KALMAN FILTER

Problem 11.5

Contents

Model

y_k = A_k * x_k + noise x_k+1 = F_k * x_k + error A_k = [cos(1150*pi*t_k) sin(1150*pi*t_k) cos(1250*pi*t_k) sin(1250*pi*t_k)] F_k = [1 0 (t_k+1 - t_k) 0; 0 1 0 (t_k+1 - t_k); 0 0 1 0; 0 0 0 1];

Setup

```
clear;
close all;
load('kalman_data.mat');
N = length(t);
I4 = eye(4);
x_cap = zeros(4, N+1);
x_cap(:, 1) = [0;0;1;pi/2];
```

Part a

```
for k = 0:N-1


    % Kalman Filtering
    A_kplus1 = A(k+1, t);
    F_k = F(k, t);
    x_cap_kk = X(k, x_cap);
    if k == 0
        P_kk = pinv(A(k, t)' * A(k, t));
    end

    x_kplus1_k = F_k * x_cap_kk;
    P_kplus1_k = F_k * P_kk * F_k' + I4;
    G_kplus1 = ((P_kplus1_k * A_kplus1') * pinv(A_kplus1 * P_kplus1_k * A_kplus1' + 1));
    x_kplus1_kplus1 = x_kplus1_k + G_kplus1 * (y(k+1) - A_kplus1 * x_kplus1_k);
    P_kk = (I4 - G_kplus1 * A_kplus1) * P_kplus1_k;
    x_cap(:, k+2) = x_kplus1_kplus1;
end
```

Part b

```
y_stack = zeros(5*(N) + 1, 1);
A_stack = zeros((N) * 5 + 1, (N+2) * 4);
st_row = 2;

for k = 0:N
    if k == 0
        y_stack(5*(k)+1:5*(k+1)) = [[1 1 1 1]*X(k, x_cap); zeros(4,1)];
        A_stack(1:5, 1:4) = [A(0, t); F(0, t)];
    elseif k < N
        y_stack(5*(k)+1:5*(k+1)) = [A(k, t)*X(k, x_cap); zeros(4,1)];
        A_stack(st_row:st_row+8, 4*(k)+1:4*(k)+4) = [-I4; A(k, t); F(k, t)];
        st_row = st_row + 5;
    else
        y_stack(5*(k)+1) = [A(k, t)*X(k, x_cap)];
        A_stack(st_row:end, 4*(k)+1:4*(k)+4) = [-I4; A(k, t)];
    end

end
x_ls = pinv(A_stack) * y_stack;
hold on;
plot(x_cap(1, :), x_cap(2, :), 'o')
plot(x_ls(1:4:end), x_ls(2:4:end), '*');
```

Helper Functions

Measurement Matrix

```
function  A_k = A(k, t)
    if k == 0
        A_k = [1 1 1 1];
        return
    else
        t_k = t(k);
    end
    A_k = [cos(1150*pi*t_k) sin(1150*pi*t_k) cos(1250*pi*t_k) sin(1250*pi*t_k)];
end

% Update Matrix
function F_k = F(k, t)
    t_kplus1 = t(k+1);
    if k == 0
        t_k = 0;
    else
        t_k = t(k);
    end

    F_k = [1    0    (t_kplus1 - t_k)   0;
           0    1     0              (t_kplus1 -  t_k);
           0    0     1             0;
           0    0     0             1];
end

% Predictions
function X_kk = X(k, x_cap)
    X_kk = x_cap(:, k+1);
```

```
end
```

Published with MATLAB® R2017a