35 Landing Vehicle Control

A controlled vehicle that is landing onto a target in the horizontal plane is equipped with a vision-based navigation system. The system gives very good resolution at low altitudes, but, unfortunately, poor resolution at high altitudes. This effect is modeled as $\sigma_\nu^2 = 0.05z^2$, where z is the altitude and σ_ν^2 is the variance of the noise ν in the measurement $y = x + \nu$; x is the true horizontal position of the craft. The sensor noise ν is considered to be of zero mean, and to have Gaussian distribution, at any particular altitude. There are no other effects, say due to roll, pitch, or yaw, in this sensor.

We consider the horizontal positioning problem in one direction only, e.g., North-South. The vehicle gets a measurement y once per second, and descends at a steady rate of $2m/s$. A controller applies a corrective thrust force according to $T = -k \times sign(y)$, and the vehicle physically responds to thrust with a velocity change: $\ddot{x} = T$. The vehicle physical behavior is also affected by a horizontal drift disturbance (due to current or winds). This is steady in time and constant over all altitudes, but its magnitude on any particular deployment can take a random value between -0.2 m/s and 0.2 m/s, uniformly distributed.

1. Is the sensor noise process either stationary or ergodic? Explain.

   Summary: The sensor noise is not stationary, because its statistics (variance in particular) are not constant through time. The sensor noise is worst at high altitudes and best at low altitudes. Ergodicity implies that the time statistics and the ensemble statistics are the same - in this case, clearly the ensemble variance at any given time cannot equal the variance over time of any particular realization. Hence the noise process is not ergodic either.

2. For an initial height of $z = 200m$, no initial error or horizontal velocity (i.e., $x(t = 0) = \dot{x}(t = 0) = 0$), and gain $k = 0.1$ what is the mean position error upon landing?

   Solution: The mean error is zero, as should be expected since everything is symmetric on the positive and negative x-axes. This is confirmed in Monte Carlo simulations; see the attached histogram with gain 0.1, and 10,000 trials.

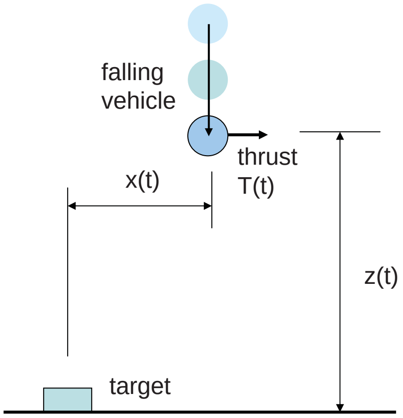
3. Under these conditions, what is the standard deviation of the position error upon landing?

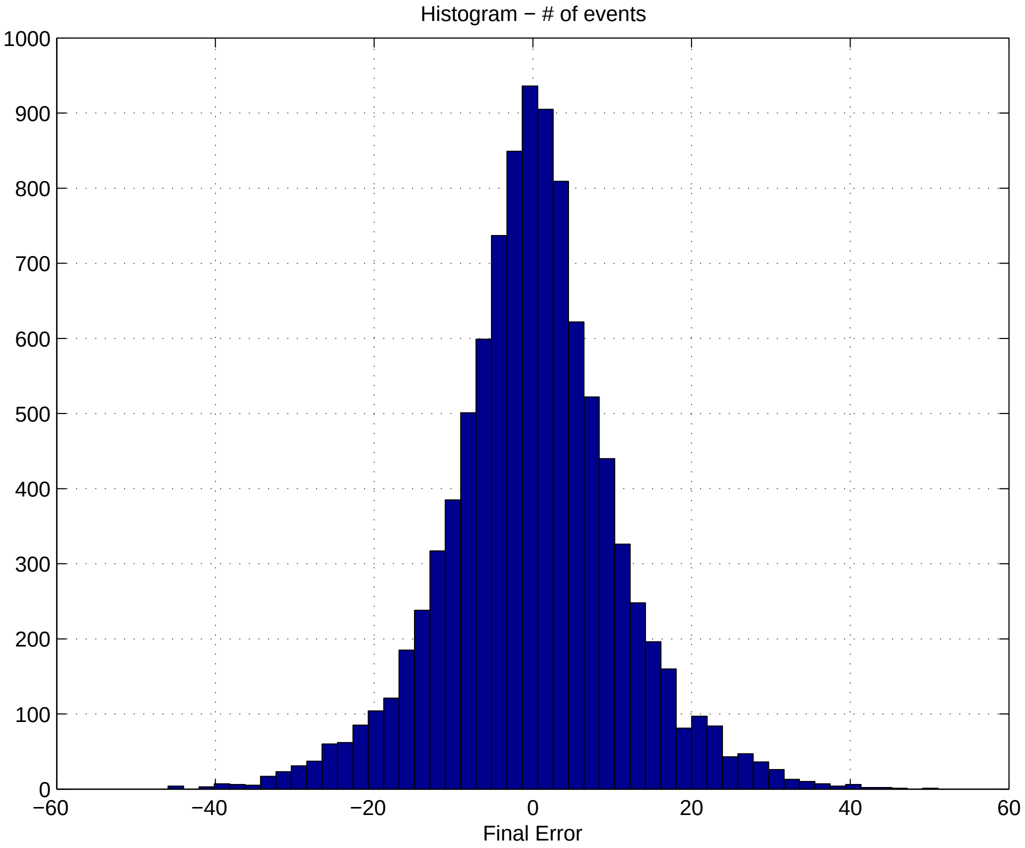
   Solution: On three "experiments" with 10,000 trials, I get [10.5, 10.7, 10.4]m as the answer - a good average is 10.5m.

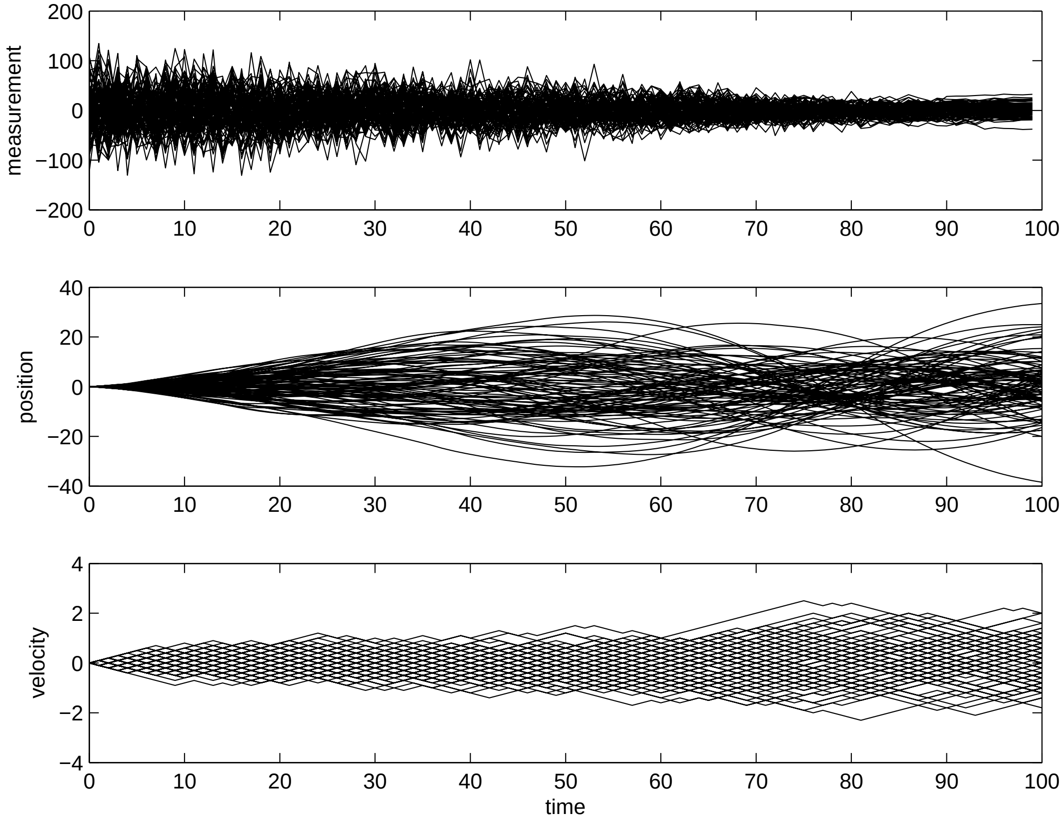
   The time traces below give the sensor noise (top subplot), the vehicle x-position (middle subplot), and the vehicle x-velocity (bottom subplot). We see that the vehicle spends a lot of effort responding to the very large sensor noise at altitude and does not really "home in" on the target effectively.

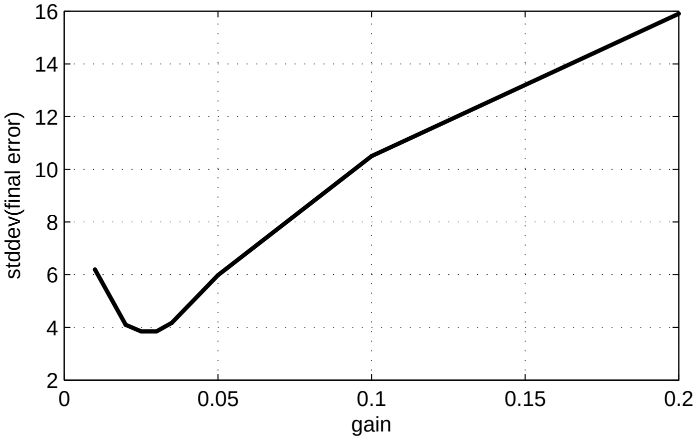
4. Can you pick another k that gives better performance?

   Solution: The last plot gives the error standard deviation for some additional values of the gain. We see that a gain of 0.027 is quite good, with a standard deviation of *about 3.85m. Note that because the vehicle has zero initial error, it is the drift that creates the initial perturbation.*









```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Falling Vehicle navigation & control simulation

clear all;
```

```
n = 100 ; % number of time steps
dt = 1 ; % time step
zdot = 2 ; % falling velocity
zInitial = 200 ; % initial elevation
N = 10000 ; % number of trials
gain = input('What is the gain? ');

figure(1);clf;hold off;
for i = 1:N,  % do many trials

    x(1) = 0 ; % set initial conditions
    xdot(1) = 0 ;

    wind = (rand-.5)*2*.2 ; % get the steady wind for this trial

    for j = 0:n-1,  % time index

        z = zInitial - zdot*j*dt ; % altitude at each time instant

        y(j+1) = x(j+1) + sqrt(.05)*z*randn ; % measurement

        % propagate the vehicle state
        xdot(j+2) = xdot(j+1) - gain*sign(y(j+1))*dt ;
        x(j+2) = x(j+1) + (xdot(j+1)+xdot(j+2))/2*dt + wind*dt ;

    end;

    if N <= 100, % make a few plots if a small # of trials
        subplot(311); plot([0:n-1],y); hold on;
        ylabel('measurement');
        subplot(312); plot([0:n]*dt,x); hold on ;
        ylabel('position');
        subplot(313); plot([0:n]*dt,xdot); hold on;
        ylabel('velocity');
        xlabel('time');
    end;

    finalError(i) = x(end) ; % store the final error
end;

figure(2);clf;hold off;
hist(finalError,50);
title('Histogram - # of events');
xlabel('Final Error');
```

```
disp(sprintf('gain:  %g.', gain)) ;
disp(sprintf('Mean error:  %g m.', mean(finalError)));
disp(sprintf('Error stddev:  %g m.', std(finalError)));

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

2.017J Design of Electromechanical Robotic Systems
Fall 2009