```
% MECH 6325 - HW 5
%Jonas Wagner
%2020-11-11

clear
close all;

% Problem 2
n=1;
A = -2;
C = 1;

Q_c = 2;
R_c = 1;

x_0 = 100;
P_0 = 2;
x_hat_0 = 7;

T_max = 1000;

% T = 0.4
T = 0.4;

x = x_0;
y = C * x + R_c * randn;
p = P_0;
x_hat = x_hat_0;

k = p * C' * inv(R_c);

X = [];
Y = [];
P = [];

K = [];
X_hat = [];

for t = T:T:T_max
    x_dot = A * x + Q_c * randn;
    x = x + x_dot * T;
    y = C * x + R_c * randn;
    p_dot = - p * C' * inv(R_c) * C * p + A * p + p * A' + Q_c;
    p = p + p_dot * T;

    k = p * C' * inv(R_c);
    x_hat_dot = A * x_hat + K * (y - C * x_hat);
    x_hat = x_hat + x_hat_dot * T;

    X = [X x];
    P = [P p];
    Y = [Y y];
```

```
    X_hat = [X_hat x_hat];
end

X_04 = X;
P_04 = P;
Y_04 = Y;
X_hat_04 = X_hat;


% T = 0.2
T = 0.2;

x = x_0;
y = C * x + R_c * randn;
p = P_0;
x_hat = x_hat_0;

k = p * C' * inv(R_c);

X = [];
Y = [];
P = [];

K = [];
X_hat = [];

for t = T:T:T_max
    x_dot = A * x + Q_c * randn;
    x = x + x_dot * T;
    y = C * x + R_c * randn;
    p_dot = - p * C' * inv(R_c) * C * p + A * p + p * A' + Q_c;
    p = p + p_dot * T;

    k = p * C' * inv(R_c);
    x_hat_dot = A * x_hat + K * (y - C * x_hat);
    x_hat = x_hat + x_hat_dot * T;

    X = [X x];
    P = [P p];
    Y = [Y y];

    X_hat = [X_hat x_hat];
end

X_02 = X;
P_02 = P;
Y_02 = Y;
X_hat_02 = X_hat;

% T = 0.1
T = 0.1;

x = x_0;
```

```
y = C * x + R_c * randn;
p = P_0;
x_hat = x_hat_0;

k = p * C' * inv(R_c);

X = [];
Y = [];
P = [];

K = [];
X_hat = [];

for t = T:T:T_max
    x_dot = A * x + Q_c * randn;
    x = x + x_dot * T;
    y = C * x + R_c * randn;
    p_dot = - p * C' * inv(R_c) * C * p + A * p + p * A' + Q_c;
    p = p + p_dot * T;

    k = p * C' * inv(R_c);
    x_hat_dot = A * x_hat + K * (y - C * x_hat);
    x_hat = x_hat + x_hat_dot * T;

    X = [X x];
    P = [P p];
    Y = [Y y];

    X_hat = [X_hat x_hat];
end

X_01 = X;
P_01 = P;
Y_01 = Y;
X_hat_01 = X_hat;


% Descritization Mehtod

% T = 0.4
T = 0.4;

F = 1 + A * T;
H = C;

Q = Q_c * T;
R = R_c / T;

x = x_0;
y = C * x + R_c * randn;
p_pre = P_0;
p_post = p_pre;

x_hat_pre = x_hat_0;
```

```
k = p * C' * inv(R_c);
x_hat_post = x_hat_pre;

X = [];
Y = [];

K = [];
X_hat_pre= [];
X_hat_post = [];
P_pre = [];
P_post = [];

for i = 1:(T_max / T)
    x = F * x + Q * randn;
    y = H * x + R * randn;

    p_pre = F * p_post * F' + Q;
    k = p_pre * H' * inv(H * p_pre * H' + R);
    x_hat_pre = F * x_hat_post;
    x_hat_post = x_hat_pre + k * (y - H * x_hat_pre);
    p_post = (eye(n) - k * H) *p_pre * (eye(n) - k * H)' + k * R * k';

    X = [X x];
    Y = [Y y];

    K = [K k];
    X_hat_pre = [X_hat_pre x_hat_pre];
    X_hat_post = [X_hat_post x_hat_post];
    P_pre = [P_pre p_pre];
    P_post = [P_post p_post];
end

X_04 = X;
Y_04 = Y;

K_04 = K;
X_hat_pre_04 = X_hat_pre;
X_hat_post_04 = X_hat_post;
P_pre_04 = P_pre;
P_post_04 = P_post;

p_ss_04 = dlyap(F,Q);


% T = 0.2
T = 0.2;

F = 1 + A * T;
H = C;

Q = Q_c * T;
R = R_c / T;

x = x_0;
```

```
y = C * x + R_c * randn;
p_pre = P_0;
p_post = p_pre;

x_hat_pre = x_hat_0;
k = p * C' * inv(R_c);
x_hat_post = x_hat_pre;

X = [];
Y = [];

K = [];
X_hat_pre= [];
X_hat_post = [];
P_pre = [];
P_post = [];

for i = 1:(T_max / T)
    x = F * x + Q * randn;
    y = H * x + R * randn;

    p_pre = F * p_post * F' + Q;
    k = p_pre * H' * inv(H * p_pre * H' + R);
    x_hat_pre = F * x_hat_post;
    x_hat_post = x_hat_pre + k * (y - H * x_hat_pre);
    p_post = (eye(n) - k * H) *p_pre * (eye(n) - k * H)' + k * R * k';

    X = [X x];
    Y = [Y y];

    K = [K k];
    X_hat_pre = [X_hat_pre x_hat_pre];
    X_hat_post = [X_hat_post x_hat_post];
    P_pre = [P_pre p_pre];
    P_post = [P_post p_post];
end

X_02 = X;
Y_02 = Y;

K_02 = K;
X_hat_pre_02 = X_hat_pre;
X_hat_post_02 = X_hat_post;
P_pre_02 = P_pre;
P_post_02 = P_post;

p_ss_02 = dlyap(F,Q);

% T = 0.1
T = 0.1;

F = 1 + A * T;
H = C;
```

```
Q = Q_c * T;
R = R_c / T;

x = x_0;
y = C * x + R_c * randn;
p_pre = P_0;
p_post = p_pre;

x_hat_pre = x_hat_0;
k = p * C' * inv(R_c);
x_hat_post = x_hat_pre;

X = [];
Y = [];

K = [];
X_hat_pre= [];
X_hat_post = [];
P_pre = [];
P_post = [];

for i = 1:(T_max / T)
    x = F * x + Q * randn;
    y = H * x + R * randn;

    p_pre = F * p_post * F' + Q;
    k = p_pre * H' * inv(H * p_pre * H' + R);
    x_hat_pre = F * x_hat_post;
    x_hat_post = x_hat_pre + k * (y - H * x_hat_pre);
    p_post = (eye(n) - k * H) *p_pre * (eye(n) - k * H)' + k * R * k';

    X = [X x];
    Y = [Y y];

    K = [K k];
    X_hat_pre = [X_hat_pre x_hat_pre];
    X_hat_post = [X_hat_post x_hat_post];
    P_pre = [P_pre p_pre];
    P_post = [P_post p_post];
end

X_01 = X;
Y_01 = Y;

K_01 = K;
X_hat_pre_01 = X_hat_pre;
X_hat_post_01 = X_hat_post;
P_pre_01 = P_pre;
P_post_01 = P_post;

p_ss_01 = dlyap(F,Q);

% Plotting Comparisons
t_04 = 0.4:(0.4):T_max;
```

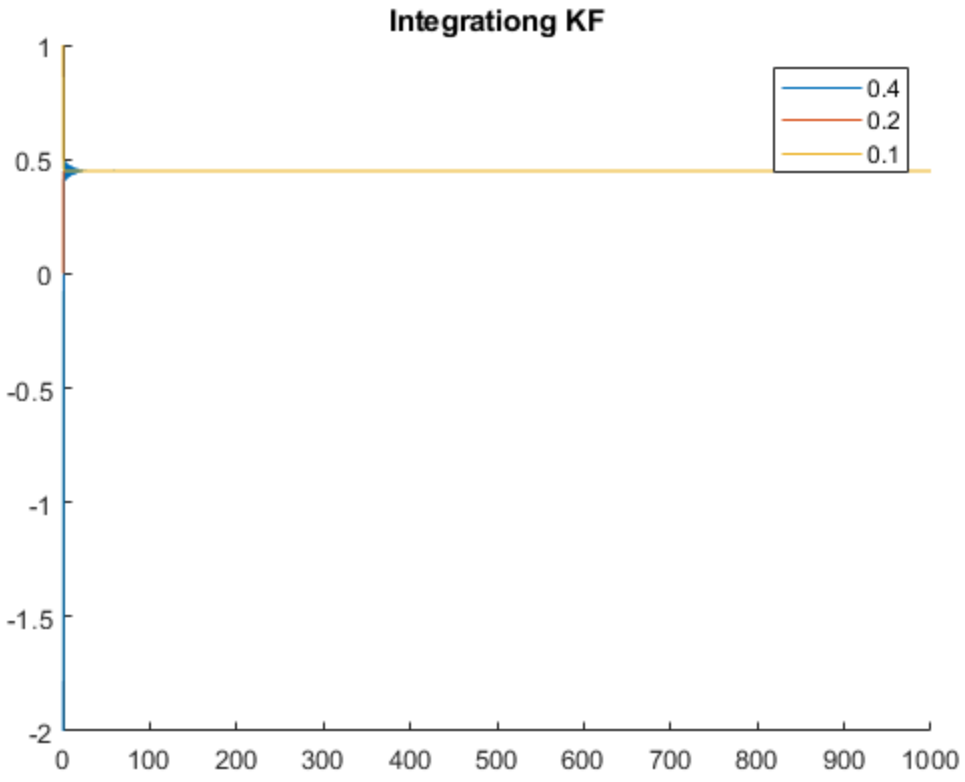
```
t_02 = 0.2:(0.2):T_max;
t_01 = 0.1:(0.1):T_max;

figure()
hold on
plot(t_04, P_04)
plot(t_02, P_02)
plot(t_01, P_01)
title('Integrationg KF')
legend('0.4','0.2','0.1')

figure()
hold on
plot(t_04, P_pre_04);
plot(t_02, P_pre_02);
plot(t_01, P_pre_01);
title('Discretization Method (A Priori)');
legend('0.4','0.2','0.1')

figure()
hold on
plot(t_04, P_post_04);
plot(t_02, P_post_02);
plot(t_01, P_post_01);
title('Discretization Method (A Postari)');
legend('0.4','0.2','0.1')

P_ss_from_plots = ...
    ["T"    "A Priori"  "A Postari";
     "0.4"  "0.825"     "0.620";
     "0.2"  "0.590"     "0.527";
     "0.1"  "0.511"     "0.486"]


clear
% close all
% Problem 5
n=2;
T_max = 10000;
T = 1;
phi = 0.9;

F = phi;

Q_1 = 1;
Q_2 = 0;
Q = diag([Q_1, Q_2]);
R = zeros(n);

x_0 = 1;
p_0 = eye(n);

x_hat_0 = zeros(2);
```

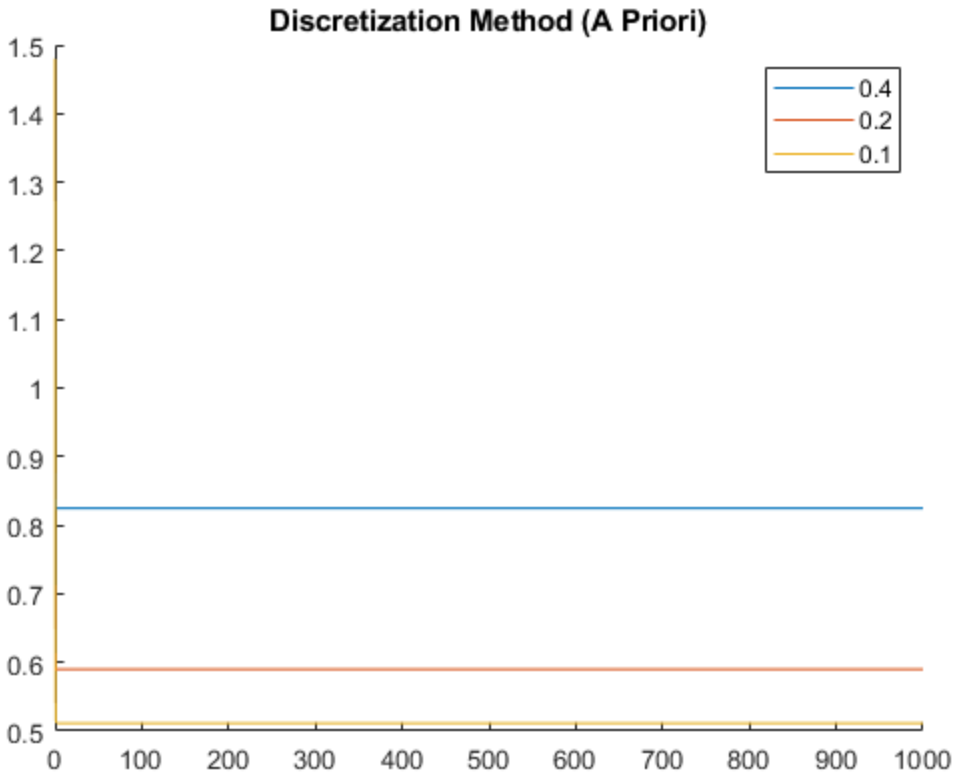
```
x = x_0;
%y = H * x + R_c * randn;
p_pre = p_0;
p_post = p_pre;

x_hat_pre = x_hat_0;
x_hat_post = x_hat_pre;

X = [];
Y = [];

K = [];
X_hat_pre= [];
X_hat_post = [];
P_pre = [];
P_post = [];

for i = 1:(T_max / T)
    % System itself
    x = phi * x + Q_1 * randn;
    y = x;

    % Kalmen Filter
    F = [x_hat_post(2)  x_hat_post(1);
        0               1];
    L = eye(n);

    p_pre = F * p_post * F' + L * Q * L';
    x_hat_pre = [x_hat_post(2) * x_hat_post(1);
             x_hat_post(2)];

    H = diag([1,0]);
    M = zeros(2);

    k = p_pre * H' * inv(p_pre(1,1)); %cheating for special case
 %inv(H * p_pre * H' + M * R * M');
    x_hat_post = x_hat_pre + k * ([y; 0] - (H * x_hat_pre));
    p_post = (eye(n) - k * H) * p_pre;

    X = [X x];
    Y = [Y y];

    K = [K k];
    X_hat_pre = [X_hat_pre x_hat_pre];
    X_hat_post = [X_hat_post x_hat_post];
    P_pre = [P_pre p_pre];
    P_post = [P_post p_post];
end

figure()
plot(X_hat_post(2,1:100))
title('Phi Estimate')
```

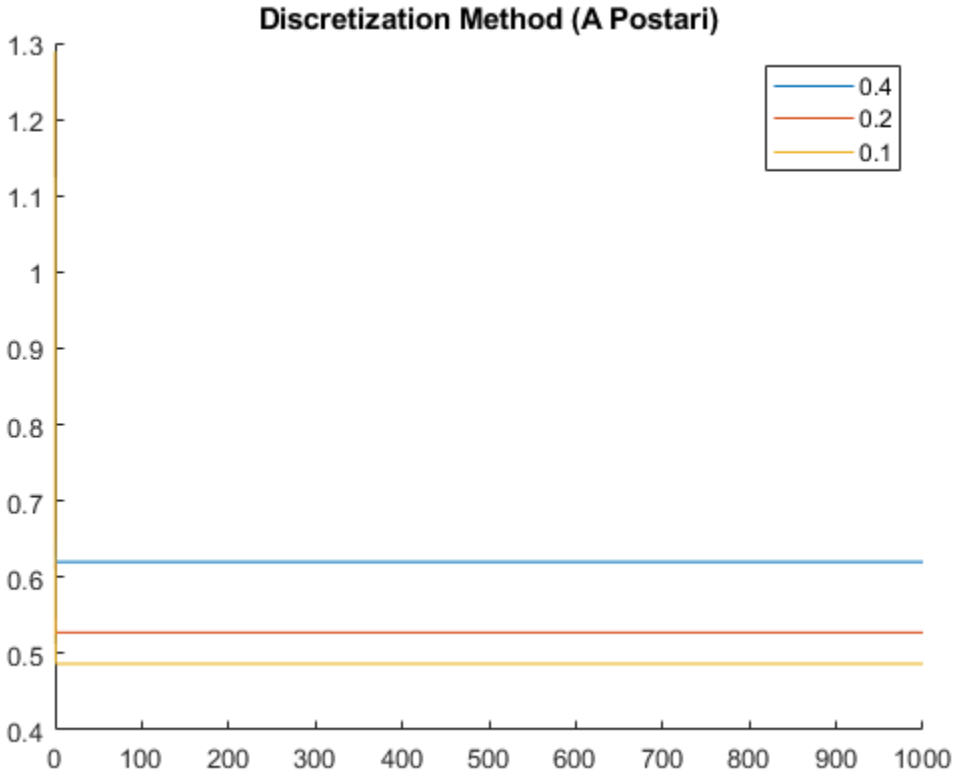
```
phi_hat_ss = mean(X_hat_post(2,0.8*T_max:T_max))
```

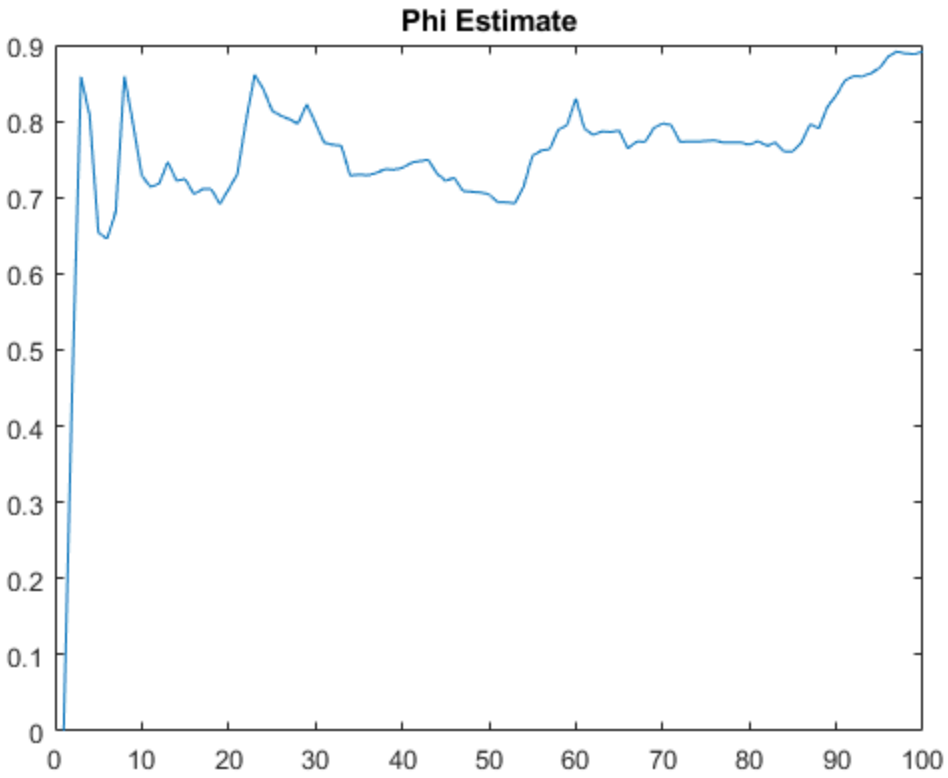
```
P_ss_from_plots =

  4×3 string array

    "T"      "A Priori"    "A Postari"
    "0.4"    "0.825"       "0.620"
    "0.2"    "0.590"       "0.527"
    "0.1"    "0.511"       "0.486"
```

```
phi_hat_ss =

    0.8958
```




Discretization Method (A Priori)
0.4
0.2
0.1
1.5
1.4
1.3
1.2
1.1
1
0.9
0.8
0.7
0.6
0.5
0
100
200
300
400
500
600
700
800
900
1000


Discretization Method (A Postari)
0.4
0.2
0.1
1.3
1.2
1.1
1
0.9
0.8
0.7
0.6
0.5
0.4
0
100
200
300
400
500
600
700
800
900
1000



Published with MATLAB® R2020b