

---

```
clc;
clear all;
load("AE584_Final_P1_meas_Ts_0_01.mat");
load("AE584_Final_P1_pos_Ts_0_01.mat");

k_all = 0:1:10000;
len = length(k_all);
T = 0.01;
x0 = [2.5, 0, 1, 0, 0.5, -0.1]';
miu = 0.4;
D1 = [zeros(3,3); 0.01*eye(3)];
D2 = diag([0.01, 0.01 0.01]);
R = 1*eye(3);
Q = 100*eye(6);
A_k_all = zeros(6,6,len);
P_all = zeros(6,6,len);
P_all(:, :, 1) = 10*eye(6);
x_k_all = zeros(6,len);
x_k_all(:, 1) = x0;
C_k_all = zeros(6,len);
Tmocap = 1;

for k = 1:(len-1)
    w1_k = D1*normrnd(0,1,[3,1]);
    x_k_k = x_k_all(:,k);
    Ak = X_to_Ak(x_k_k,miu,T);
    x_k1_k = Ak*x_k_k+w1_k;
    if(mod((k)*T,Tmocap)==0)
        Ck1=X_to_C(x_k1_k);
        y_k1 =[Range(k+1);Azimuth(k+1);Elevation(k+1)]+D2*normrnd(0,1,[3,1]);
    else
        Ck1 = zeros(3,6);
        y_k1 = zeros(3,1);
    end
    Pkk = P_all(:, :, k);
    P_k1_k = Ak*Pkk*Ak'+Q;
    Kk = P_k1_k*Ck1'*inv(Ck1*P_k1_k*Ck1'+R);
    P_k1_k1 = P_k1_k-Kk*Ck1*P_k1_k;
    x_k1_k1 = x_k1_k+Kk*(y_k1-X_to_g(x_k1_k));
    x_k_all(:,k+1)=x_k1_k1;
end
x_t_1 = x_k_all;

save("data.mat", "x_t_1");
```

---

```

clear all;
load("AE584_Final_P1_meas_Ts_0_01.mat");
load("AE584_Final_P1_pos_Ts_0_01.mat");
load("data.mat");

k_all_2 = 0:1:10000;
len2 = length(k_all_2);
T2 = 0.01;
x0_2 = [2.5, 0, 1, 0, 0.5, -0.1]';
miu = 0.4;
D1_2 = [zeros(3,3); 0.01*eye(3)];
D2_2 = diag([0.01, 0.01 0.01]);
R2 = 1*eye(3);
Q2 = 100*eye(6);
A_k_all_2 = zeros(6,6,len2);
P_all_2 = zeros(6,6,len2);
P_all_2(:, :, 1) = 10*eye(6);
x_k_all_2 = zeros(6,len2);
x_k_all_2(:, 1) = x0_2;
C_k_all_2 = zeros(6,len2);
Tmocap2 = 0.1;

for k1 = 1:(len2-1)
    w1_k_2 = D1_2*normrnd(0,1,[3,1]);
    x_k_k_2 = x_k_all_2(:,k1);
    Ak_2 = X_to_Ak(x_k_k_2,miu,T2);
    x_k1_k_2 = Ak_2*x_k_k_2+w1_k_2;
    if(mod((k1)*T2,Tmocap2)==0)
        Ck1_2=X_to_C(x_k1_k_2);
        y_k1_2=[Range(k1+1);Azimuth(k1+1);Elevation(k1+1)]+D2_2*normrnd(0,1,[3,1]);
    else
        Ck1_2 = zeros(3,6);
        y_k1_2 = zeros(3,1);
    end
    Pkk_2= P_all_2(:, :, k1);
    P_k1_k_2 = Ak_2*Pkk_2*Ak_2'+Q2;
    Kk_2 = P_k1_k_2*Ck1_2'*inv(Ck1_2*P_k1_k_2*Ck1_2'+R2);
    P_k1_k1_2 = P_k1_k_2-Kk_2*Ck1_2*P_k1_k_2;
    x_k1_k1_2 = x_k1_k_2+Kk_2*(y_k1_2-X_to_g(x_k1_k_2));
    x_k_all_2(:,k1+1)=x_k1_k1_2;
end
x_t_2 = x_k_all_2;

```

---

```
figure(1)
hold on
plot3(Xref,Yref,Zref,'b','LineWidth',2);
plot3(x_t_1(1,:),x_t_1(2,:),x_t_1(3:),'g');
plot3(x_t_2(1,:),x_t_2(2,:),x_t_2(3:),'r');
hold off
axis equal;
xlabel("x");
ylabel("y");
zlabel("z");
legend("reference trajectory", "T_{mocap}=1 estimated trajectory", "T_{mocap}=0.1 estimated trajectory");
```

```
figure(2)
rcwA_Ts_0_01 = [Xref;Yref;Zref];
for i=1:1:3
    subplot(3,1,i)
    hold on
    plot(0:0.01:100,rcwA_Ts_0_01(i,:));
    plot(0:0.01:100,x_t_1(i:),'g');
    plot(0:0.01:100,x_t_2(i:),'r');
    hold off
    legend("reference", "T_{mocap}=1", "T_{mocap}=0.1");
    if(i==1)
        st = "trajectory versus time x-axis";
    elseif(i==2)
        st = "trajectory versus time y-axis";
    else
        st = "trajectory versus time z-axis";
    end
    title(st);
    xlabel("time(s)");
end
```

```
function matrix = vec_to_mat(w)
wx = w(1);
wy = w(2);
wz = w(3);
matrix = [0,-wz,wy;wz,0,-wx;-wy,wx,0];
end
```

```
function matrix = X_to_Ak(X,miu,Ts)
```

---

```

xk = X(1);
yk = X(2);
zk = X(3);
r = sqrt(xk^2+yk^2+zk^2);
Fxyz = zeros(3,3);
Fxyz(1,1) = 3*miu*xk^2/r^5-miu/r^3;
Fxyz(1,2) = 3*miu*xk*yk/r^5;
Fxyz(1,3) = 3*miu*xk*zk/r^5;
Fxyz(2,1) = 3*miu*xk*yk/r^5;
Fxyz(2,2) = 3*miu*yk^2/r^5-miu/r^3;
Fxyz(2,3) = 3*miu*zk*yk/r^5;
Fxyz(3,1) = 3*miu*xk*zk/r^5;
Fxyz(3,2) = 3*miu*zk*yk/r^5;
Fxyz(3,3) = 3*miu*zk^2/r^5-miu/r^3;
F = [zeros(3,3), eye(3);
      Fxyz,      zeros(3,3)];
matrix = expm(F*Ts);
end

```

```

function gx = X_to_g(Xk)
    x = Xk(1);
    y = Xk(2);
    z = Xk(3);
    g = [ sqrt(x^2+y^2+z^2);
          -atan2(x,y);
          atan2(z,sqrt(x^2+y^2))];
    gx = g;
    gx(2) = AzUnwrap(g(2),0);
end

```

```

function C = X_to_C(Xk)
    x = Xk(1);
    y = Xk(2);
    z = Xk(3);
    pos = [x, y, z];
    r = sqrt(x^2+y^2+z^2);
    r_min = sqrt(x^2+y^2);
    C = zeros(3,6);
    C(1,1:3) = pos/r;
    C(2,1) = -y/(r_min^2);
    C(2,2) = x/(r_min^2);
    C(3,1) = -x*z/(r^2*r_min);
    C(3,2) = -y*z/(r^2*r_min);

```

---

```
    C(3,3) = r_min/r^2;  
end
```