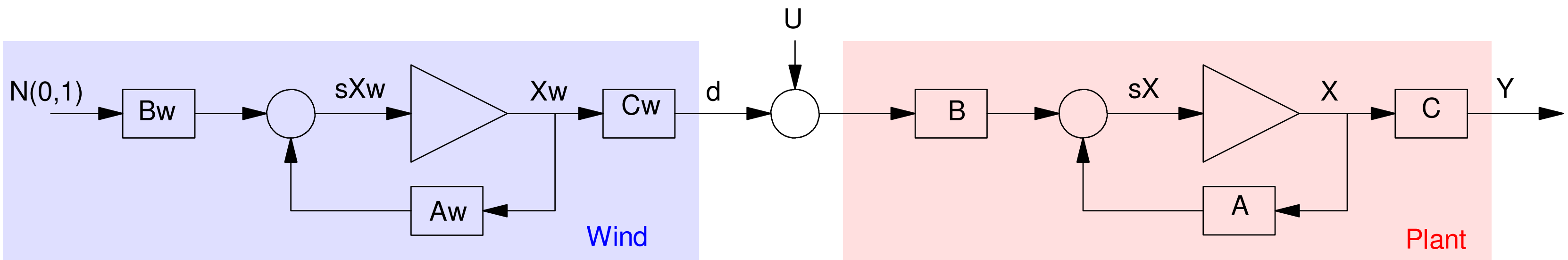
which has the following time response:

```
>> G = tf([100,10],[1,10,25])

Transfer function:
  1000 s + 10
---------------
s^2 + 10 s + 25

>> y = step3(A,B,C,D,t,X0,N);
>> plot(t,N)
>> plot(t,y)
```

This is termed 'pink noise': noise with a spectral content.



Example: Add pink noise to the 4th-order RC filter:



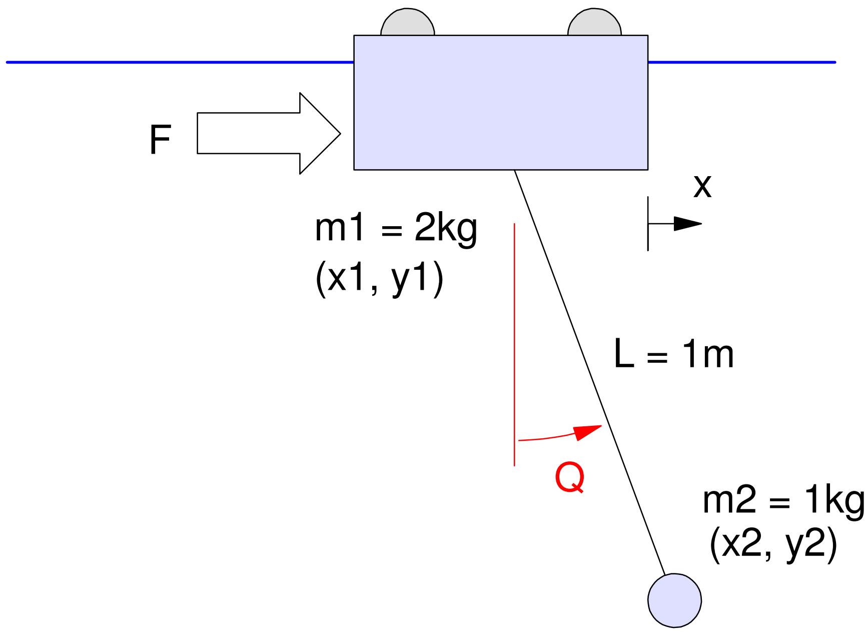

The dynamics of the plant and disturbance with no feedback are:

$$s\begin{bmatrix} X \\ X_w \end{bmatrix} = \begin{bmatrix} A & BC_w \\ 0 & A_w \end{bmatrix}\begin{bmatrix} X \\ X_w \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} U + \begin{bmatrix} 0 \\ B_w \end{bmatrix} \eta$$

# Example: Gantry System.

Assume

- 4th-order Gantry system
- Wind adds to the force input
- Both position and angle are measured




Ref = 0

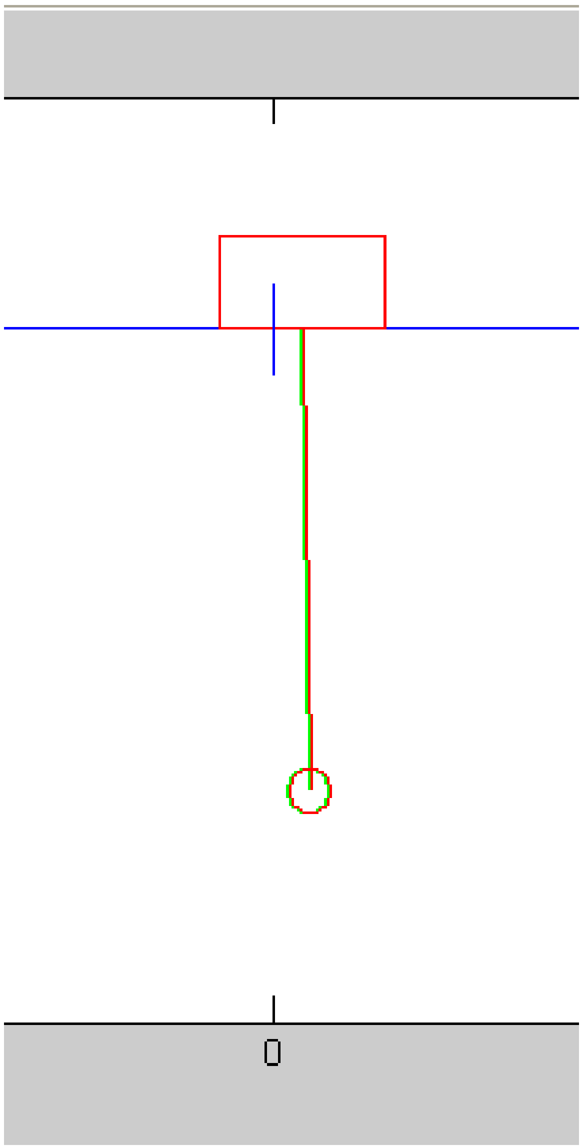
- Try to hold the cart at x = 0

Control Law

- U = -Kx X

# Matlab Simulation

```
% Lecture #33
% Pink Noise
% Gantry plus full-order observer

X = [0;0;0;0];
dX = zeros(4,1);
Ref = 0;
dt = 0.01;
U = 0;
t = 0;
C = [1,0,0,0;0,1,0,0];

% Wind Disturbance
Aw = [-5,0; 1, -5];
Bw = [1;0];
Cw = [100,-450];
Xw = zeros(2,1);

%Observer
Ae = [0,0,1,0;0,0,0,1;0,4.9,0,0;0,-14.7,0,0];
Be = [0;0;0.5;-0.5];
Ce = [1,0,0,0 ; 0,1,0,0];
Xe = X;
H =  lqr(Ae', Ce', diag([1,1,1,1]), 0.01*diag([1,1]))';
y = [];
n = 0;

% Feedback
Kx = [3.1623   -3.8670    5.6097   -1.3269];
Kr = 0;
```

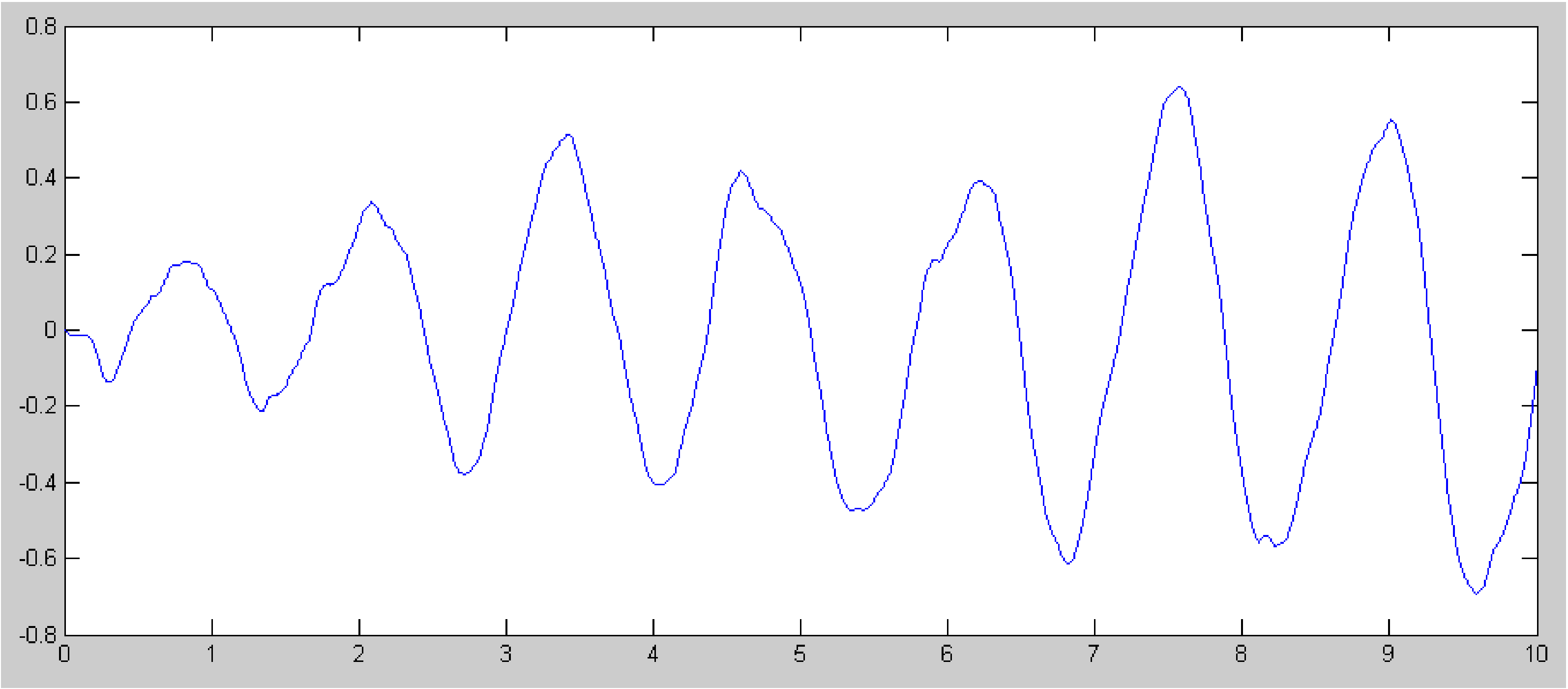


```
while(t < 20)
 U = -Kx*X;
 Ref = 0;
 Wind = Cw*Xw;

 dX = GantryDynamics(X, U + Wind);
 dXe = Ae*Xe + Be*U + H*(C*X - Ce*Xe);
 dXw = Aw*Xw + Bw*randn;

 X = X + dX * dt;
 Xe = Xe + dXe * dt;
 Xw = Xw + dXw * dt;

 t = t + dt;

 n = mod(n+1, 5);
 if(n == 0)
    GantryDisplay3(X, Xe, Ref);
    plot([Ref, Ref],[-0.1,0.1],'b');
 end

 y = [y ; X(1), Xe(1) ];

end

hold off
t = [1:length(y)]' * dt;
plot(t,y);
```

with wind applying a force disturbance on the displacement:

```
W = 1;
V = 0.01;
```

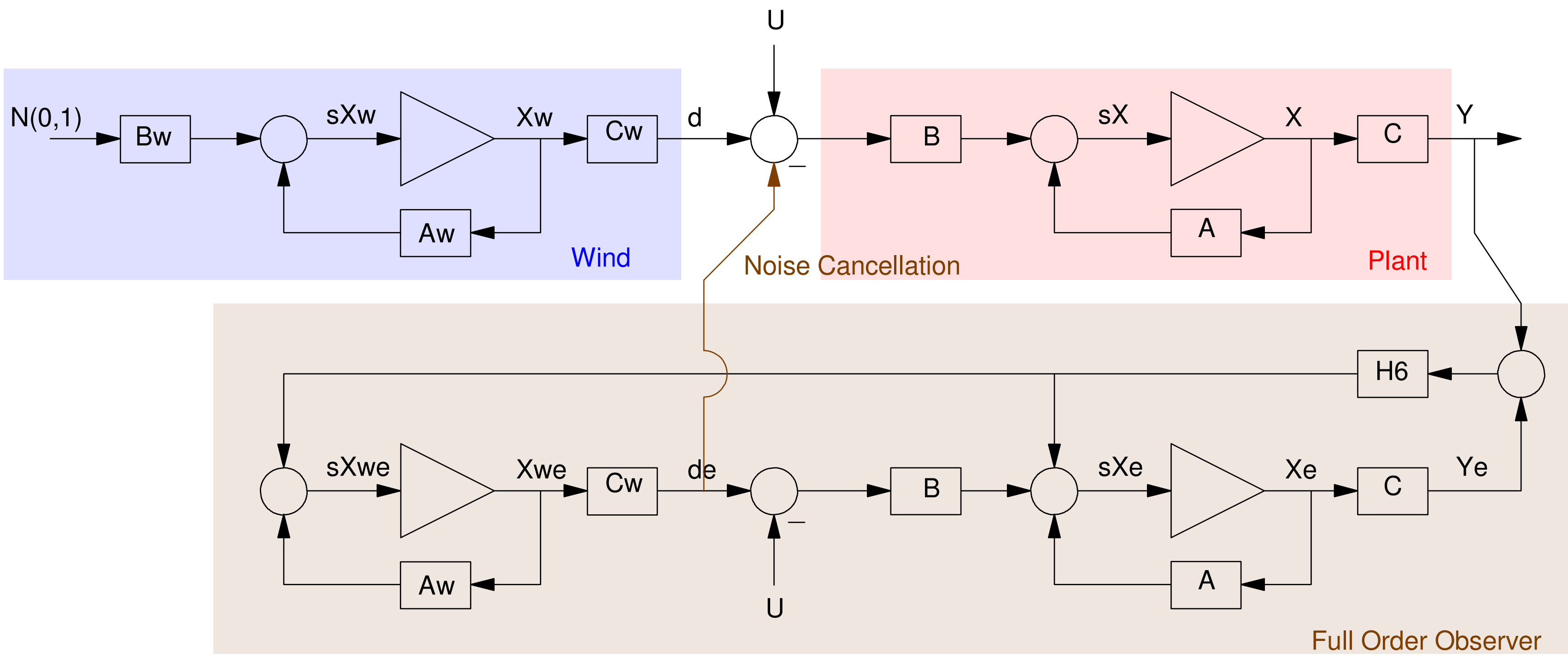
```
t = [0:0.001:10]';
N = size(t);

nu = W*randn(N);
ny = V*randn(N);

A = [0,0,1,0;0,0,0,1;0,4.7,0,0;0,-14.7,0,0];
B = [0;0;0.5;-0.5];
C = [1,0,0,0];

Aw = [-5  0 ; 1  -5];
Bw = [1 ; 0];
Cw = [100 -450];

A6 = [A, B*Cw ; zeros(2,4), Aw];
B6 = [B ; 0; 0];
C6 = [0,0,1,0,0,0];     % velocity
D6 = 0;

F = [0; 0; 0; 0; Bw];

X0 = zeros(6,1);

y = step3(A6, B6, C6, D6, t, X0, nu);

plot(t,y);
```




Velocity of the Gantry Resulting from Wind Disturbances:

# Observer for Noise Cancellation

Now add a full-order observer to estimate the states - including the states of the disturbance:

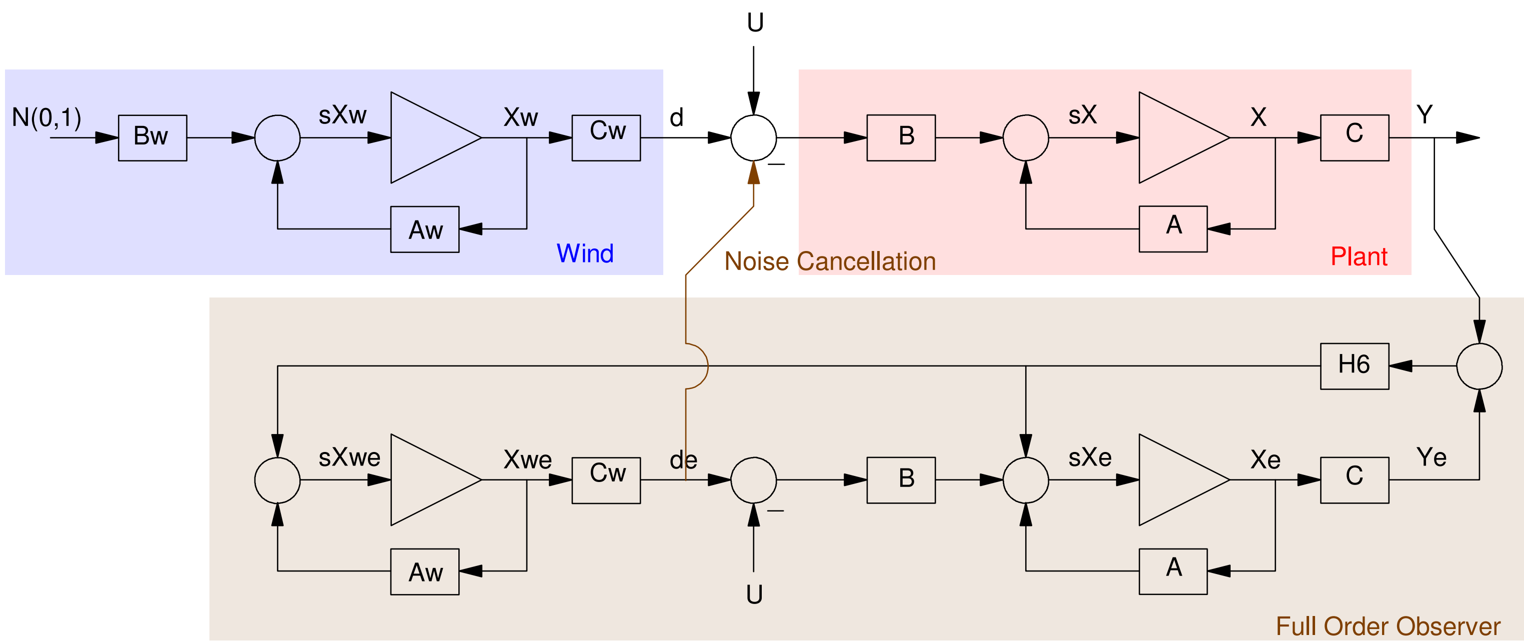



Plant & Pink Noise (wind) & Full-Order Observer

## Plant & Disturbance

$$\begin{bmatrix} sX \\ sX_w \end{bmatrix} = \begin{bmatrix} A & BC_w \\ 0 & A_w \end{bmatrix} \begin{bmatrix} X \\ X_w \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} U$$

$$Y = \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} X \\ X_w \end{bmatrix}$$

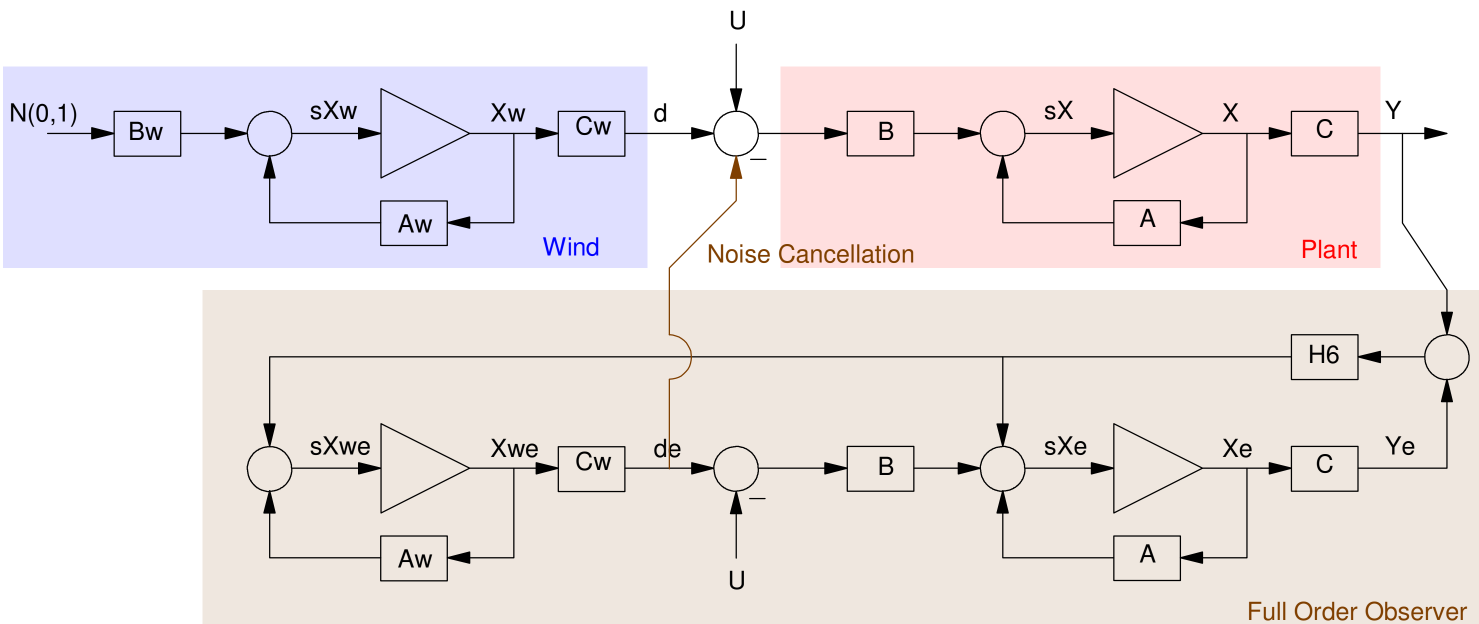


# Design an optimal observer (Kalman filter)

```
Q = W*F*F'*W;
R = V^2;

H6 = lqr(A6', C6', Q, R)'

      33.7
   -  31.3
     567.2
   - 563.7
    0567.0
    3.432.1
```

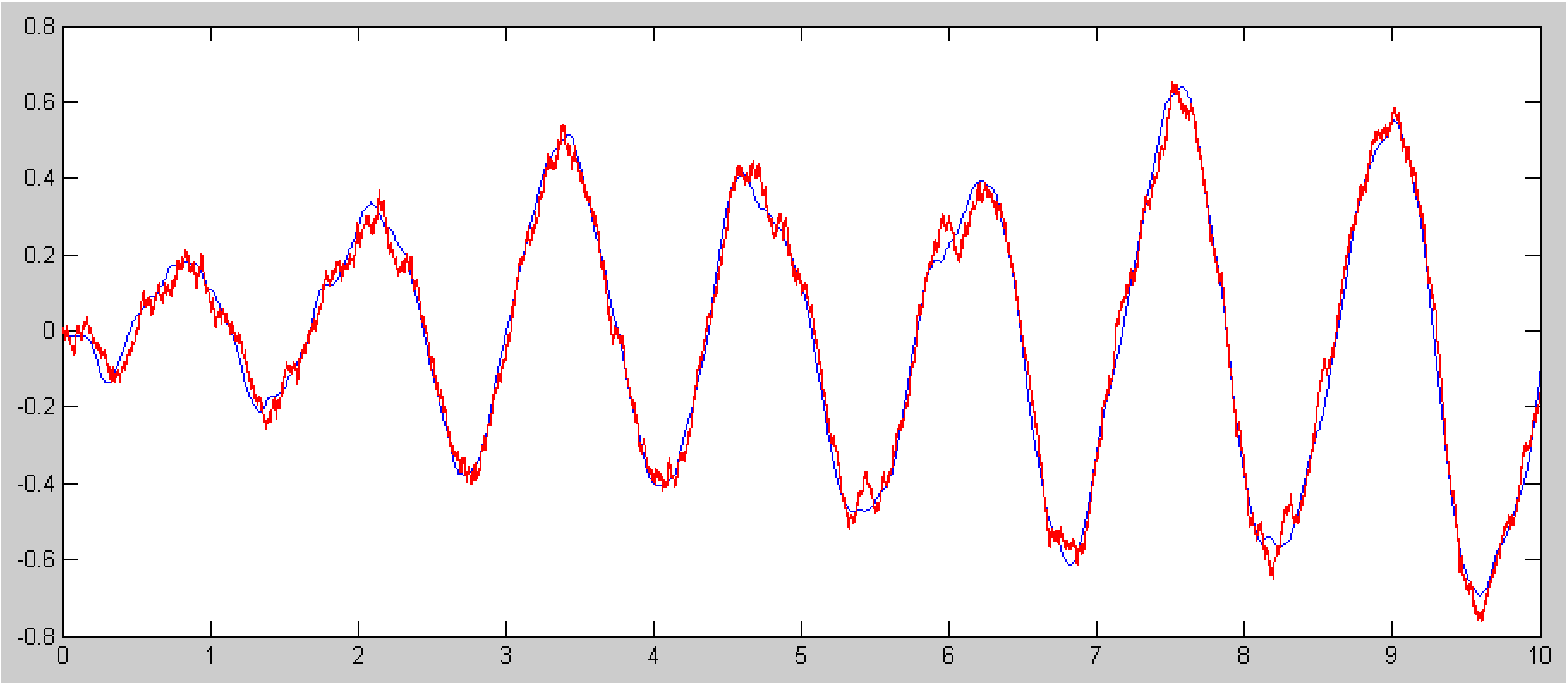
The plant plus the observer is then:

$$\begin{bmatrix} sX \\ sX_w \\ s\hat{X} \\ s\hat{X}_w \end{bmatrix} = \begin{bmatrix} A & BC_w & 0 & 0 \\ 0 & A_w & 0 & 0 \\ H_x C & 0 & A - H_x C & BC_w \\ H_w C & 0 & 0 - H_w C & A_w \end{bmatrix} \begin{bmatrix} X \\ X_w \\ \hat{X} \\ \hat{X}_w \end{bmatrix} + \begin{bmatrix} B \\ 0 \\ B \\ 0 \end{bmatrix} U + \begin{bmatrix} 0 \\ B_w \\ 0 \\ 0 \end{bmatrix} \eta$$

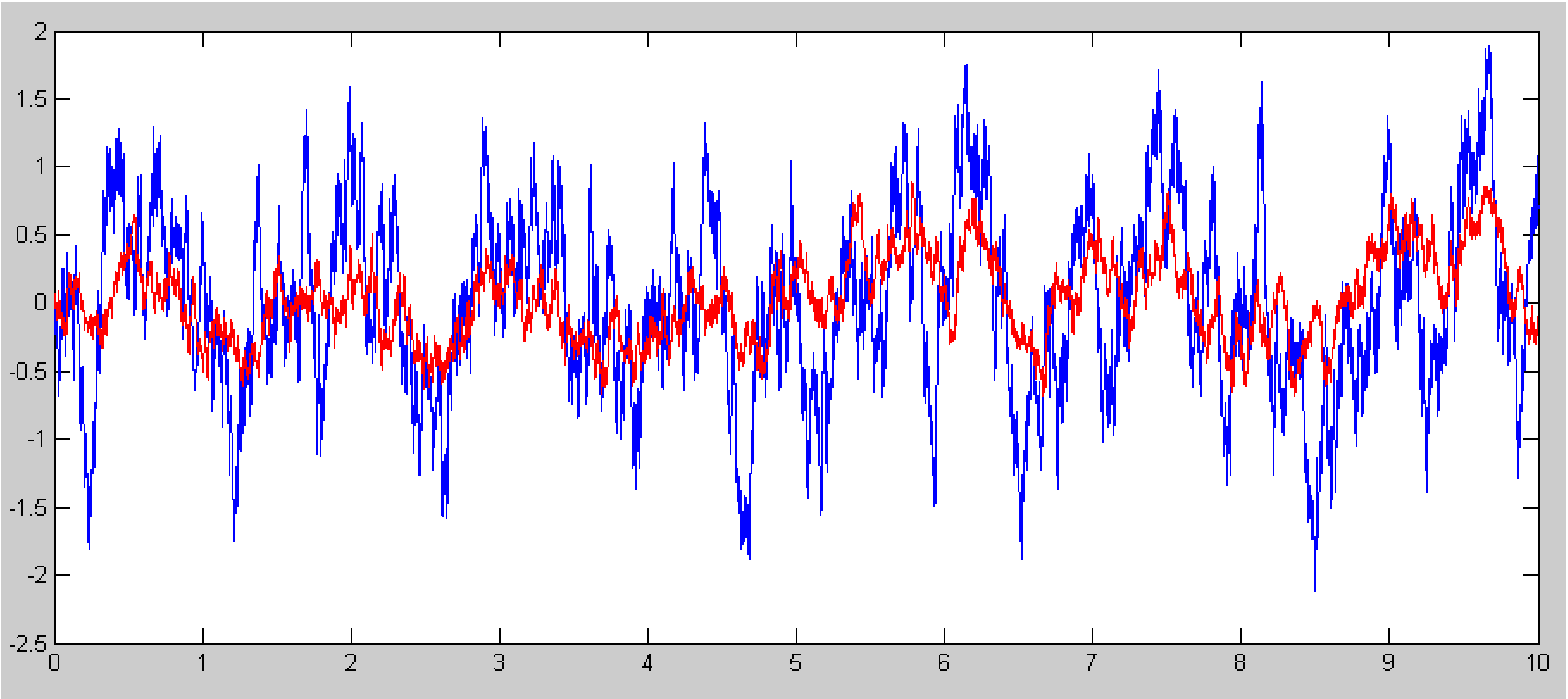


## Put the plant and observer together:

```
A12 = [A6, zeros(6,6) ; H6*C6, A6-H6*C6];
B12 = [B6, [0;0;0;0;Bw], zeros(6,1) ; B6, zeros(6,1), H6]

C12 = zeros(4,12);
C12(1,3) = 1;      % velocity
C12(2,9) = 1;      % velocity estimate
C12(3,6) = 1;      % d
C12(4,12) = 1;     % d estimate

D12 = zeros(4,3);
X0 = zeros(12,1);
t = [0:0.001:10]';
N = size(t);
U = [zeros(N), nu, ny];

y = step3(A12, B12, C12, D12, t, X0, U);
plot(t,y0(:,1),'b',t,y0(:,2),'r')
```



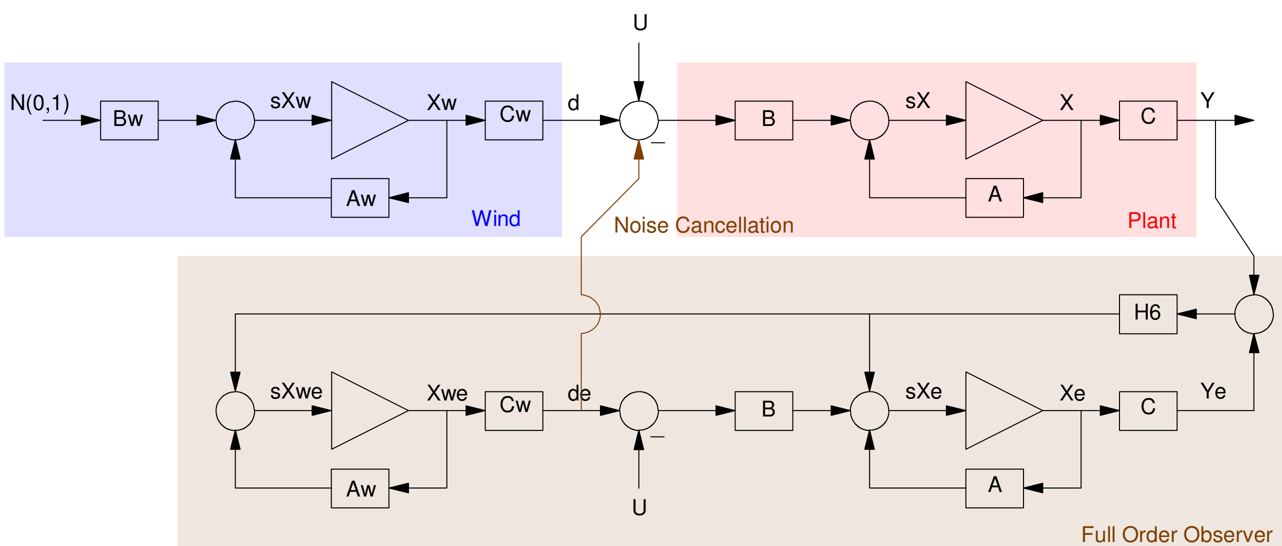

Cart Velocity (blue) and its estimate (red).




Disturbance (blue) and its estimate (red)

If you know the disturbance, you can cancel it at the input. This results in the system being

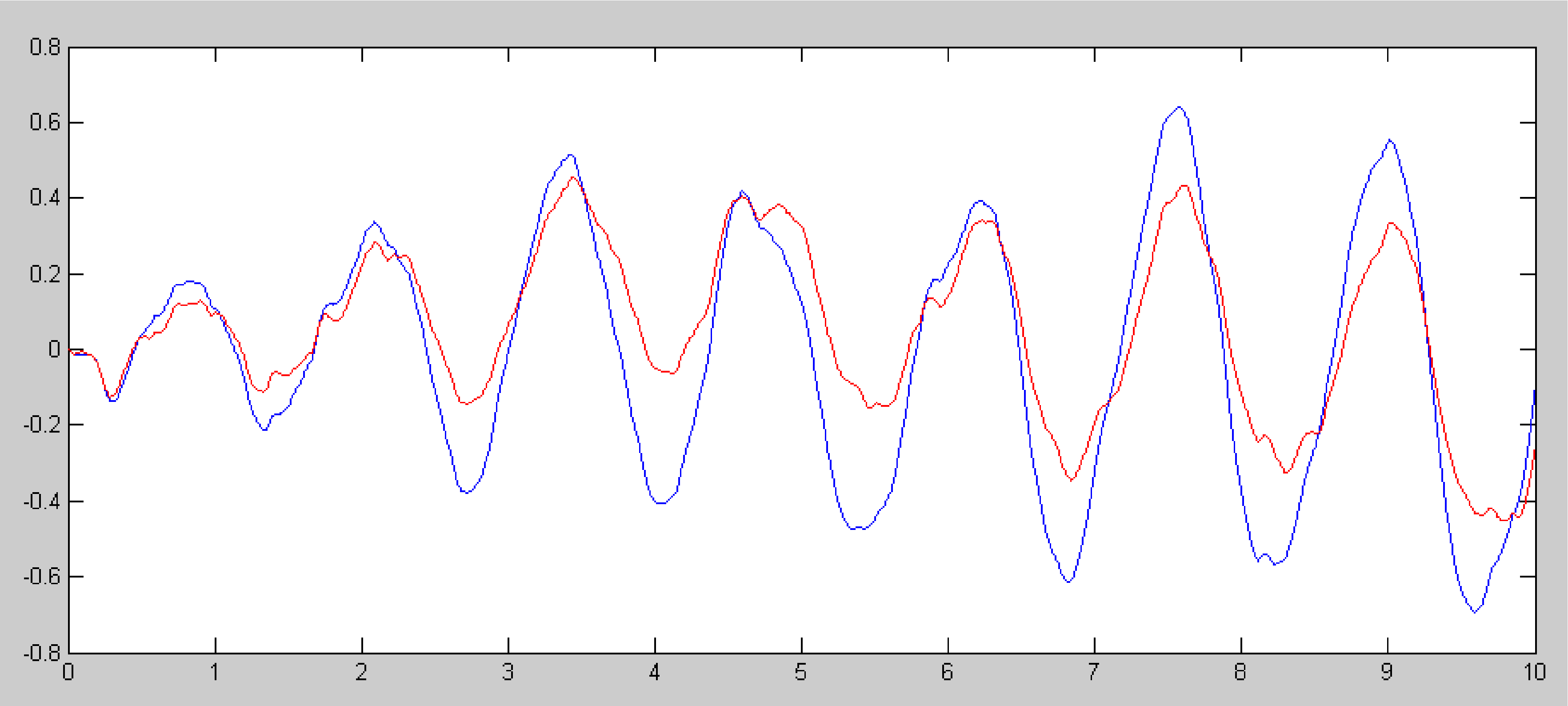
$$\begin{bmatrix} sX \\ sX_w \\ s\hat{X} \\ s\hat{X}_w \end{bmatrix} = \begin{bmatrix} A & BC_w & 0 & \underline{-BC_w} \\ 0 & A_w & 0 & 0 \\ H_xC & 0 & A - H_xC & \underline{0} \\ H_wC & 0 & 0 - H_wC & A_w \end{bmatrix} \begin{bmatrix} X \\ X_w \\ \hat{X} \\ \hat{X}_w \end{bmatrix} + \begin{bmatrix} B \\ 0 \\ B \\ 0 \end{bmatrix} U + \begin{bmatrix} 0 \\ B_w \\ 0 \\ 0 \end{bmatrix} \eta$$



This should reduce the effect of the disturbance on the output. In Matlab:

```
A12 = [A6, zeros(6,6) ; H6*C6, A6-H6*C6];

A12(1:4,11:12) = -B*Cw;
A12(7:10,11:12) = 0*B*Cw;


B12 = [B6, [0;0;0;0;Bw], zeros(6,1) ; B6, zeros(6,1), H6]
C12 = zeros(4,12);
C12(1,3) = 1;      % dx
C12(2,9) = 1;      % dx estimate
C12(3,6) = 1;      % d
C12(4,12) = 1;     % d estimate

D12 = zeros(4,3);

X0 = zeros(12,1);
t = [0:0.001:10]';
N = size(t);
U = [zeros(N), nu, ny];

y = step3(A12, B12, C12, D12, t, X0, U);
plot(t,y(:,1),'b',t,y(:,2),'r')
```

# Plotting the velocity with and without noise cancellation




Velocity due to wind disturbance without noise calcellation (blue) and with noise calcellation (red)

## Summary

A Kalman filter is a full-order observer for a system with white-noise on the inputs and sensors

- White-noise = Gaussian noise
- Uncorrelated
- Frequency content = 1

If the actual disturbance isn't white noise, you can do better

- Model the disturbance as a dynamic system
- Estimate the states of the disturbance
  - The part of the noise that is correlated
- Subtract these estimates from the input to the observer
  - Noise cancellation

The result is a better estimate of the states

- along with an estimated of the disturbances