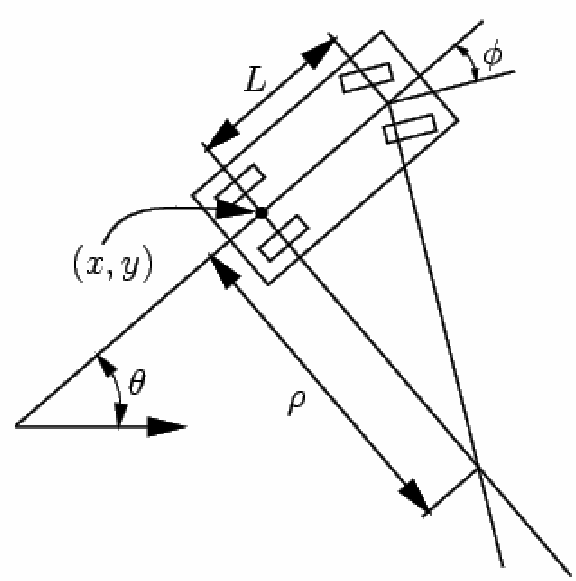
Homework 2 – Extended Kalman Filter

A robot is moving in an environment based on the following kinematic model:

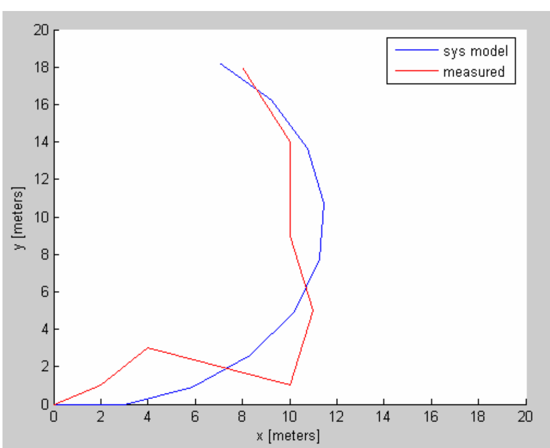


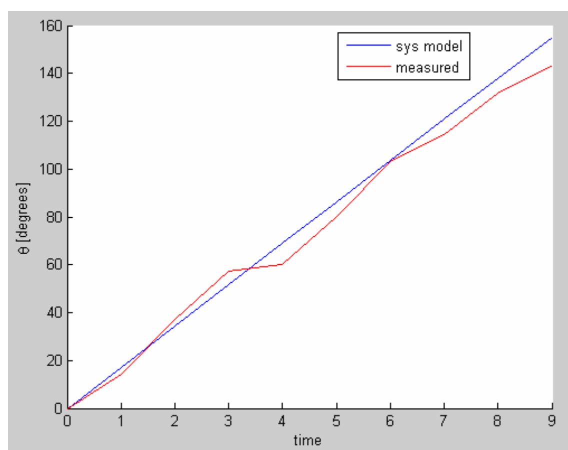
$$\dot{x} = V\cos\theta$$

$$\dot{y} = V\sin\theta$$

$$\dot{\theta} = \frac{V\tan\phi}{L}$$

Let's assume we have the ability to directly measure its position in terms of x, y, and θ, say from an overhead camera. However, this measurement is extremely noisy as seen from the graphs below.





1. Implement an Extended Kalman Filter (EKF) in MATLAB based on the above model and measurements given in dataEKF.m (dataEKF.m can be downloaded from the course web site). Do not use any built-in MATLAB functions for this exercise. Use the following values when implementing your filter: (40 points)

```
L = 1;        % length between front and rear axles
vel = 3;      % the robot has a constant velocity of 3 m/s
phi = 0.1;    % the robot has a constant steering angle of 0.1 radians
```

$\Delta t = 1$; % fixed

Q = zeros(3,3); % let's assume a small process variance
R = eye(3); % let's assume a moderate observation variance

xhat(:,1) = 0; % initialize xhat
P(1:3,1:3) = 1; % if we're certain about xhat=0 then P=0

See lecture notes for the Jacobian matrices F, W, H, and V.

Note: You only need to keep a running tab for xhat for plotting position afterwards. This is not needed for EKF implementation. You also don't need to keep a running tab of P; it is done here to better understand the code. With 10 iterations (i.e. k=1 to k=10), your filtered data should look like the following: