导航与制导实验报告

西安交通大學

INS 与 GPS 组合导航
仿真实验报告

自动化 94--胡欣盈--2194323176

目　录

一、实验要求

1、完成 INS 与 GPS 位置组合导航的仿真;

2、画出组合导航后的位置误差、速度误差曲线;

3、画出原始轨迹与组合导航后的轨迹比较图；（画图时，弧度制单位要转换成度分秒制单位）

4、结果分析

5、提交纸版实验报告（附上代码）

二、全局变量与组合导航仿真变量

全局变量：

```
R=6378160; %地球半径（长半轴）

f=1/298.3; %地球扁率

wie=7.2921151467e-5; %地球自转角速率

g0=9.7803267714; %重力加速度基础值

deg=π/180; %角度

min=deg/60; %角分

sec=min/60; %角秒

hur=3600; %小时

dph=deg/hur; %度/时

ts=0.1; %仿真采样时间
```

组合导航仿真变量：

```
GPS_Sample_Rate=10; %GPS 采样时间

Runs=10; %由于随机误差，使用 Kalman 滤波时，应多次滤波，
```

以求平均值

Tg = 3600; %陀螺仪 Markov 过程相关时间

Ta = 1800; %加速度计 Markov 过程相关时间

三、Kalman Filter

估计状态初始值： Xk = zeros(18,1);

估计协方差初始值：

Pk=diag([min,min,min,0.5,0.5,0.5,30/Re,30/Re,30,
0.1*dph, 0.1*dph, 0.1*dph, 0.1*dph, 0.1*dph, 0.1*dph,
1.e-3,1.e-3,1.e-3].^2)

系统噪声方差：

Qk=1e-6*diag([0.01,0.01,0.01,0.01,0.01,0.01,0.9780,0.97
80,0.9780]).^2

量测噪声方差： Rk=diag([1e-5,1e-5,10.3986]).^2

系数矩阵 F，G，H 的表示，参考课件 6.2.1。

四、数据文件说明

dataWbibN.txt %叠加噪声的陀螺仪角速度输出

dataFbibN.txt %叠加噪声的加速度计比力输出

dataPos.txt %原始轨迹的位置数据（依次是纬度、经度、高度）

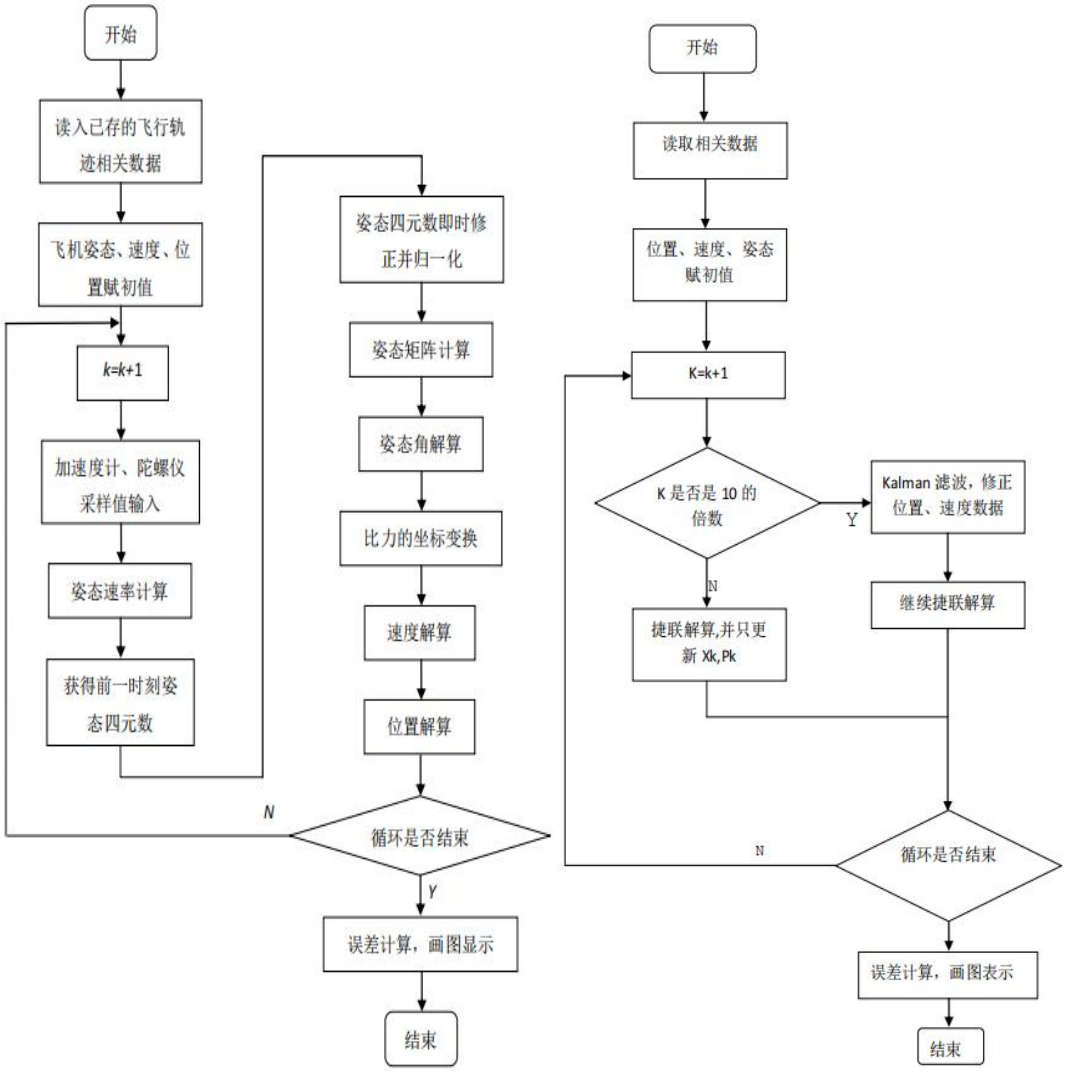
dataVn.txt %原始轨迹的速度数据（依次是东速度、北速度、天速度）

att0=[0;0;0.3491] %姿态解算矩阵初始值（依次是俯仰角、横滚角、航向角ψ）

dataGPSposN.txt %叠加噪声的 GPS 位置数据（即等间隔采样原始轨迹的位置数据，采样间隔是 10，即第 10、20,……的数据，并叠加噪声）

五、算法流程图

捷联解算四元数法流程图和整个仿真粗略流程图如下图所示：



六、实验结果

组合导航后的位置误差、速度误差曲线、原始轨迹与组合导航后的轨迹比较图如下图所示，结果与所给参考结果一致。观察轨迹图，可发现飞行轨迹肉眼可见的完全贴合原始轨迹，体现了组合导航结果

精度高，误差小的特点。

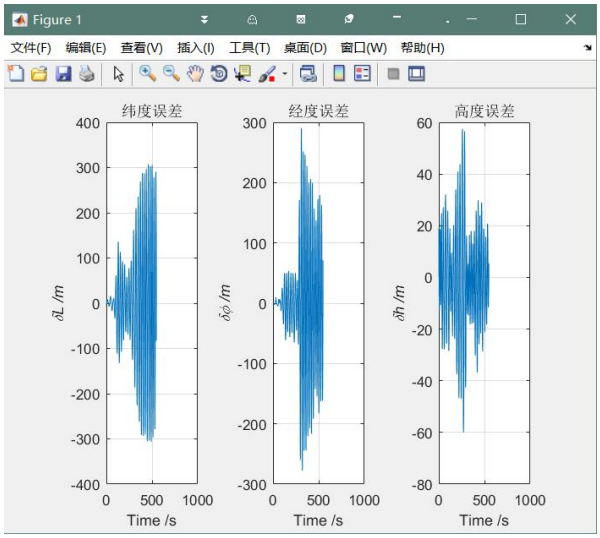


Figure 1
文件(F) 编辑(E) 查看(V) 插入(I) 工具(T) 桌面(D) 窗口(W) 帮助(H)
纬度误差
经度误差
高度误差
δL /m
δφ /m
δh /m
Time /s
Time /s
Time /s

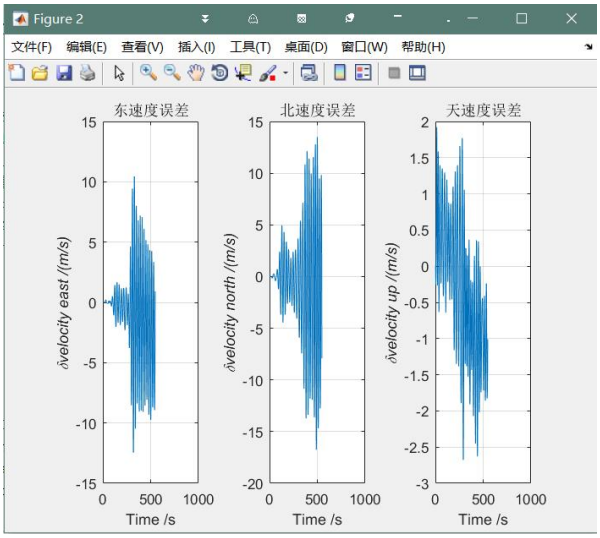


Figure 2
文件(F) 编辑(E) 查看(V) 插入(I) 工具(T) 桌面(D) 窗口(W) 帮助(H)
东速度误差
北速度误差
天速度误差
δvelocity east /(m/s)
δvelocity north /(m/s)
δvelocity up /(m/s)
Time /s
Time /s
Time /s

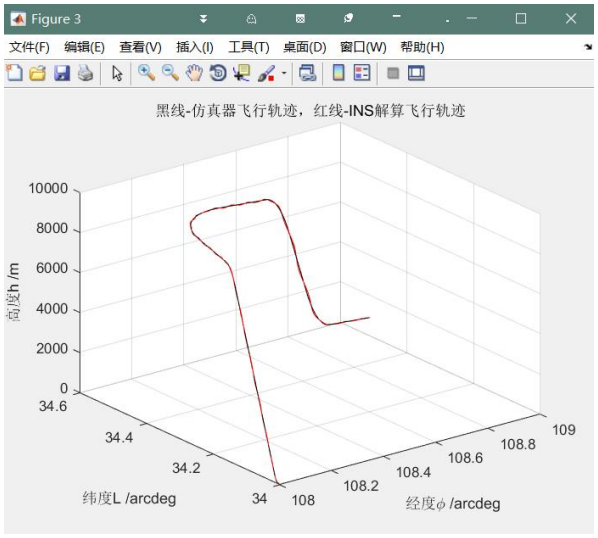


Figure 3
文件(F) 编辑(E) 查看(V) 插入(I) 工具(T) 桌面(D) 窗口(W) 帮助(H)
黑线-仿真器飞行轨迹，红线-INS解算飞行轨迹
高度h /m
纬度L /arcdeg
经度φ /arcdeg

七、代码

组合导航代码如下所示，在原有例程的基础上添加的代码在 127 行标注。

```
clc;clear;close all;
%捷联惯导更新仿真，四元数法，一阶近似算法，不考虑圆锥补偿和划桨补偿


ts=0.1;%采样时间

Re=6378160;%地球长半轴
wie=7.2921151467e-5;%地球自转角速率
f=1/298.3;%地球扁率
g0=9.7803;%重力加速度基础值
deg=pi/180;%角度
min=deg/60;%角分
sec=min/60;%角秒
hur=3600;%小时
dph=deg/hur;%度/小时

%读取数据
wbibS=dlmread('dataWbibN.txt');
fbS=dlmread('dataFbibN.txt');
posS=dlmread('dataPos.txt');
vtetS=dlmread('dataVn.txt');
p_gps=dlmread('dataGPSposN.txt');

%统计矩阵初始化
[mm,nn]=size(posS);
posSta=zeros(mm,nn);
vtSta=posSta;
attSta=posSta;

posC(:,1)=posS(:,1);  %位置向量初始值
vtC(:,1)=vtetS(:,1);  %速度向量初始值
attC(:,1)=[ 0;
                0;
          0.3491];  %姿态解算矩阵初始值


Qk=1e-6*diag([0.01,0.01,0.01,0.01,0.01,0.01,0.9780,0.9780,0.9780]).^2;%系统噪声方差矩阵
Rk=diag([1e-5,1e-5,10.3986]).^2;    %测量噪声方差
```

```
39.
40.  Tg = 3600*ones(3,1);              %陀螺仪 Markov 过程相关时间
41.  Ta = 1800*ones(3,1);              %加速度计 Markov 过程相关时间
42.
43.  GPS_Sample_Rate=10;  %GPS 采样率太高效果也不好
44.
45.
46.  StaNum=10;%重复运行次数，用于求取统计平均值
47.
48.  for OutLoop=1:StaNum
49.
50.      Pk = diag([min,min,min, 0.5,0.5,0.5, 30./Re,30./Re,30, 0.1*dph,0.1*dph,0.1*dph, 0.
    1*dph,0.1*dph,0.1*dph, 1.e-3,1.e-3,1.e-3].^2); %初始估计协方差矩阵
51.      Xk = zeros(18,1);  %初始状态
52.      %%
53.      N=size(posS,2);
54.  %     j=1;
55.      for k=1:N-1
56.          si=sin(attC(1,k));ci=cos(attC(1,k));
57.          sj=sin(attC(2,k));cj=cos(attC(2,k));
58.          sk=sin(attC(3,k));ck=cos(attC(3,k));
59.          %k 时刻姿态矩阵
60.          M=[cj*ck+si*sj*sk,     ci*sk,   sj*ck-si*cj*sk;
61.              -cj*sk+si*sj*ck,  ci*ck,   -sj*sk-si*cj*ck;
62.              -ci*sj,           si,      ci*cj]; %即 Cnb 矩阵
63.          q_1=[                       sqrt(abs(1.0+M(1,1)+M(2,2)+M(3,3)))/2.0;
64.              sign(M(3,2)-M(2,3))*sqrt(abs(1.0+M(1,1)-M(2,2)-M(3,3)))/2.0;
65.              sign(M(1,3)-M(3,1))*sqrt(abs(1.0-M(1,1)+M(2,2)-M(3,3)))/2.0;
66.              sign(M(2,1)-M(1,2))*sqrt(abs(1.0-M(1,1)-M(2,2)+M(3,3)))/2.0];
67.          fn(:,k)=M*fbS(:,k);%比力的坐标变换
68.
69.
70.
71.          %捷联惯导解算
72.          wnie=wie*[0;cos(posC(1,k));sin(posC(1,k))];%地球系相对惯性系的转动角速度在导航系
    （地理系）的投影
73.          %计算主曲率半径
74.          Rm=Re*(1-2*f+3*f*sin(posC(1,k))^2)+posC(3,k);
75.          Rn=Re*(1+f*sin(posC(1,k))^2)+posC(3,k);
76.
77.          wnen=[-vtC(2,k)/(Rm+posC(3,k));vtC(1,k)/(Rn+posC(3,k));vtC(1,k)*tan(posC(1,k))
    /(Rn+posC(3,k))];%导航系相对相对地球系的转动角速度在导航系的投影
78.          g=g0+0.051799*sin(posC(1,k))^2-0.94114e-6*posC(3,k);%重力加速度
79.          gn=[0;0;-g];%导航坐标系的重力加速度
```

```
80.
81.
82.         wbnbC(:,k)=wbibS(:,k)-M\(wnie+wnen); %姿态角转动角速率计算
83.         q=1.0/2*qmul(q_1,[0;wbnbC(:,k)])*ts+q_1;  %姿态四元数更新
84.         q=q/sqrt(q'*q);%四元数归一化
85.
86.         %姿态角更新
87.         q11=q(1)*q(1);q12=q(1)*q(2);q13=q(1)*q(3);q14=q(1)*q(4);
88.         q22=q(2)*q(2);q23=q(2)*q(3);q24=q(2)*q(4);
89.         q33=q(3)*q(3);q34=q(3)*q(4);
90.         q44=q(4)*q(4);
91.         T=[q11+q22-q33-q44,  2*(q23-q14),      2*(q24+q13);
92.             2*(q23+q14),     q11-q22+q33-q44,  2*(q34-q12);
93.             2*(q24-q13),     2*(q34+q12),      q11-q22-q33+q44];
94.
95.         attC(:,k+1)=[asin(T(3,2));atan(-T(3,1)/T(3,3));atan(T(1,2)/T(2,2))];
96.         %横滚角 gamma 修正
97.         if T(3,3)<0
98.            if attC(2,k+1)<0
99.                attC(2,k+1)=attC(2,k+1)+pi;
100.           else
101.               attC(2,k+1)=attC(2,k+1)-pi;
102.           end
103.        end
104.        %航向角 psi 修正
105.        if T(2,2)<0
106.           if T(1,2)>0
107.               attC(3,k+1)=attC(3,k+1)+pi;
108.           else
109.               attC(3,k+1)=attC(3,k+1)-pi;
110.           end
111.        end
112.        if abs(T(2,2))<1.0e-20
113.           if T(1,2)>0
114.              attC(3,k+1)=pi/2.0;
115.           else
116.              attC(3,k+1)=-pi/2.0;
117.           end
118.        end
119.
120.        %速度更新
121.        vtC(:,k+1)=vtC(:,k)+ts*(fn(:,k)+gn-cross((2*wnie+wnen),vtC(:,k)));
122.        %位置更新
123.        posC(1,k+1)=posC(1,k)+ts*vtC(2,k)/(Rm+posC(3,k));                    %纬度
```

```
        posC(2,k+1)=posC(2,k)+ts*vtC(1,k)/((Rn+posC(3,k))*cos(posC(1,k)));%经度
        posC(3,k+1)=posC(3,k)+ts*vtC(3,k);          %高度

        %添加程序
        [ F,G,H ] = GetConSis( vtC(:,k), posC(:,k), q , fn(:,k), Tg, Ta );%利用 GetConSis 得到连续系统状态转移矩阵 F、连续系统噪声矩阵 G、连续系统测量矩阵 K 矩阵
        %ins 输出速度向量\位置向量\姿态四元数\解算输入，导航系下比力向量\加速度计误差漂移相关时间\陀螺仪误差漂移相关时间

        if mod(k+1,10) == 0 %判断 k 是否为 10 的倍数，是则进行 Kalman 滤波，进行修正
            Zk = - posC(:,k+1) + p_gps(:,(k+1)/10);%Zk   量测位置误差向量
            [ E_att, E_pos, E_vn, Xk, Pk ] = kalman_GPS_INS_correct( Xk, Qk, Pk, F, G, H, ts , Zk , Rk);%Kalman 滤波
            %k-1 时刻状态向量（估计值）\系统噪声方差矩阵\估计均方误差矩阵   迭代步长\量测位置误差向量\测量噪声方差矩阵

            %修正位置、速度数据
            posC(:,k+1) = posC(:,k+1) + E_pos;
            vtC(:,k+1) = vtC(:,k+1) + E_vn;
        else
            [ E_att, E_pos, E_vn, Xk, Pk ] = kalman_GPS_INS_correct( Xk, Qk, Pk, F, G, H, ts );%GPS 修正惯性导航
        end

    end



    %统计矩阵更新
    attSta=attSta+attC;
    vtSta=vtSta+vtC;
    posSta=posSta+posC;

end
%对统计矩阵取平均
attC=1./StaNum*attSta;
posC=1./StaNum*posSta;
vtC=1./StaNum*vtSta;

%解算值与仿真值的误差 作图
%解算矩阵均为 3x(N+1)，需做处理
%N 点数据，相邻两点采样间隔为 0.1s,做作图横轴修正
for i=1:N
    time(i)=i*ts;
```

```
    Rm = Re*(1-2*f+3*f*(sin(attC(1,i)))^2);
    Rn = Re*(1+f*(sin(attC(1,i))))^2);
    Latitude_error(i)=(posC(1,i)-posS(1,i))*Rm;
    Longitude_error(i)=(posC(2,i)-posS(2,i))*Rn*cos(attC(1,i));
end

posCp=posC(:,1:N);
figure;
subplot(131);plot(time,Latitude_error);title('纬度误差
');xlabel('Time /s');ylabel('\it\deltaL /m');grid on;
subplot(132);plot(time,Longitude_error);title('经度误差
');xlabel('Time /s');ylabel('\it\delta\phi /m');grid on;
subplot(133);plot(time,posCp(3,:)-posS(3,:));title('高度误差
');xlabel('Time /s');ylabel('\it\deltah /m');grid on;

vtCp=vtC(:,1:N);
figure;
subplot(131);plot(time,vtCp(1,:)-vtetS(1,:));title('东速度误差
');xlabel('Time /s');ylabel('\it\deltavelocity east /(m/s)');grid on;
subplot(132);plot(time,vtCp(2,:)-vtetS(2,:));title('北速度误差
');xlabel('Time /s');ylabel('\it\deltavelocity north /(m/s)');grid on;
subplot(133);plot(time,vtCp(3,:)-vtetS(3,:));title('天速度误差
');xlabel('Time /s');ylabel('\it\deltavelocity up /(m/s)');grid on;

%三维飞行轨迹图
figure;
plot3(posS(2,:)/pi*180,posS(1,:)/pi*180,posS(3,:),'k');
hold on;
plot3(posCp(2,:)/pi*180,posCp(1,:)/pi*180,posCp(3,:),'r');grid on;
ylabel('纬度 L /arcdeg');xlabel('经度\phi /arcdeg');zlabel('高度 h /m');title('黑线-仿真器飞行轨迹，红线-INS 解算飞行轨迹');
```