Homework #10

Theodor Chakhachiro
UMID: 15801216
ROB501 - Mathematics for Robotics
University of Michigan, Ann Arbor

December 29, 2021

Problem 1

Part a

Computing the partial derivatives, we obtain:

$$\frac{\partial f(x)}{\partial x_1} = 6x_2 - 3x_3^3$$
$$\frac{\partial f(x)}{\partial x_2} = 6x_1 + \frac{4}{3}x_2^3$$
$$\frac{\partial f(x)}{\partial x_3} = -9x_1x_3^2$$

Thus, the Jacobian of $f(x)$ is written as:

$$\mathbf{J} = \begin{bmatrix} \frac{\partial f(x)}{\partial x_1} & \frac{\partial f(x)}{\partial x_2} & \frac{\partial f(x)}{\partial x_3} \end{bmatrix} = \begin{bmatrix} 6x_2 - 3x_3^3 & 6x_1 + \frac{4}{3}x_2^3 & -9x_1x_3^2 \end{bmatrix} = \begin{bmatrix} 6x_2 - 3x_3^3 \\ 6x_1 + \frac{4}{3}x_2^3 \\ -9x_1x_3^2 \end{bmatrix}^T$$

Now, computing the Jacobian of $f(x)$ at $x^* = \begin{bmatrix} 1 & 3 & -1 \end{bmatrix}^T$, we get:

$$\mathbf{J} = \begin{bmatrix} \frac{\partial f(x)}{\partial x_1} & \frac{\partial f(x)}{\partial x_2} & \frac{\partial f(x)}{\partial x_3} \end{bmatrix}_{x=x^*} = \begin{bmatrix} 6(3) - 3(-1)^3 & 6(1) + \frac{4}{3}(3)^3 & -9(1)(-1)^2 \end{bmatrix} = \begin{bmatrix} 21 \\ 42 \\ -9 \end{bmatrix}^T$$

Part b

For this part, the table below contains the values of δ along with the respective norm difference between the estimate of the approximation of the Jacobian estimate using symmetric differences $||J_{est} - J||$:

Table 1: Approximation of the Jacobian using Symmetric Differences

δ	4	2	1	0.5	0.25	0.125	0.0625	0.0312	0.0156
$\|\|J_{est} - J\|\|$	80	20	5	1.25	0.3125	0.0781	0.0195	0.0049	0.0012

Since there is not much of a change in the approximation of the Jacobian for $\delta = 0.0312$ and $\delta = 0.0156$, we stop there and choose $\delta = 0.0312$ as the optimal choice for the closest approximation. Also, we notice that as $\delta \longrightarrow 0$, $J_{est} = J$. The MALTAB code used to solve this part is displayed below:

```
%% HW10 Problem1
clear all
clc
%% Initializing
syms x [1 3]
f = 3*x1*(2*x2-x3^3)+x2^4/3;
J = [diff(f,x1) diff(f,x2) diff(f,x3)];
n = 6;
del = flip(2.^(-n:2));
J_sym = zeros(3,length(del));
%% Sanity Check
x1 = 1; x2 = 3; x3 = -1;
J_sol = double(subs(J)');
%% Compute Symmetric estimate
f1 = @(x1,x2,x3) 3*x1*(2*x2-x3^3)+x2^4/3;
x_star = [1 3 -1];
e = 1;
for i=1:length(del)
    J_sym(1,i) = (f1(x_star(1)+del(i)*e,x_star(2),x_star(3))-f1(x_star(1)-del(i)*e,x_star(2),x_star(3)))/(2*del(i));
    J_sym(2,i) = (f1(x_star(1),x_star(2)+del(i)*e,x_star(3))-f1(x_star(1),x_star(2)-del(i)*e,x_star(3)))/(2*del(i));
    J_sym(3,i) = (f1(x_star(1),x_star(2),x_star(3)+del(i)*e)-f1(x_star(1),x_star(2),x_star(3)-del(i)*e))/(2*del(i));
end
%% Choose most accurate solution
[del_min,idx] = min(vecnorm(J_sym-J_sol,2,1));
del_opt = del(idx);
```

Part c

Following the same procedure as part b, we generated the approximation of the Jacobian using the symmetric differences estimate for the following values of delta:

$$\delta = \begin{bmatrix} 4 & 2 & 1 & \dots & 0.001 \end{bmatrix}$$

We notice that a good estimate of the Jacobian at $x^* = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \end{bmatrix}$ is:

$$\mathbf{J}_{est} = \begin{bmatrix} 117.3699 \\ -41.3695 \\ -636.4515 \\ -3.852 \\ -11.2049 \end{bmatrix}^T$$

The MALTAB code used to solve this part is displayed below:

```
%% HW10 Problem1 Part c
clear all
clc
%% Initializing
n = 10;
del = flip(2.^(-n:2));
J_sym = zeros(5,length(del));
```

```
%% Compute Symmetric estimate
f1 = @(x1,x2,x3) 3*x1*(2*x2-x3^3)+x2^4/3;
x_star = [1 1 1 1 1];
e1 = [1 0 0 0 0];
e2 = [0 1 0 0 0];
e3 = [0 0 1 0 0];
e4 = [0 0 0 1 0];
e5 = [0 0 0 0 1];
for i=1:length(del)
    J_sym(1,i) = (funcPartC(x_star+del(i)*e1)-funcPartC(x_star-del(i)*e1))/(2*del(i));
    J_sym(2,i) = (funcPartC(x_star+del(i)*e2)-funcPartC(x_star-del(i)*e2))/(2*del(i));
    J_sym(3,i) = (funcPartC(x_star+del(i)*e3)-funcPartC(x_star-del(i)*e3))/(2*del(i));
    J_sym(4,i) = (funcPartC(x_star+del(i)*e4)-funcPartC(x_star-del(i)*e4))/(2*del(i));
    J_sym(5,i) = (funcPartC(x_star+del(i)*e5)-funcPartC(x_star-del(i)*e5))/(2*del(i));
end
%% Choose most accurate solution
J_sym
% [del_min,idx] = min(vecnorm(J_sym-J_sol,2,1));
% del_opt = del(idx);
```

Problem 2

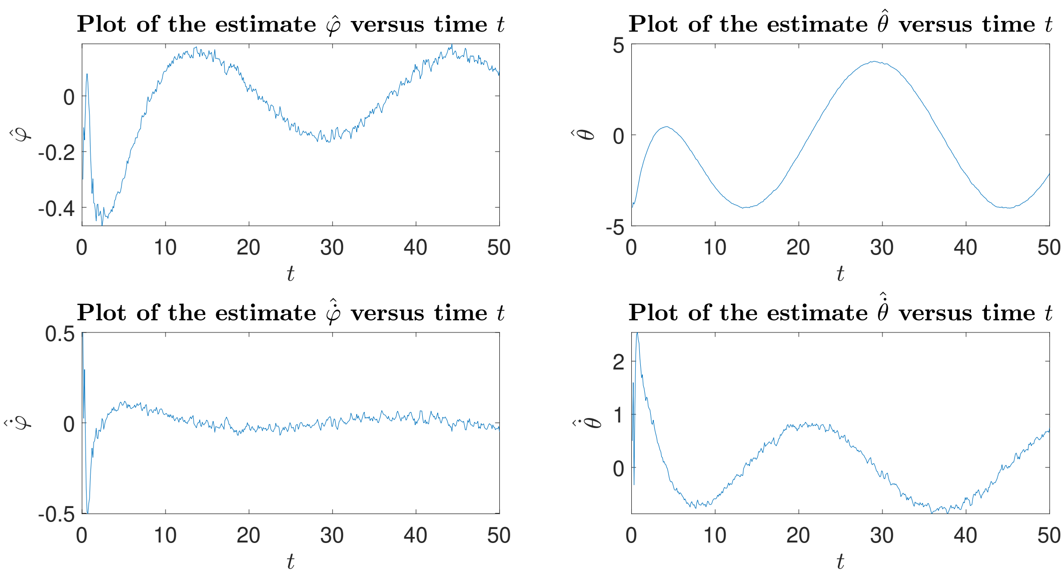

Figure 1: Subplots of the estimate state vector $\hat{x}$ versus time t

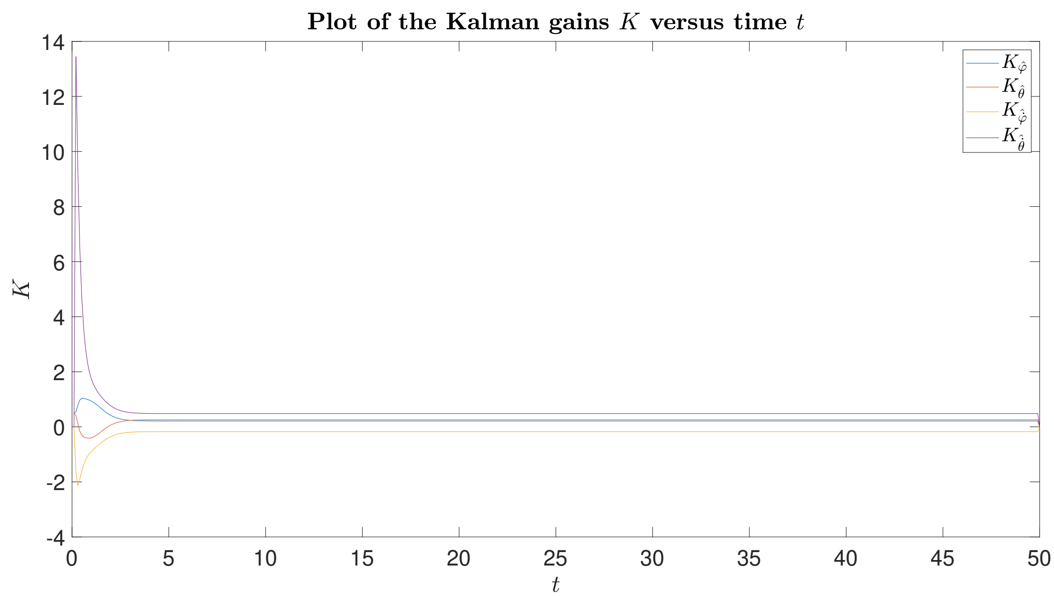

Figure 2: Plot of the Kalman gains K versus time t

The steady state values of the Kalman gain are tabulated below along with the results obtained using MATLAB's **dlqe** command:

$$K_{ss} = \begin{bmatrix} 0.2113 \\ 0.2559 \\ -0.1744 \\ 0.4816 \end{bmatrix} \qquad K_{dlqe} = \begin{bmatrix} 0.2113 \\ 0.2559 \\ -0.1744 \\ 0.4816 \end{bmatrix}$$

The MALTAB code used to solve this part is displayed below:

```
%% HW10 Problem2
clear all
clc
load SegwayData4KF.mat
%% Part a: One-step Kalman Filter
P_k = P0;
E_k = x0;
for i=1:2
    K_k = P_k*C'*inv(C*P_k*C'+Q);
    x_k = A*E_k+B*u(i)+A*K_k*(y(i)-C*E_k);
    P_k = A*(P_k-K_k*C*P_k)*A'+G*R*G';
end
%% Part b: Kalman Filter
P = cell(length(x0),length(x0),length(t));
K = zeros(length(x0),length(t));
x = zeros(size(K));
P{1} = P0;
x(:,1) = x0;
for i=1:length(t)-1
    K(:,i) = P{i}*C'*inv(C*P{i}*C'+Q);
    x(:,i+1) = A*x(:,i)+B*u(i)+A*K(:,i)*(y(i)-C*x(:,i));
    P{i+1} = A*(P{i}-K(:,i)*C*P{i})*A'+G*R*G';
end
%% Steady State values
[Kss,Pss] = dlqe(A,G,C,R,Q);
K_f = K(:,i);
P_f = P{i};
```

```
%% Plots
figure(1)
subplot(2,2,1);
plot(t,x(1,:));
title('\textbf{Plot of the estimate $\hat{\varphi}$ versus time $t$}','Interpreter','latex')
ylabel('$\hat{\varphi}$','Interpreter','latex')
xlabel('$t$','Interpreter','latex')
set(gca,'fontsize',20)

subplot(2,2,2);
plot(t,x(2,:));
title('\textbf{Plot of the estimate $\hat{\theta}$ versus time $t$}','Interpreter','latex')
ylabel('$\hat{\theta}$','Interpreter','latex')
xlabel('$t$','Interpreter','latex')
set(gca,'fontsize',20)

subplot(2,2,3);
plot(t,x(3,:));
title('\textbf{Plot of the estimate $\hat{\dot{\varphi}}$ versus time $t$}','Interpreter','
    latex')
ylabel('$\hat{\dot{\varphi}}$','Interpreter','latex')
xlabel('$t$','Interpreter','latex')
set(gca,'fontsize',20)

subplot(2,2,4);
plot(t,x(4,:));
title('\textbf{Plot of the estimate $\hat{\dot{\theta}}$ versus time $t$}','Interpreter','
    latex')
ylabel('$\hat{\dot{\theta}}$','Interpreter','latex')
xlabel('$t$','Interpreter','latex')
set(gca,'fontsize',20)

figure(2)
plot(t,K)
legend('$K_{\hat{\varphi}}$','$K_{\hat{\theta}}$','$K_{\hat{\dot{\varphi}}}$','$K_{\hat{\dot
    {\theta}}}$','Interpreter','latex')
title('\textbf{Plot of the Kalman gains $K$ versus time $t$}','Interpreter','latex')
ylabel('$K$','Interpreter','latex')
xlabel('$t$','Interpreter','latex')
set(gca,'fontsize',40)
```

Problem 3

The given discrete model is as follows:

$$x_{k+1} = x_k + \delta_t u_{k+1}$$
$$\hat{z}_k = \frac{2}{c}(5 - x_k)$$

First, for the prediction step, we compute $\hat{x}_{1|0}$ and $P_{1|0}$:

$$\hat{x}_{1|0} = \mu_0 + \delta_t \hat{u}_1 = 1 + 0.1 * 10 = 2$$
$$P_{1|0} = P_0 + \delta_t^2 * R = 0.25 + 0.1^2 * 16 = 0.41$$

Next, for the measurement update step, we compute the Kalman Gain:

$$\begin{aligned}
K_1 &= P_{1|0}C^T(CP_{1|0}C^T + Q)^{-1} \\
&= 0.41 * (-\frac{2}{c})(\frac{4}{c^2} * 0.41 + 10^{-18})^{-1} \\
&= -1.422 * 10^8
\end{aligned}$$

Now we update the covariance:

$$\begin{aligned}
P_{1|1} &= P_{1|0} - K_1CP_{1|0} \\
&= 0.41 - 1.422 * 10^8 * \frac{2}{c} * 0.41 \\
&= 0.02132
\end{aligned}$$

Now, we calculate the estimate of the new measurement for non-stochastic conditions:

$$\begin{aligned}
\hat{z}_{k+1|k} &= \frac{2}{c}(5 - \hat{x}_{k+1|k}) \\
\hat{z}_{1|0} &= \frac{2}{c}(5 - \hat{x}_{1|0}) \\
&= \frac{2}{3 * 10^8} * (5 - 2) \\
&= 2 * 10^{-8}
\end{aligned}$$

Thus, we can now calculate the new state estimate:

$$\begin{aligned}
\hat{x}_{k+1|k+1} &= \hat{x}_{k+1|k} + K_{k+1}(z_{k+1} - \hat{z}_{k+1|k}) \\
\hat{x}_{1|1} &= \hat{x}_{1|0} + K_1(z_1 - \hat{z}_{1|0}) \\
&= 2 - 1.422 * 10^8 * (2.2 * 10^{-8} - 2 * 10^{-8}) \\
&= 1.7156
\end{aligned}$$

Finally, x_1 follows the following normal distribution:

$$x_1 \sim \mathcal{N}(1.7156, 0.02132)$$

Problem 4

$$\hat{x} = \underset{x^Tx=1}{\arg\min} \quad x^TA^TAx$$

We want to find x that minimizes $||Ax||_2$ subject to $||x||_2 = 1$. Since A is a real $m \times n$ matrix, we can write the SVD of A as such:

$$A = U\Sigma V^T$$

Where U and V are orthogonal matrices. Thus, we can write the following:

$$\begin{aligned}||Ax||_2 &= x^T A^T Ax \\ &= x^T V \Sigma^T U^T U \Sigma V^T x \\ &= x^T V \Sigma^T \Sigma V^T x \\ &= ||\Sigma V^T x||_2\end{aligned}$$

Now, since Σ is a diagonal matrix whose singular values denoted as $\sigma_i = \sqrt{\lambda_i(A^T A)}$ are order such that $\sigma_1 \geq \sigma_2 \ldots \geq \sigma_p$, we can write a lower bound of the previous norm as such:

$$||\Sigma V^T x||_2 \geq \sigma_p ||V^T x||_2$$

Where:

$$||V^T x||_2 = x^T V^T V x = ||x||_2$$

Thus, the lower bound becomes:

$$||Ax||_2 = ||\Sigma V^T x||_2 \geq \sigma_p ||x||_2$$

Since we seek to minimize $||Ax||_2$, we find x such that:

$$||Ax||_2 = \sigma_p ||x||_2 = ||\lambda_p x||_2$$

As, we can see, the previous equality is of the form $Av_i = \lambda_i v_i$. Thus, x is the eigenvector of $A^T A$ associated with σ_p. The orthogonal matrix V can be decomposed as:

$$V = [v_1|v_2|\ldots|v_p]$$

Where each v_i is associated with σ_i. Thus, we can deduce that $x = v_p$, which is the last column vector of V. Finally we can write:

$$\hat{x} = \underset{x^T x=1}{\arg\min} \quad x^T A^T A x = v_p$$

Problem 5

From MATLAB, we first compute the SVD of A, we obtain:

$$U = \begin{bmatrix} -0.2150 & 0.9147 & 0.3422 \\ -0.5209 & 0.1890 & -0.8324 \\ -0.8261 & -0.3572 & 0.4358 \end{bmatrix} \quad S = \begin{bmatrix} 40.2854 & 0 & 0 \\ 0 & 0.1859 & 0 \\ 0 & 0 & 0.0051 \end{bmatrix}$$

$$V = \begin{bmatrix} -0.4797 & -0.6642 & -0.5734 \\ -0.8116 & 0.0875 & 0.5776 \\ -0.3335 & 0.7424 & -0.5810 \end{bmatrix}$$

Now, the smallest non-zero singular value observed is $\sigma_r = 0.0051$. Thus, the rank 2 approximation is given by:

$$\hat{A} = A + \Delta A$$

Where:

$$\Delta A = -\sigma_r u_r v_r^T$$

u_r and v_r are respectively the third column of U and V. We obtain:

$$\Delta A = \begin{bmatrix} 0.0010 & -0.0010 & 0.0010 \\ -0.0024 & 0.0025 & -0.0025 \\ 0.0013 & -0.0013 & 0.0013 \end{bmatrix}$$

Computing $||\Delta A||$, we get:

$$||\Delta A|| = \sqrt{\lambda_{max}(\Delta A^T \Delta A)} = 0.0051 = \sigma_r \implies \text{rank}(\hat{A}) < 3$$

Computing the rank 2 approximation of A, we get:

$$\hat{A} = A + \Delta A = \begin{bmatrix} 4.0420 & 7.0450 & 3.0150 \\ 10.0426 & 17.0345 & 7.0245 \\ 16.0073 & 27.0037 & 11.0493 \end{bmatrix}$$

The MALTAB code used to solve this part is displayed below:

```
%% HW10 Problem5
clear all
clc
%% Initializing
A = [4.041 7.046 3.014;10.045 17.032 7.027;16.006 27.005 11.048];
%% SVD + rank 2 approximation
[U,S,V] = svd(A);
[sig_r,idx] = min(S(S~=0));
del_A = -sig_r*U(:,idx)*V(:,idx)';
[~,L] = eig(del_A'*del_A);
l_max = max(L(L~=0));
A_hat = A+del_A;
%% Sanity Check
norm_del_A = sqrt(l_max);
rank(A_hat)
```