

A white paper with blue text and blue text

Description automatically generated

A screenshot of a computer

Description automatically generated

A white paper with black text and red writing

Description automatically generated

A close-up of a math equation

Description automatically generated

A math equations on a piece of paper

Description automatically generated

A screenshot of a cell phone

Description automatically generated

**MatLab Code:**

**Part 1:**

% initialization of all the parameters of the four tank system

clc; clear; close all;

Al = 28;

A2 = 32;

A3 = 28;

A4 = 32;

a1 = 0.071; a3 = 0.071;

a2 = 0.057; a4 = 0.057;

kc = 0.5;

g = 981;

gamma1 = 0.7; gamma2 = 0.6;

k1 = 3.33; k2 = 3.35;

kc = 0.5;

v1 = 3; v2 = 3;

h0 = (12.4; 12.7; 1.8; 1.4];

**Part 2:**

% initialization of the states and covariances

x0 = h0;

x\_post = x0;

C= [kc 0 0 0; 0 kc 0 0];

n = 4;

P0 = 10^5 \* eye(4);

N = 500;

**Part 3:**

L = chol(P0):

x0 = (x0 \* ones(1,N))’ \* randn(N,n) \* L;

% Initialization of the states

x1\_post = x0(:,1);

x2\_post = x0(:,2);

x3\_post = x0(:,3);

x4\_post = x0(:,4);

**Part 4:**

% Prediction equation

for i = 1:N

x1\_pri(i) = -a1/A1\*sqrt(2\*g\*x1\_post(i)) + a3/A1\*sqrt(2\*g\*x3\_post(i)) + (gamma1\*k1\*v1)/A1 + w1(i);

x2\_pri(i) = -a2/A2\*sqrt(2\*g\*x2\_post(i)) + a4/A2\*sqrt(2\*g\*x4\_post(i)) + (gamma2\*k1\*v2)/A2 + w2(i);

x3\_pri(i) = -a3/A3\*sqrt(2\*g\*x3\_post(i)) + (1 - gamma2)\*k2\*v2/A3 + w3(i)

x4\_pri(i) = -a4/A4\*sqrt(2\*g\*x4\_post(i)) + (1 – gamma1)\*k1\*v1/A4 + w4(i)

End

**Part 5:**

x\_pri = abs(x\_pri);

x1\_pri = abs(x1\_pri);

x2\_pri = abs(x2\_pri);

x3\_pri = abs(x3\_pri);

x4\_pri = abs(x4\_pri);

**Part 6:**

M = length(wt);

Q = cumsum(wt);

indx = zeros(1, N);

T = linspace(0,1-(1/N),N) + **rand/N;**

i = 1;j =1;

while(i<=N && j<=M)

while Q(j) < T(i)

j = j+1

end

indx (i) = j;

x1\_post(i) = x1\_pri(j);

x2\_post(i) = x2\_pri(j);

x3\_post(i) = x3\_pri(j);

x4\_post(i) = x4\_pri(j);

i + 1;

end

**Part 7:**

% Importance Weights (Likelihood Function)

z1 = 12.4;

z2 = 12.7;

z = [z1; z2J;

z\_true = z \* ones(1,N);

R = 10 \* eye(2);

z\_est =C\*x\_pri

v = z\_true - z\_est;

for i = 1:N

q(i) = exp(-0.5 \* (v(;,i))’ \* inv(R) \* v(:,i)));

end

% Normalizing the weights

for i = 1:N

wt(i) = q(i)/sum(q);

end