Table of Contents


```
clear all; close all;
```

Do my best to remove for loops to make vectoried

```
load('coe_elp','coeM')
mu  = 398600; % mu for earth
rng('default') % For reproducibility
s = rng;
```

Create orbit

```
inc = 30; % deg
RAAN = 40;% deg
e = 0.3; % ecc for now, e = 0, e = 1.2
%e=0.6;
e=0;
w = 70;% deg, arg of perapsis
rp = 2*7178.1; % km

a=rp*(1-e); % get semi major
ra=a/(1+e); % get appoapsis
h=sqrt(a*(1-e^2)*mu); % get momentum
TAd=[47,107,138]; % TA
```

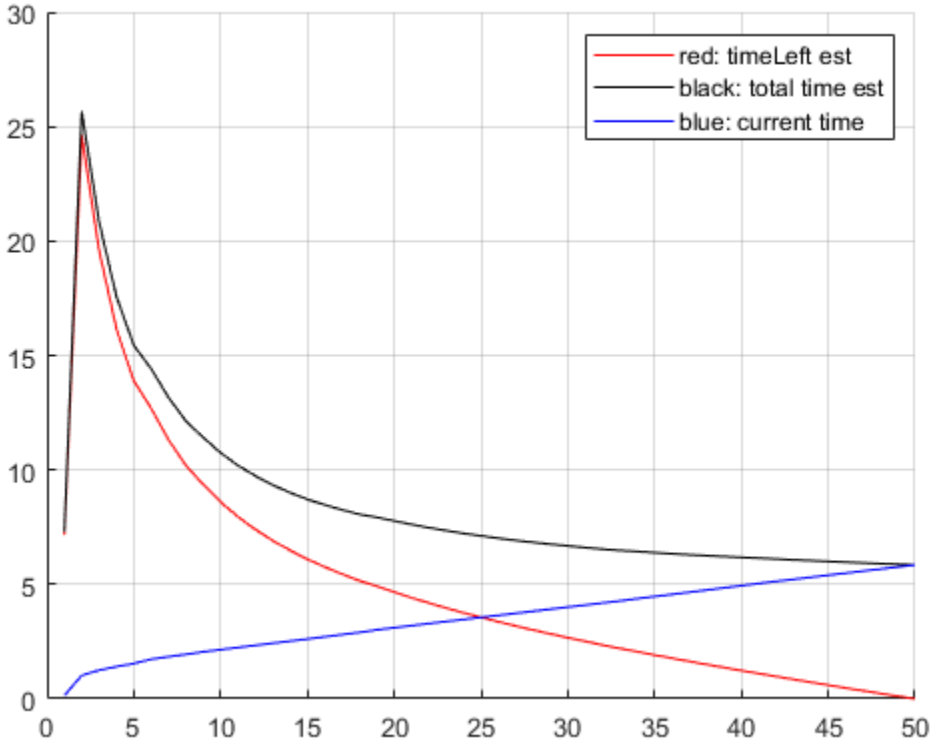
Note in hindsight this is not super nessisary

```
for i=1:3 % add given TAs
    TA=TAd(i);
    coeM(i,:)=[h e RAAN inc w TA a];
end


numbSamp=10; % set numbSamp
%sigmaA=linspace(0,.5,40); % set sigma (km) to go over
%TAdistA=linspace(35,42,1600); % set TA dist to go over
TAdistA=linspace(1,120,200); % set TA dist to go over
```

```
%sigmaA=linspace(0,2,400); % set sigma (km) to go over
sigmaA=linspace(0,2,50); % set sigma (km) to go over
%TAdistA=linspace(1,60,800); % set TA dist to go over
OrbType='circ';
coe=coeM(:,1:6);

%EdistA=atan(
MAdistA=TAdistA;

for i=1:length(MAdistA)
    Earr(i)=kepler_E(e,MAdistA(i)*pi/180);
end
TAdistAReal=2*atan(sqrt((1+e)/(1-e))*tan(Earr/2));
TAdistA=TAdistAReal*180/pi;
%
%{
OrbType='circ';
[rmsGv2,rmsHGv2,rmsME] =
 checkDisc(numbSamp,sigmaA,TAdistA,coeM,mu,OrbType,rp);

%% Plot
close all;
offSet=1;
figure(1)
surf(TAdistA(offSet:end),sigmaA(2:end),rmsGv2(2:end,offSet:end));
ylabel('\sigma')
xlabel('$\bigtriangleup$ M (deg)','Interpreter','latex')
zlabel('RMS')
%set(gca,'zscale','log')
tiS=sprintf('RMS');
title(tiS)
colorbar

figure(2)
surf(TAdistA(offSet:end),sigmaA(2:end),rmsHGv2(2:end,offSet:end));
ylabel('\sigma')
xlabel('$\bigtriangleup$ M (deg)','Interpreter','latex')
zlabel('RMS')
%set(gca,'zscale','log')
tiS=sprintf('RMS');
title(tiS)
colorbar

figure(3)
surf(TAdistA(offSet:end),sigmaA(2:end),rmsME(2:end,offSet:end));
ylabel('\sigma')
xlabel('$\bigtriangleup$ M (deg)','Interpreter','latex')
zlabel('RMS')
%set(gca,'zscale','log')
tiS=sprintf('RMS');
title(tiS)
colorbar
```

```
keyboard
%}
[rmsCir,rmsMH,rmsGv2,rmsHGv2] =
 RMS_G_HG(numbSamp,sigmaA,TAdistA,coeM,mu,OrbType,rp);
rmsEc=rmsCir;
offSet=1;

0.000
1 of 50, for circ,  0.1 sec elaps, 7.3 est tot, 7.2 left
2 of 50, for circ,  1.0 sec elaps, 25.7 est tot, 24.6 left
3 of 50, for circ,  1.3 sec elaps, 20.8 est tot, 19.6 left
4 of 50, for circ,  1.4 sec elaps, 17.6 est tot, 16.2 left
5 of 50, for circ,  1.5 sec elaps, 15.4 est tot, 13.9 left
6 of 50, for circ,  1.7 sec elaps, 14.4 est tot, 12.7 left
7 of 50, for circ,  1.8 sec elaps, 13.2 est tot, 11.3 left
8 of 50, for circ,  1.9 sec elaps, 12.1 est tot, 10.2 left
9 of 50, for circ,  2.1 sec elaps, 11.4 est tot, 9.4 left
10 of 50, for circ,  2.1 sec elaps, 10.7 est tot, 8.6 left
11 of 50, for circ,  2.2 sec elaps, 10.2 est tot, 8.0 left
12 of 50, for circ,  2.3 sec elaps, 9.7 est tot, 7.4 left
13 of 50, for circ,  2.4 sec elaps, 9.3 est tot, 6.9 left
14 of 50, for circ,  2.5 sec elaps, 9.0 est tot, 6.5 left
15 of 50, for circ,  2.6 sec elaps, 8.7 est tot, 6.1 left
16 of 50, for circ,  2.7 sec elaps, 8.5 est tot, 5.8 left
17 of 50, for circ,  2.8 sec elaps, 8.3 est tot, 5.5 left
18 of 50, for circ,  2.9 sec elaps, 8.1 est tot, 5.2 left
19 of 50, for circ,  3.0 sec elaps, 7.9 est tot, 4.9 left
20 of 50, for circ,  3.1 sec elaps, 7.8 est tot, 4.7 left
21 of 50, for circ,  3.2 sec elaps, 7.6 est tot, 4.4 left
22 of 50, for circ,  3.3 sec elaps, 7.5 est tot, 4.2 left
23 of 50, for circ,  3.4 sec elaps, 7.3 est tot, 4.0 left
24 of 50, for circ,  3.5 sec elaps, 7.2 est tot, 3.8 left
25 of 50, for circ,  3.6 sec elaps, 7.1 est tot, 3.6 left
26 of 50, for circ,  3.6 sec elaps, 7.0 est tot, 3.4 left
27 of 50, for circ,  3.7 sec elaps, 6.9 est tot, 3.2 left
28 of 50, for circ,  3.8 sec elaps, 6.8 est tot, 3.0 left
29 of 50, for circ,  3.9 sec elaps, 6.8 est tot, 2.8 left
30 of 50, for circ,  4.0 sec elaps, 6.7 est tot, 2.7 left
31 of 50, for circ,  4.1 sec elaps, 6.6 est tot, 2.5 left
32 of 50, for circ,  4.2 sec elaps, 6.5 est tot, 2.4 left
33 of 50, for circ,  4.3 sec elaps, 6.5 est tot, 2.2 left
34 of 50, for circ,  4.4 sec elaps, 6.4 est tot, 2.1 left
35 of 50, for circ,  4.5 sec elaps, 6.4 est tot, 1.9 left
36 of 50, for circ,  4.6 sec elaps, 6.3 est tot, 1.8 left
37 of 50, for circ,  4.7 sec elaps, 6.3 est tot, 1.6 left
38 of 50, for circ,  4.8 sec elaps, 6.3 est tot, 1.5 left
39 of 50, for circ,  4.8 sec elaps, 6.2 est tot, 1.4 left
40 of 50, for circ,  4.9 sec elaps, 6.2 est tot, 1.2 left
41 of 50, for circ,  5.0 sec elaps, 6.1 est tot, 1.1 left
42 of 50, for circ,  5.1 sec elaps, 6.1 est tot, 1.0 left
43 of 50, for circ,  5.2 sec elaps, 6.1 est tot, 0.8 left
44 of 50, for circ,  5.3 sec elaps, 6.0 est tot, 0.7 left
45 of 50, for circ,  5.4 sec elaps, 6.0 est tot, 0.6 left
46 of 50, for circ,  5.5 sec elaps, 6.0 est tot, 0.5 left
```

```
47 of 50, for circ,  5.6 sec elaps, 5.9 est tot, 0.4 left
48 of 50, for circ,  5.7 sec elaps, 5.9 est tot, 0.2 left
49 of 50, for circ,  5.8 sec elaps, 5.9 est tot, 0.1 left
50 of 50, for circ,  5.9 sec elaps, 5.9 est tot, 0.0 left
```



pLots

```
close all;
figure(1)
surf(MAdistA(offSet:end),sigmaA(2:end),rmsCir(2:end,offSet:end));
ylabel('\sigma')
xlabel('$\bigtriangleup$ M (deg)','Interpreter','latex')
zlabel('RMS - a ')
set(gca,'zscale','log')
tiS=sprintf('Gibbs Method RMS-a, ecc = %.3f',e);
title(tiS)
colorbar

figure(2)
surf(MAdistA(offSet:end),sigmaA(2:end),rmsMH(2:end,offSet:end))
ylabel('\sigma')
xlabel('$\bigtriangleup$ M (deg)','Interpreter','latex')
zlabel('RMS-a')
set(gca,'zscale','log')
tiS=sprintf('Herrick Gibbs RMS-a, ecc = %.3f',e);
title(tiS)
```

```
colorbar

figure(12)
surf(TAdistA(offSet:end),sigmaA(2:end),rmsMH(2:end,offSet:end))
ylabel('\sigma')
xlabel('$\bigtriangleup$ TA (deg)','Interpreter','latex')
zlabel('RMS-a')
set(gca,'zscale','log')
tiS=sprintf('Herrick Gibbs RMS-a, ecc = %.3f',e);
title(tiS)
colorbar

mAarr=[];
tf21A=[];
tf32A=[];
tf31A=[];
EarrA=[];
Tt=2*pi/( (mu^2 *(1-coeM(1,2)^2)^(3/2))/ coeM(1,1)^3);
for TAdistC=1:length(TAdistA)%120
    TAdist=TAdistA(TAdistC);
    TAarr=(0:TAdist:2*TAdist)';
    E=2*atan( sqrt(1-coeM(1,2))/sqrt(1+coeM(1,2))
 *tan(.5*TAarr*pi/180));
    MAarr= E-coeM(1,2)*sin(E);

    tf21=(Tt/(2*pi))*(MAarr(2)-MAarr(1));%*pi/180;
    tf32=(Tt/(2*pi))*(MAarr(3)-MAarr(2));%*pi/180;
    tf31=(Tt/(2*pi))*(MAarr(3)-MAarr(1));%*pi/180;
    mAarr(TAdistC)=MAarr(2)*180/pi;
    tf21A(TAdistC)=tf21;
    tf32A(TAdistC)=tf32;
    tf31A(TAdistC)=tf31;
    EarrA(TAdistC)=E(2);
    mastM(TAdistC,:)=MAarr.*180/pi;
end
figure(13)
surf(mAarr(offSet:end),sigmaA(2:end),rmsMH(2:end,offSet:end))
ylabel('\sigma')
xlabel('$\bigtriangleup$ mAarr (deg)','Interpreter','latex')
zlabel('RMS-a')
set(gca,'zscale','log')
tiS=sprintf(' mAarr Herrick Gibbs RMS-a, ecc = %.3f',e);
title(tiS)
colorbar

figure(14)
surf(tf21A(offSet:end),sigmaA(2:end),rmsMH(2:end,offSet:end))
ylabel('\sigma')
xlabel('$\bigtriangleup$ tf21A (sec)','Interpreter','latex')
zlabel('RMS-a')
set(gca,'zscale','log')
tiS=sprintf('tf21A Herrick Gibbs RMS-a, ecc = %.3f',e);
title(tiS)
colorbar
```

```
figure(15)
surf(tf32A(offSet:end),sigmaA(2:end),rmsMH(2:end,offSet:end))
ylabel('\sigma')
xlabel('$\bigtriangleup$ tf32A (deg)','Interpreter','latex')
zlabel('RMS-a')
set(gca,'zscale','log')
tiS=sprintf('tf32A Herrick Gibbs RMS-a, ecc = %.3f',e);
title(tiS)
colorbar

figure(16)
surf(tf31A(offSet:end),sigmaA(2:end),rmsMH(2:end,offSet:end))
ylabel('\sigma')
xlabel('$\bigtriangleup$ tf31A (deg)','Interpreter','latex')
zlabel('RMS-a')
set(gca,'zscale','log')
tiS=sprintf('tf31A Herrick Gibbs RMS-a, ecc = %.3f',e);
title(tiS)
colorbar

figure(17)
surf(EarrA(offSet:end),sigmaA(2:end),rmsMH(2:end,offSet:end))
ylabel('\sigma')
xlabel('$\bigtriangleup$ EarrA (deg)','Interpreter','latex')
zlabel('RMS-a')
set(gca,'zscale','log')
tiS=sprintf('EarrA Herrick Gibbs RMS-a, ecc = %.3f',e);
title(tiS)
colorbar

tdiffA=[tf21A;tf32A;tf31A;];
%tf31A(offSet:end)

figure(18)
hold on
%surf(MAdistA(offSet:end),(1:1:3),tdiffA(:,offSet:end))
plot(MAdistA(offSet:end),tdiffA(1,offSet:end))
plot(MAdistA(offSet:end),tdiffA(2,offSet:end))
plot(MAdistA(offSet:end),tdiffA(3,offSet:end))
ylabel('t21, t32, t31')
xlabel('$\bigtriangleup$ M (deg)','Interpreter','latex')
%zlabel('tf')
%set(gca,'zscale','log')
tiS=sprintf('tf all Herrick Gibbs RMS-a, ecc = %.3f',e);
title(tiS)
legend('t21','t32','t31')
%colorbar

figure(19)
surf(TAdistA(offSet:end),(1:1:3),tdiffA(:,offSet:end))
ylabel('t21, t32, t31')
xlabel('$\bigtriangleup$ TA (deg)','Interpreter','latex')
zlabel('tf')
```

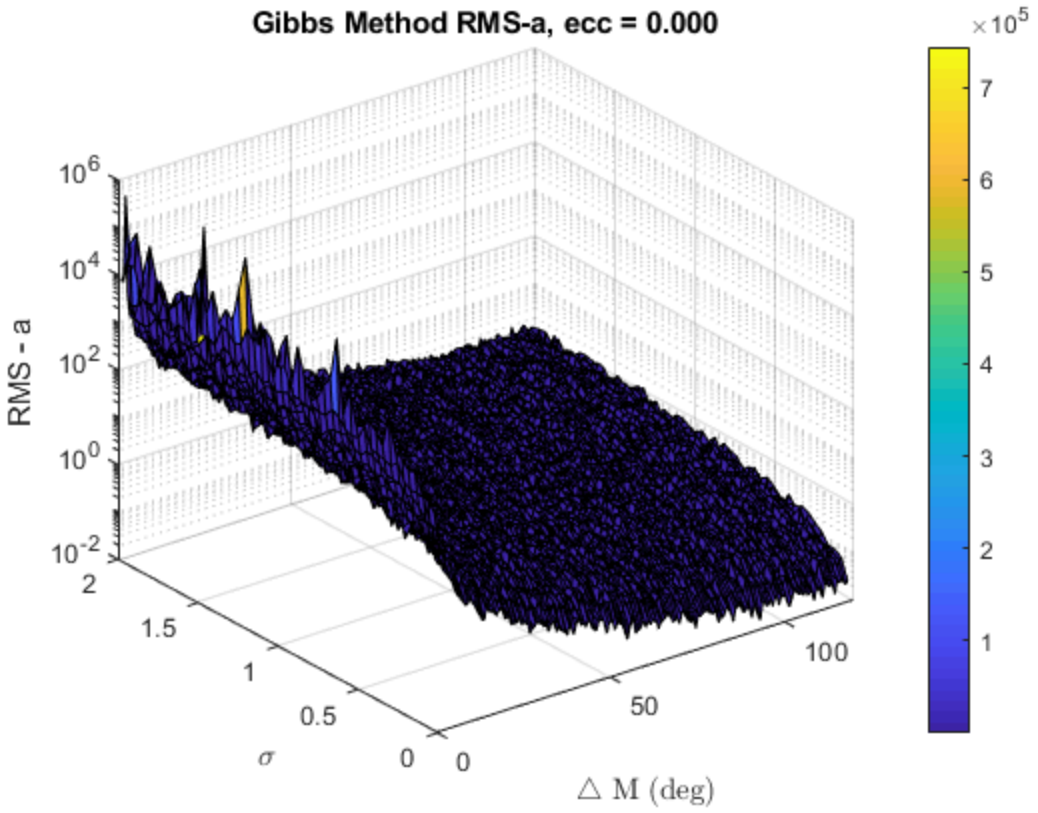
```
%set(gca,'zscale','log')
tiS=sprintf('tf all Herrick Gibbs RMS-a, ecc = %.3f',e);
title(tiS)
colorbar


figure(20)
hold on
%surf(MAdistA(offSet:end),(1:1:3),tdiffA(:,offSet:end))
plot(MAdistA(offSet:end),mastM(offSet:end,1))
plot(MAdistA(offSet:end),mastM(offSet:end,2))
plot(MAdistA(offSet:end),mastM(offSet:end,3))
ylabel('m1, m2, m3')
xlabel('$\bigtriangleup$ M (deg)','Interpreter','latex')
%zlabel('tf')
%set(gca,'zscale','log')
tiS=sprintf('tf all Herrick Gibbs RMS-a, ecc = %.3f',e);
title(tiS)
legend('m1','m2','m3')

figure(21)
hold on
%surf(MAdistA(offSet:end),(1:1:3),tdiffA(:,offSet:end))
plot(TAdistA(offSet:end),tdiffA(1,offSet:end))
plot(TAdistA(offSet:end),tdiffA(2,offSet:end))
plot(TAdistA(offSet:end),tdiffA(3,offSet:end))
ylabel('t21, t32, t31')
xlabel('$\bigtriangleup$ TA (deg)','Interpreter','latex')
%zlabel('tf')
%set(gca,'zscale','log')
tiS=sprintf('tf all Herrick Gibbs RMS-a, ecc = %.3f',e);
title(tiS)
legend('t21','t32','t31')
%colorbar


%{
figure(11)
surf(MAdistA(offSet:end),sigmaA(2:end),rmsGv2(2:end,offSet:end));
ylabel('\sigma')
xlabel('$\bigtriangleup$ M (deg)','Interpreter','latex')
zlabel('RMS-v2')
set(gca,'zscale','log')
tiS=sprintf('RMS-Gibbs');
title(tiS)
colorbar

figure(12)
surf(MAdistA(offSet:end),sigmaA(2:end),rmsHGv2(2:end,offSet:end))
ylabel('\sigma')
xlabel('$\bigtriangleup$ M (deg)','Interpreter','latex')
zlabel('RMS-v2')
set(gca,'zscale','log')
tiS=sprintf('RMS-Herrick');
```

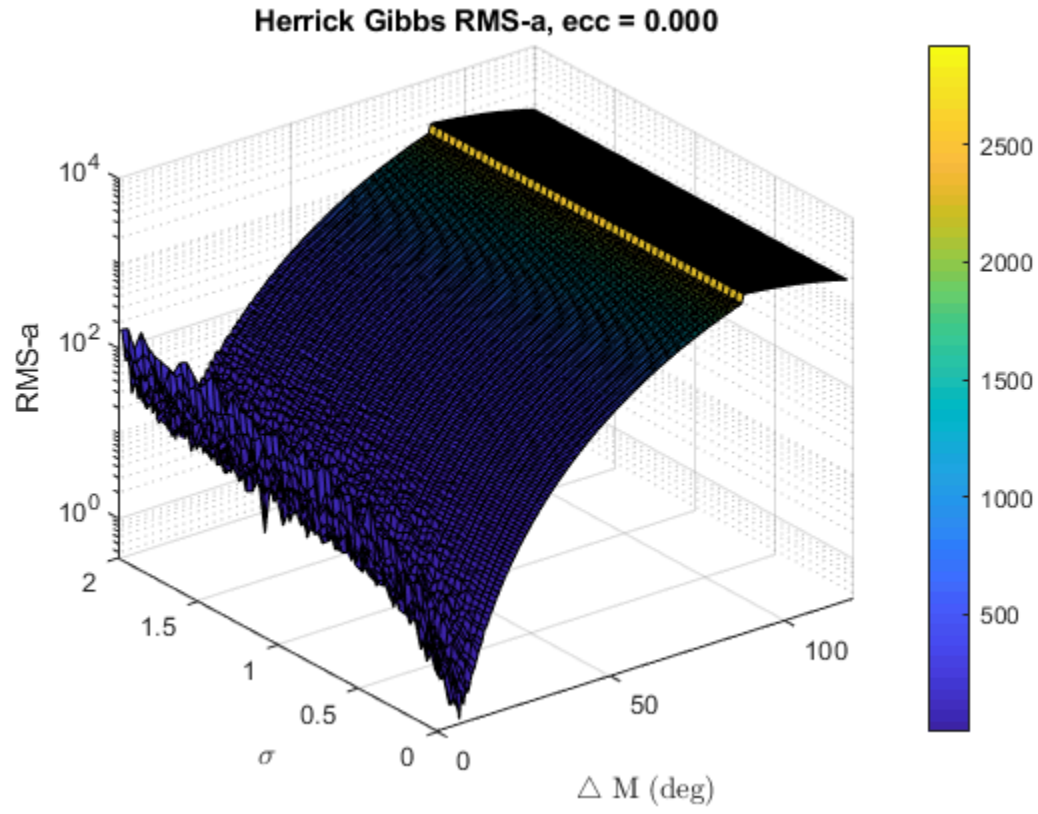
```
title(tiS)
colorbar
%}

rmsInd=[];
rmsBest=[];
for i=1:length(rmsCir(:,1))
    for j=1:length(rmsCir(1,:))
        if rmsCir(i,j)>rmsMH(i,j)
            rmsBest(i,j)=rmsMH(i,j);
            rmsInd(i,j)=1; % for HH
            rmsBestV2(i,j)=rmsHGv2(i,j);
        else
            rmsBest(i,j)=rmsCir(i,j);
            rmsBestV2(i,j)=rmsGv2(i,j);
            rmsInd(i,j)=0; % for gib
        end
    end
end
rmsDiff=rmsCir-rmsMH;

figure(3)
surf(MAdistA(offSet:end),sigmaA(2:end),rmsBest(2:end,offSet:end));
ylabel('\sigma')
xlabel('$\bigtriangleup$ M (deg)','Interpreter','latex')
zlabel('RMS-a')
set(gca,'zscale','log')
tiS=sprintf('Best RMS-a, ecc = %.3f',e);
title(tiS)
colorbar
%{
figure(13)
surf(MAdistA(offSet:end),sigmaA(2:end),rmsBestV2(2:end,offSet:end));
ylabel('\sigma')
xlabel('$\bigtriangleup$ M (deg)','Interpreter','latex')
zlabel('RMS-v2')
set(gca,'zscale','log')
tiS=sprintf('RMS-Best');
title(tiS)
colorbar
%}
figure(4)
hold on
s1 =
 surf(MAdistA(offSet:end),sigmaA(2:end),rmsCir(2:end,offSet:end),'FaceAlpha',0.5)
s2 =
 surf(MAdistA(offSet:end),sigmaA(2:end),rmsMH(2:end,offSet:end),'FaceAlpha',0.5)
s1.EdgeColor = 'none';
s2.EdgeColor = 'none';

ylabel('\sigma')
xlabel('$\bigtriangleup$ M (deg)','Interpreter','latex')
zlabel('RMS')
%set(gca,'zscale','log')
```

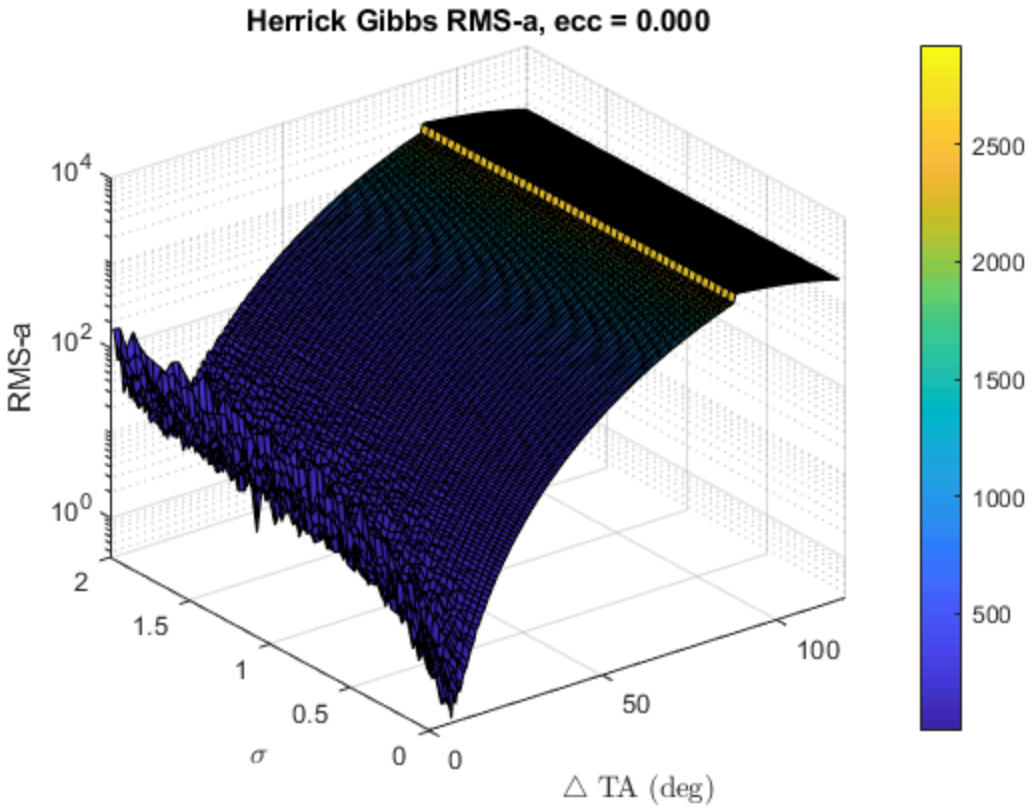
```
%plot([1:5],rmsP)
tiS=sprintf('both plots');
title(tiS)
%colorbar
set(gca,'zscale','log')
view(17,22)
grid on

%{
figure(5)
surf(MAdistA(offSet:end),sigmaA(2:end),rmsDiff(2:end,offSet:end));
ylabel('\sigma')
xlabel('\bigtriangleup M (deg)')
zlabel('RMS')
zlim([-300, 1000])
%set(gca,'zscale','log')
%plot([1:5],rmsP)
tiS=sprintf('RMS diff (circ-HH)');
title(tiS)
colorbar
%}

figure(7)
hold on
surf(TAdistA(offSet:end),sigmaA(2:end),rmsCir(2:end,offSet:end),'FaceAlpha',0.5)
surf(TAdistA(offSet:end),sigmaA(2:end),rmsMH(2:end,offSet:end),'FaceAlpha',0.5)

ylabel('\sigma')
xlabel('$\bigtriangleup$ M (deg)','Interpreter','latex')
zlabel('RMS')
set(gca,'zscale','log')

tiS=sprintf('both plots');
title(tiS)
grid on
view(17,22)

vct=[1:1:22];
figure(8)
contour(rmsBest(:,2:end),vct,'ShowText','on')
ylabel('\sigma')
xlabel('$\bigtriangleup$ M (deg)','Interpreter','latex')
grid on

for j=1:length(rmsDiff(1,:))
    for i=1:length(rmsDiff(:,1))
        if rmsDiff(i,j)<0
            pntBet(j)=i;
            break;
        end
    end
end
```

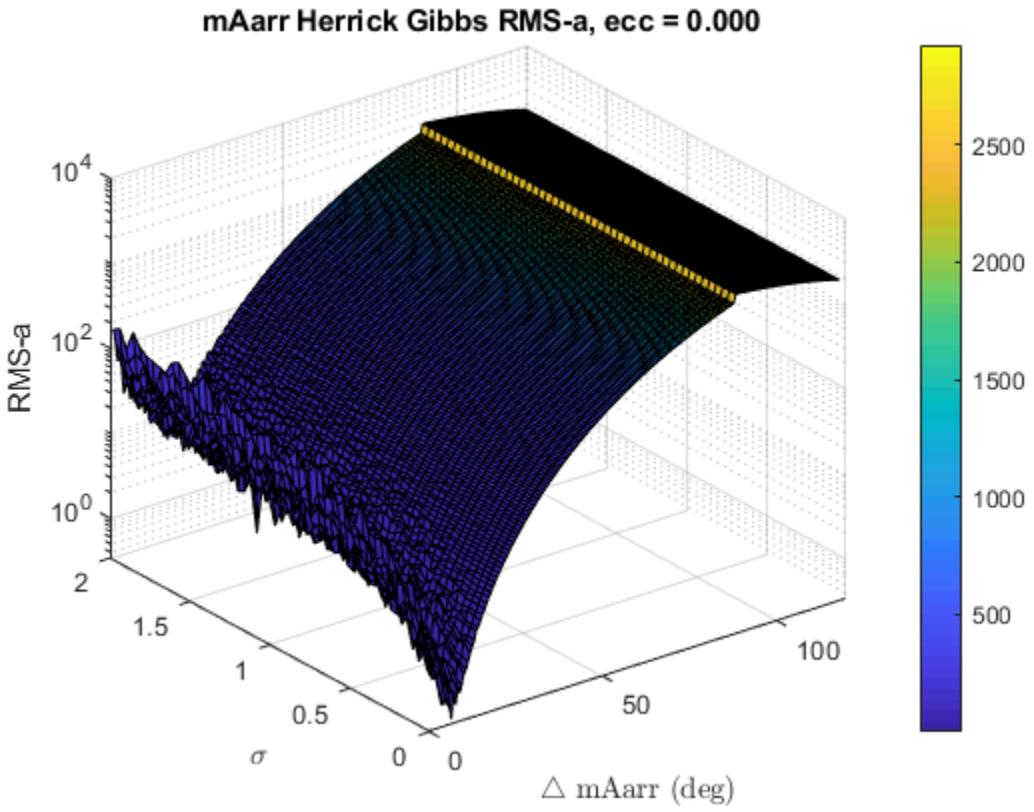
```
for i=1:length(rmsDiff(:,1))
    for j=1:length(rmsDiff(1,:))
        if rmsDiff(i,j)<0
            pntBet2(i)=j;
            break;
        end
    end
end
```

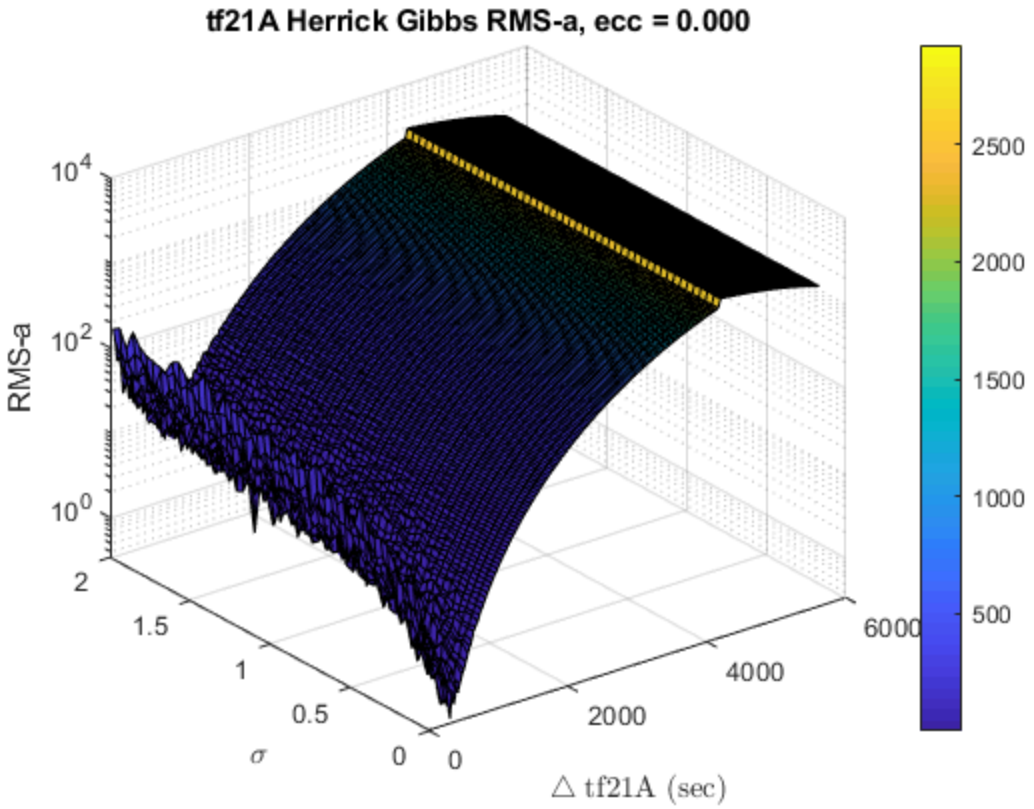
```
s1 =

  Surface with properties:

      EdgeColor: [0 0 0]
      LineStyle: '-'
      FaceColor: 'flat'
  FaceLighting: 'flat'
      FaceAlpha: 0.5000
          XData: [1×200 double]
          YData: [1×49 double]
          ZData: [49×200 double]
          CData: [49×200 double]

  Use GET to show all properties


s2 =

  Surface with properties:

      EdgeColor: [0 0 0]
      LineStyle: '-'
      FaceColor: 'flat'
  FaceLighting: 'flat'
      FaceAlpha: 0.5000
          XData: [1×200 double]
          YData: [1×49 double]
          ZData: [49×200 double]
          CData: [49×200 double]

  Use GET to show all properties
```

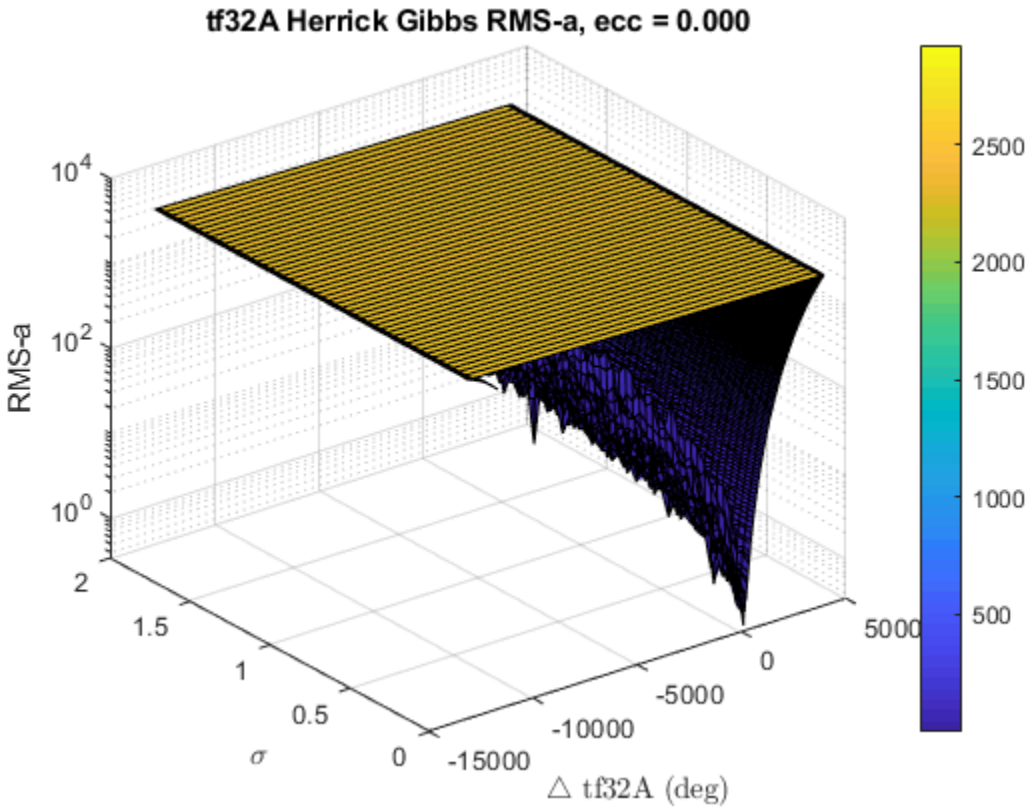

Gibbs Method RMS-a, ecc = 0.000
RMS - a
10^6
10^4
10^2
10^0
10^{-2}
2
1.5
1
0.5
0
σ
0
50
100
△ M (deg)
$\times 10^5$
7
6
5
4
3
2
1

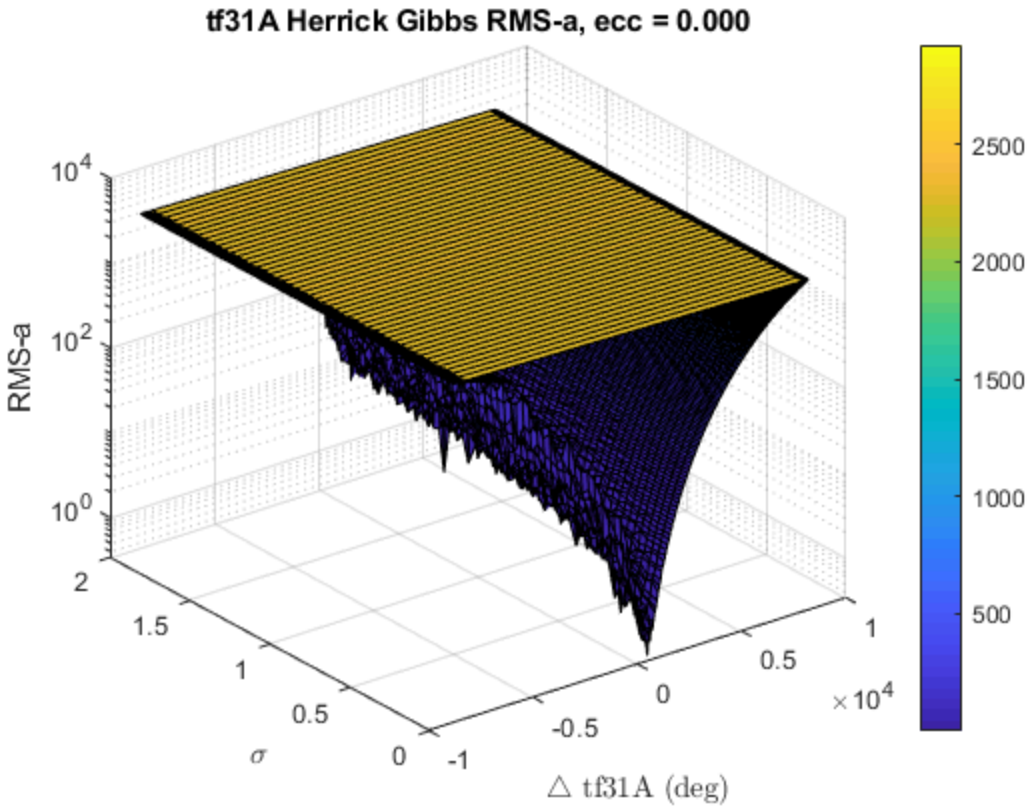

Herrick Gibbs RMS-a, ecc = 0.000
RMS-a
10^4
10^2
10^0
2
1.5
1
0.5
0
σ
0
50
100
△ M (deg)
2500
2000
1500
1000
500

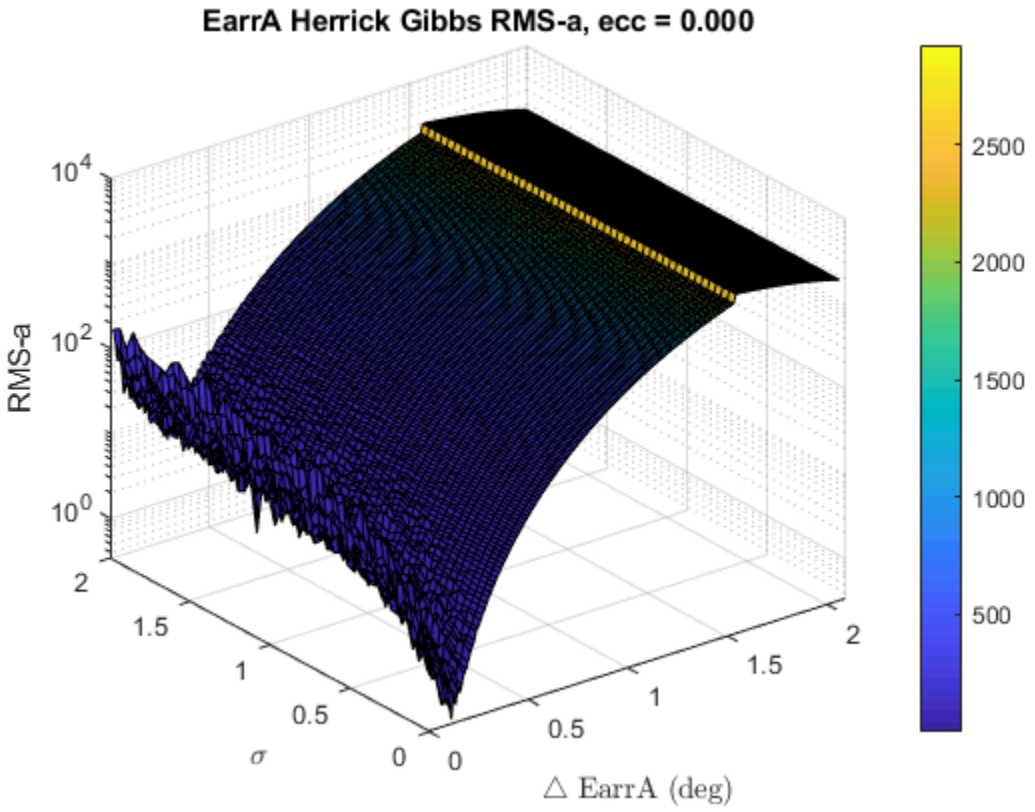

Herrick Gibbs RMS-a, ecc = 0.000
RMS-a
10^4
10^2
10^0
2
1.5
1
0.5
0
σ
0
50
100
△ TA (deg)
2500
2000
1500
1000
500

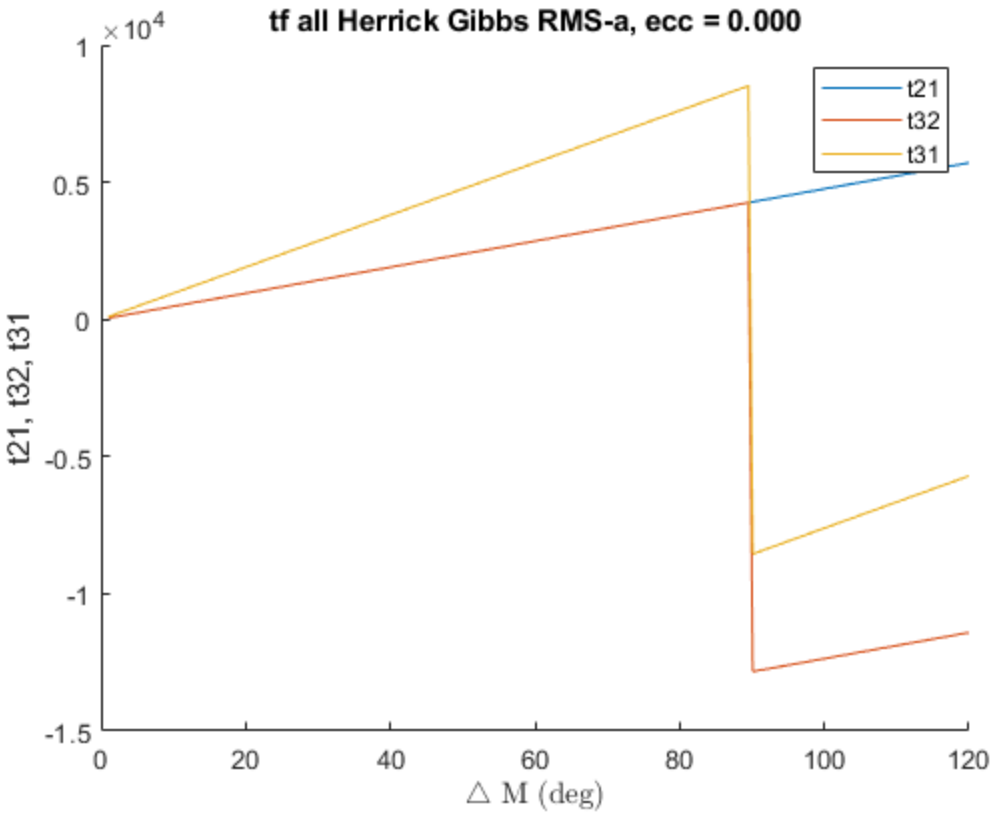

mAarr Herrick Gibbs RMS-a, ecc = 0.000
RMS-a
10^4
10^2
10^0
2
1.5
1
0.5
0
σ
0
50
100
△ mAarr (deg)
2500
2000
1500
1000
500

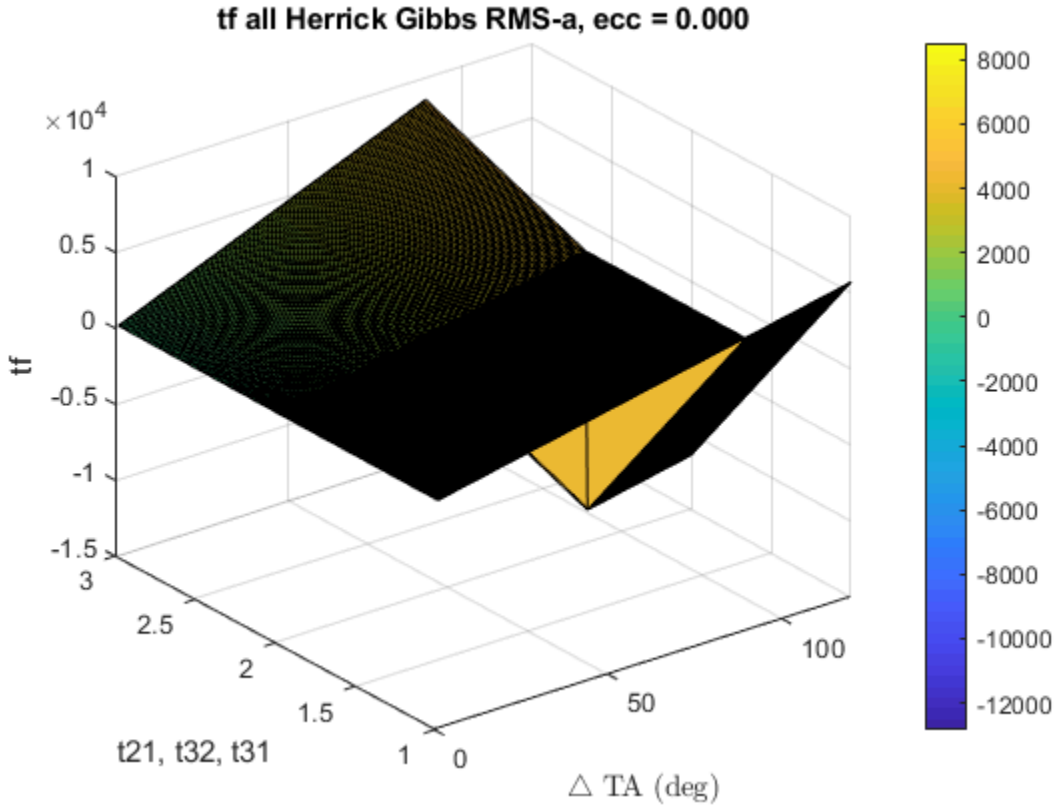

tf21A Herrick Gibbs RMS-a, ecc = 0.000
RMS-a
10^4
10^2
10^0
2
1.5
1
0.5
0
σ
0
2000
4000
6000
$\triangle$ tf21A (sec)
2500
2000
1500
1000
500

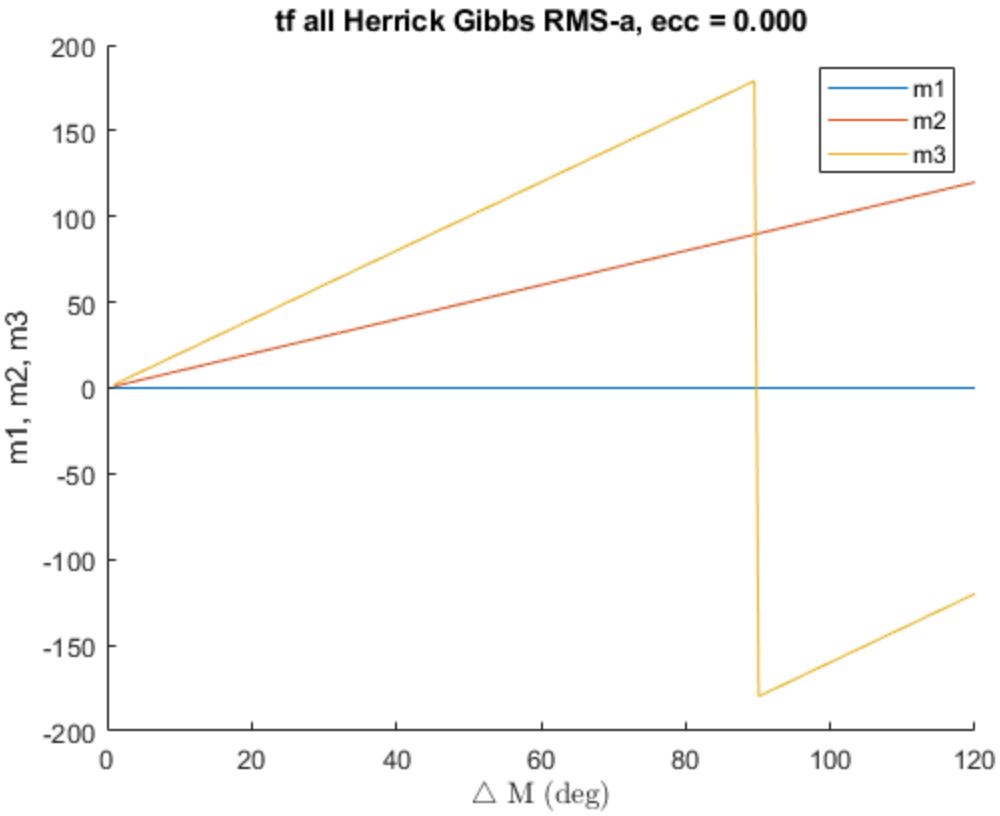

tf32A Herrick Gibbs RMS-a, ecc = 0.000
RMS-a
10^4
10^2
10^0
2
1.5
1
0.5
0
σ
-15000
-10000
-5000
0
5000
$\triangle$ tf32A (deg)
2500
2000
1500
1000
500

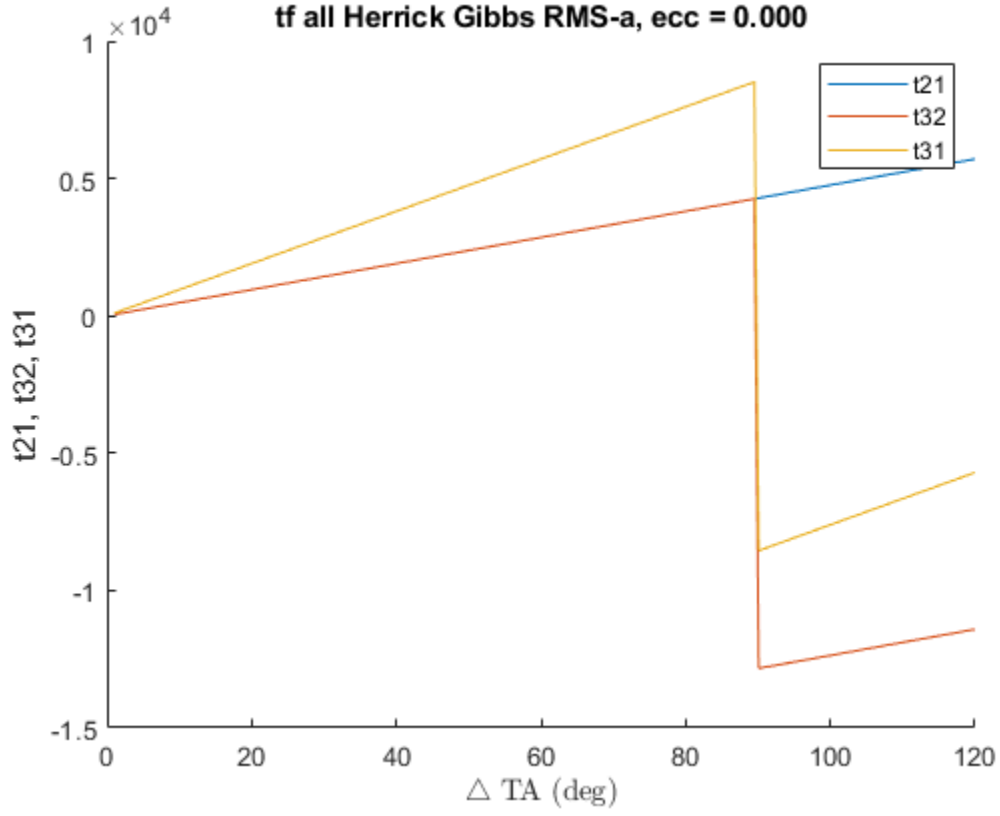

tf31A Herrick Gibbs RMS-a, ecc = 0.000
RMS-a
10^4
10^2
10^0
2
1.5
1
0.5
0
σ
-1
-0.5
0
0.5
1
$\times 10^4$
$\triangle$ tf31A (deg)
2500
2000
1500
1000
500

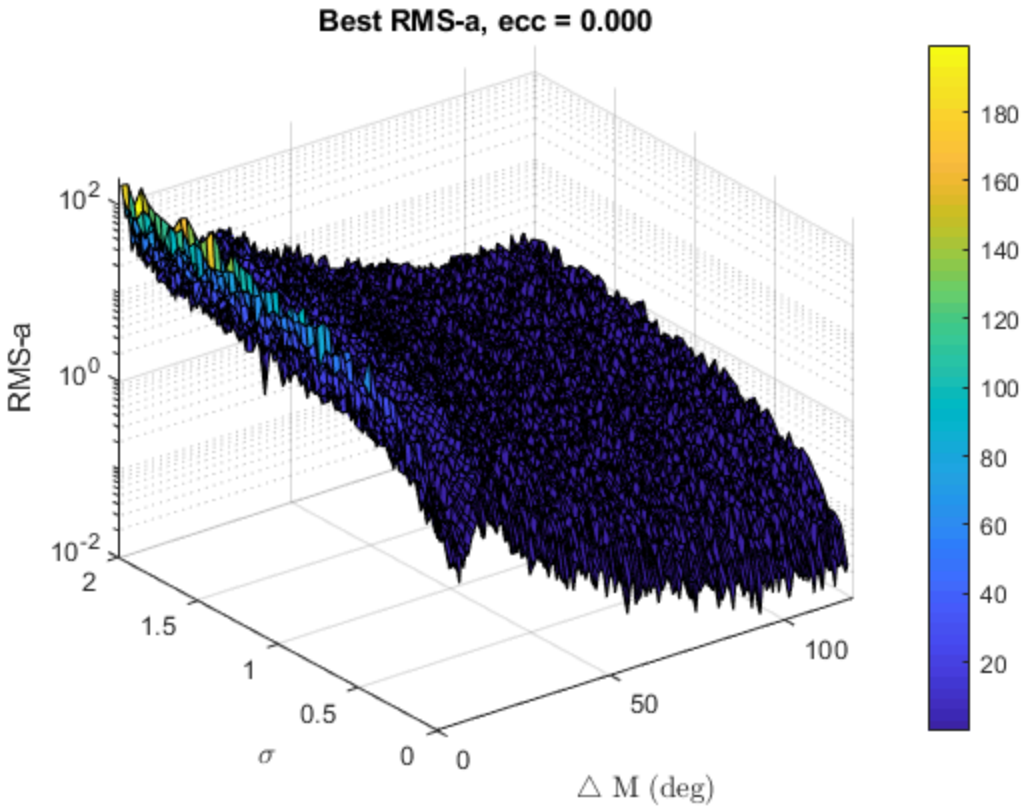

EarrA Herrick Gibbs RMS-a, ecc = 0.000
RMS-a
10^4
10^2
10^0
2
1.5
1
0.5
0
σ
0
0.5
1
1.5
2
$\triangle$ EarrA (deg)
2500
2000
1500
1000
500

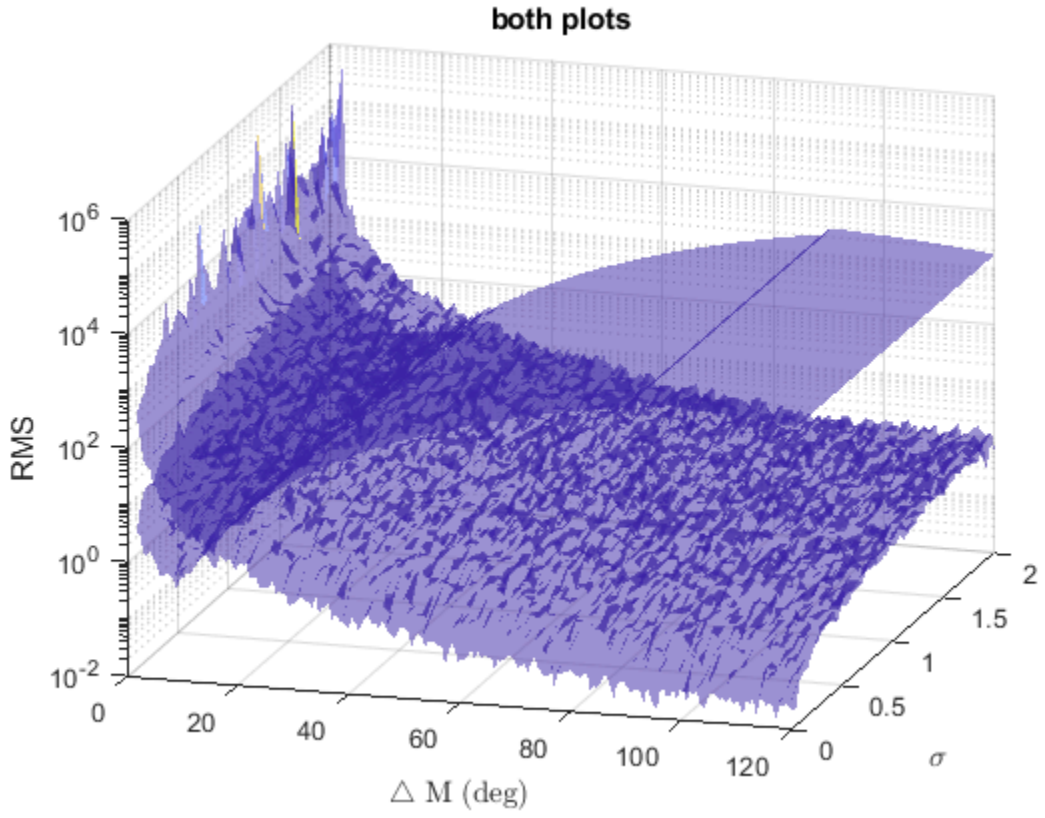

tf all Herrick Gibbs RMS-a, ecc = 0.000
×10^4
1
0.5
0
-0.5
-1
-1.5
t21, t32, t31
0
20
40
60
80
100
120
△ M (deg)
t21
t32
t31

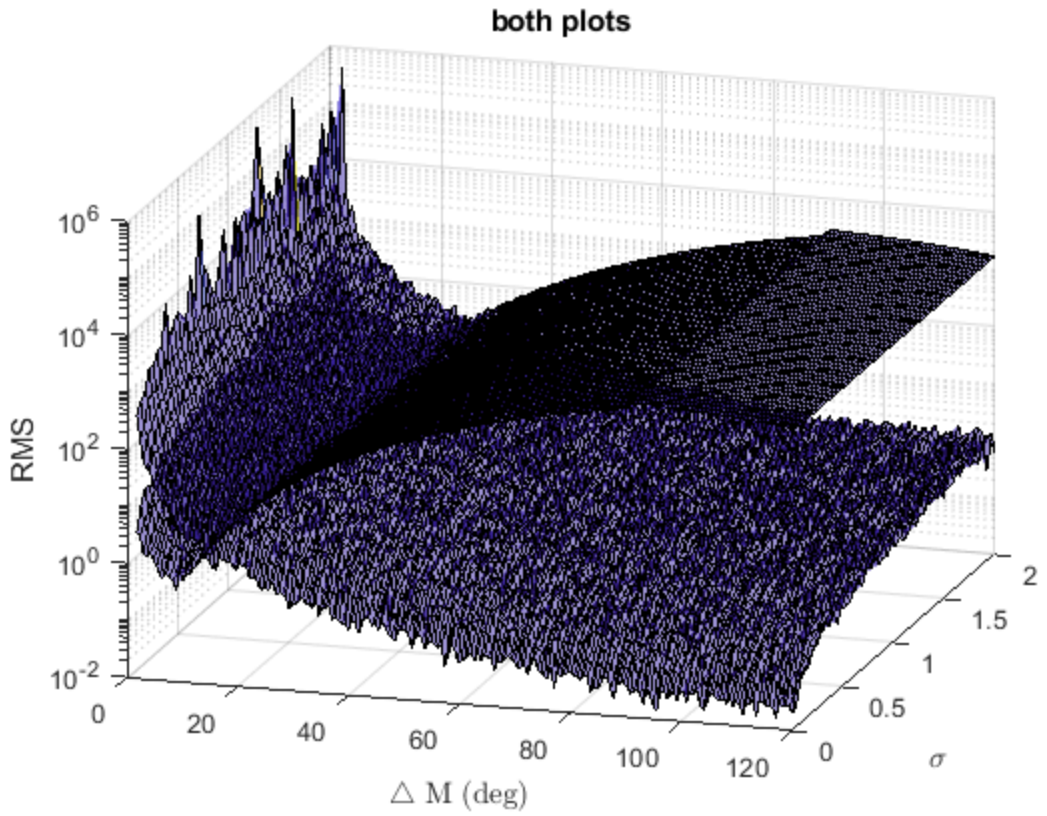

tf all Herrick Gibbs RMS-a, ecc = 0.000
×10^4
1
0.5
0
-0.5
-1
-1.5
tf
3
2.5
2
1.5
1
t21, t32, t31
0
50
100
△ TA (deg)
8000
6000
4000
2000
0
-2000
-4000
-6000
-8000
-10000
-12000

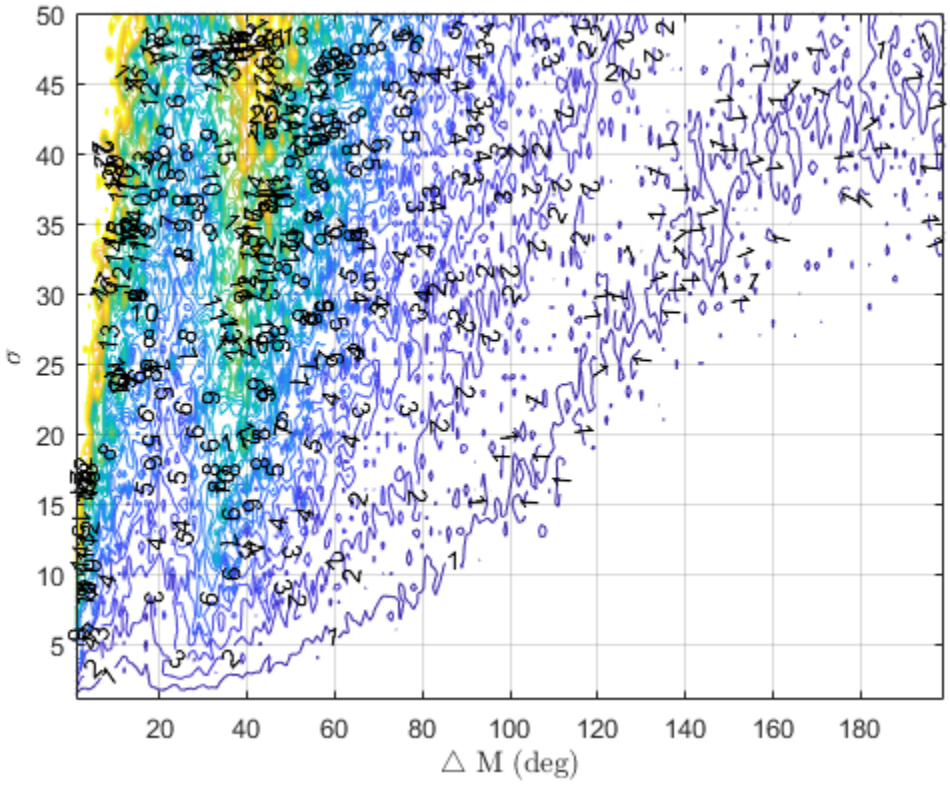

tf all Herrick Gibbs RMS-a, ecc = 0.000
m1
m2
m3
m1, m2, m3
200
150
100
50
0
-50
-100
-150
-200
0
20
40
60
80
100
120
△ M (deg)


tf all Herrick Gibbs RMS-a, ecc = 0.000
×10^4
t21
t32
t31
t21, t32, t31
1
0.5
0
-0.5
-1
-1.5
0
20
40
60
80
100
120
△ TA (deg)


Best RMS-a, ecc = 0.000
RMS-a
10^2
10^0
10^{-2}
2
1.5
1
0.5
0
σ
0
50
100
$\triangle$ M (deg)
180
160
140
120
100
80
60
40
20


both plots
RMS
10^6
10^4
10^2
10^0
10^{-2}
0
20
40
60
80
100
120
$\triangle$ M (deg)
0
0.5
1
1.5
2
σ


both plots
RMS
10^6
10^4
10^2
10^0
10^{-2}
0
20
40
60
80
100
120
△ M (deg)
0
0.5
1
1.5
2
σ


σ
50
45
40
35
30
25
20
15
10
5
20
40
60
80
100
120
140
160
180
△ M (deg)

plot pntB

```
figure(9)
plot(pntBet2,sigmaA,'-s')
ylabel('\sigma')
xlabel('delta M')
grid on


figure(10)
hold on;
% Define the input grid
%[x, y] = meshgrid(linspace(-1, 1));
% Calculate the two surfaces
%z1 = y.^2 + 2*x;
%z2 = 2*y.^3 - x.^2;
% Visualize the two surfaces
z1=rmsCir(2:end,offSet:end);
z2=rmsMH(2:end,offSet:end);
x=MAdistA(offSet:end);
y=sigmaA(2:end);
surface(MAdistA(offSet:end),sigmaA(2:end),z1,'FaceColor', [0.5 1.0
 0.5], 'EdgeColor', 'none');
surf(MAdistA(offSet:end),sigmaA(2:end),z2, 'FaceColor', [1.0 0.5
 0.0], 'EdgeColor', 'none');
ylabel('\sigma')
xlabel('$\bigtriangleup$ M (deg)','Interpreter','latex')
zlabel('RMS - a ')
set(gca,'zscale','log')
tiS=sprintf('Gibbs Method RMS-a, ecc = %.3f',e);
title(tiS)
%colorbar
grid on

%surface(x, y, z1, 'FaceColor', [0.5 1.0 0.5], 'EdgeColor', 'none');
%surface(x, y, z2, 'FaceColor', [1.0 0.5 0.0], 'EdgeColor', 'none');
%view(3);
camlight;
view(17,22)
%axis vis3d
% Take the difference between the two surface heights and find the
 contour
% where that surface is zero.
zdiff = z1 - z2;
C = contours(x, y, zdiff, [0 0]);
% Extract the x- and y-locations from the contour matrix C.
xL = C(1, 2:end);
yL = C(2, 2:end);
% Interpolate on the first surface to find z-locations for the
 intersection
% line.
zL = interp2(x, y, z1, xL, yL);
% Visualize the line.
```

```
line(xL, yL, zL, 'Color', 'k', 'LineWidth', 3);

%colorbar

%{
OrbType='elp';
[rmsP] = RMS_COE(numbSamp,sigmaA,TAdistA,coeM,mu,OrbType);
rmsEc=rmsP;
%}
%{
%% Plot Section
%save('wksp10')
offSet=20;%20
figure(1)
surf(TAdistA(offSet:end),sigmaA(2:end),rmsP(2:end,offSet:end));
ylabel('sigma')
xlabel('\tri M (deg)')
zlabel('RMS')
set(gca,'zscale','log')
%plot([1:5],rmsP)
tiS=sprintf('RMS');
title(tiS)
colorbar

%x=[20,17,23]; xmax=20; x(x>xmax)=xmax
rmsPMax=100;
rmsP_clip=rmsP;
rmsP_clip(rmsP_clip>rmsPMax) = rmsPMax;
figure(11)
surf(TAdistA,sigmaA(2:end),rmsP_clip(2:end,:))
ylabel('sigma')
xlabel('\tri M (deg)')
zlabel('RMS')
set(gca,'zscale','log')
%plot([1:5],rmsP)
tiS=sprintf('RMS clip for  %d samples',numbSamp);
title(tiS)
colorbar

figure(12)
Z=rmsP(2:end,offSet:end);
x=TAdistA(offSet:end);
y=sigmaA(2:end);
[dfdx,dfdy] = gradient(Z);
surf(x,y,Z,sqrt(dfdx.^2 + dfdy.^2))
colorbar
tiS=sprintf('grad RMS for  %d samples',numbSamp);
title(tiS)

figure(13)
Z=rmsP_clip;
x=TAdistA;
y=sigmaA;
[dfdx,dfdy] = gradient(Z);
```

```
surf(x,y,Z,sqrt(dfdx.^2 + dfdy.^2))
colorbar
tiS=sprintf('grad RMS clip for  %d samples',numbSamp);
title(tiS)


figure(14)
subplot(2,1,1);
surf(TAdistA,sigmaA(2:end),rmsP_clip(2:end,:))
ylabel('sigma')
xlabel('\tri M (deg)')
zlabel('RMS')
set(gca,'zscale','log')
%plot([1:5],rmsP)
tiS=sprintf('RMS clip for  %d samples',numbSamp);
title(tiS)
colorbar

subplot(2,1,2);
Z=rmsP_clip;
x=TAdistA;
y=sigmaA;
[dfdx,dfdy] = gradient(Z);
surf(x,y,Z,sqrt(dfdx.^2 + dfdy.^2))
colorbar
tiS=sprintf('grad RMS clip for  %d samples',numbSamp);
title(tiS)

figure(2)
vp1=[(1:1:9),(10:2:28),(30:5:100)];
%contour(TAdistA(offSet:end),sigmaA,rmsP(:,offSet:end),150,'ShowText','on')
contour(TAdistA(offSet:end),sigmaA,rmsP(:,offSet:end),vp1,'ShowText','on')
%contour3(TAdistA,sigmaA,rmsP)
ylabel('sigma')
xlabel('\tri M (deg)')
zlabel('RMS')
%set(gca,'zscale','log')
%plot([1:5],rmsP)
tiS=sprintf('RMScount');
title(tiS)
vp=[1,10,50,100,500];
offSet=1;
figure(3)
hold on;
contour(TAdistA(offSet:end),sigmaA,rmsP(:,offSet:end),vp,'ShowText','on')
%contour3(TAdistA,sigmaA,rmsP)
ylabel('sigma')
xlabel('\tri M (deg)')
zlabel('RMS')
grid on
%set(gca,'zscale','log')
%plot([1:5],rmsP)
tiS=sprintf('cotours for %d samples',numbSamp);
title(tiS)
```

```
figure(4)
hold on;
contour(TAdistA(offSet:end),sigmaA,rmsP(:,offSet:end),vp,'-
r','ShowText','on')

%contour3(TAdistA,sigmaA,rmsP)
ylabel('sigma')
xlabel('\tri M (deg)')
zlabel('RMS')
grid on
%set(gca,'zscale','log')
%plot([1:5],rmsP)
tiS=sprintf('cotours for %d samples',numbSamp);
title(tiS)



figure(5)
hold on
vp2=[(1:1:9),(10:5:50)];
%contour(TAdistA(offSet:end),sigmaA,rmsP(:,offSet:end),150,'ShowText','on')

contour(TAdistA(offSet:end),sigmaA,rmsCir(:,offSet:end),vp2,'-
k','ShowText','on')
%}
%{
function [rmsM] = RMS_COE(numbSamp,sigmaA,TAdistA,coeM,mu,OrbType)
    if strcmp(OrbType,'circ')%OrbType=='circ'
        fgid=101;
    else
        fgid=99;
    end
    figure(fgid)

        %addpoints(h,xdrw(xc),tdrw(xc),'-r');
        %addpoints(hTot,xdrw(xc),tTot,'-k');
        %addpoints(hCurr,xdrw(xc),tr,'-b');
    h = animatedline('Color','r');
    hTot = animatedline('Color','k');
    hCurr= animatedline('Color','b');
    legend('red: timeLeft est ','black: total time est','blue: current
 time')
    grid on
    xdrw=1:1:length(sigmaA);
    tdrw=[];
    xc=1;
    %numbSamp=100; % set numbSamp
    %sigmaA=linspace(0,20,200/2); % set sigma (km) to go over
    %TAdistA=linspace(1,120,120/2); % set TA dist to go over
    coe=coeM(:,1:6);
    coeLp=coe;
    aReal=coeM(1,7); % set aReal
```

```
    %fprintf('%.3f\n',tic);
    tic
    fprintf('%.3f\n',toc);

    for sigmaC=1:length(sigmaA)

        sigma=sigmaA(sigmaC);
        for TAdistC=1:length(TAdistA)%120
            TAdist=TAdistA(TAdistC);
            TAarr=(0:TAdist:2*TAdist)';
            TAarr=TAarr*pi/180;

            coeLp(:,6)=TAarr;
            for i=1:3
                [r, v] = sv_from_coe(coeLp(i,:),mu);
                rn = normrnd(0,sigma,[numbSamp,3]);
                rRand(1:numbSamp,1:3)=r(1:3)+rn(1:numbSamp,1:3);
%/1000;
                rMast(:,:,i)=rRand;
            end
            for k=1:numbSamp
               r1=rMast(k,:,1);
               r2=rMast(k,:,2);
               r3=rMast(k,:,3);
               [r2p,v2p] = gibbs_Fun(r1,r2,r3,mu);
               coe = coe_from_sv(r2p,v2p,mu);
               a(k,1)=coe(7);
            end
           aR=ones(length(a),1)*aReal;
           rmsP(sigmaC,TAdistC)=sqrt(mean((a(:)-aR).^2));
        end
        fprintf('%d of %d, for %s,  ',sigmaC,length(sigmaA),OrbType);
        tr=toc;
        pctR=sigmaC/length(sigmaA);
        tTot=tr/pctR;
        tLeft=tTot-tr;
        fprintf('%.1f sec elaps, %.1f est tot, %.1f left
\n',tr,tTot,tLeft);
        tdrw=[tdrw,tLeft];
        addpoints(h,xdrw(xc),tdrw(xc));
        addpoints(hTot,xdrw(xc),tTot);
        addpoints(hCurr,xdrw(xc),tr);
          % hTot = animatedline;
    %hCurr= animatedline;

        xc=xc+1;
        drawnow

    end
    rmsM=rmsP;
end
%}
```

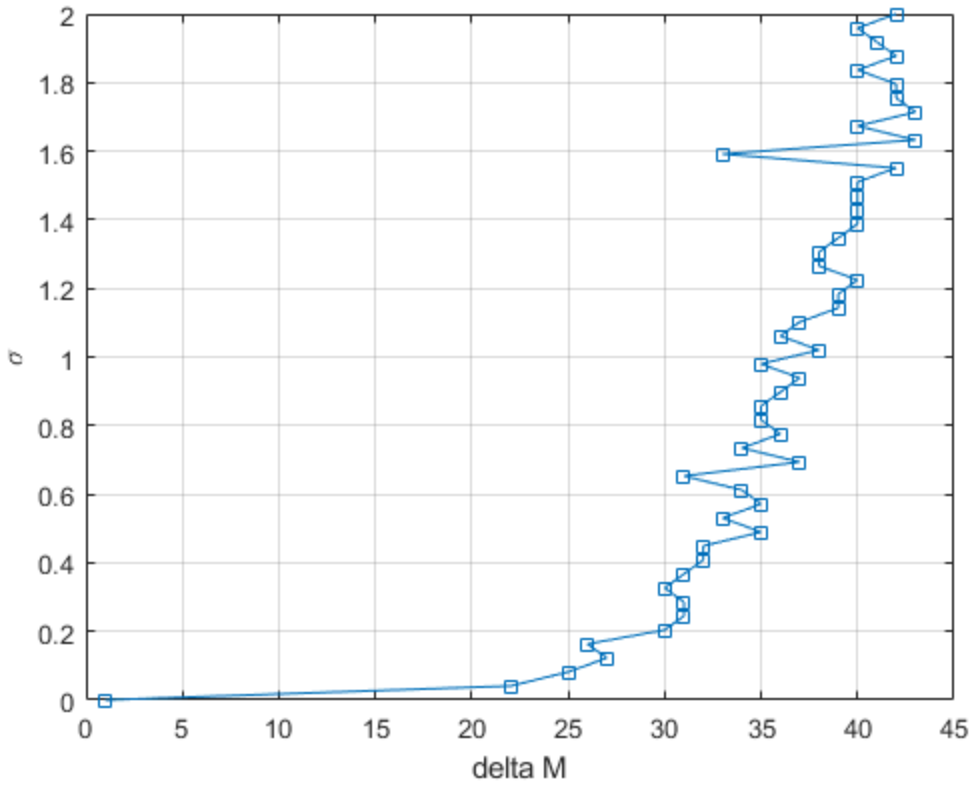
```
function [rmsM,rmsMH,rmsGv2,rmsHGv2] =
 RMS_G_HG(numbSamp,sigmaA,TAdistA,coeM,mu,OrbType,rp)
    if strcmp(OrbType,'circ')%OrbType=='circ'
        fgid=101;
    else
        fgid=99;
    end
    figure(fgid)

    h = animatedline('Color','r');
    hTot = animatedline('Color','k');
    hCurr= animatedline('Color','b');
    legend('red: timeLeft est ','black: total time est','blue: current
 time')
    grid on
    xdrw=1:1:length(sigmaA);
    tdrw=[];
    xc=1;

    coe=coeM(:,1:6);
    coeLp=coe;
    aReal=coeM(1,7); % set aReal

    tic
    fprintf('%.3f\n',toc);

    tFlightF=2*pi/(2*pi*rp^(3/2) /sqrt(mu));
    tFlightF=1/tFlightF;
    Tt=2*pi/( (mu^2 *(1-coeM(1,2)^2)^(3/2))/ coeM(1,1)^3);

    for sigmaC=1:length(sigmaA)

        sigma=sigmaA(sigmaC);
        for TAdistC=1:length(TAdistA)%120
            TAdist=TAdistA(TAdistC);
            TAarr=(0:TAdist:2*TAdist)';
            E=2*atan( sqrt(1-coeM(1,2))/sqrt(1+coeM(1,2))
 *tan(.5*TAarr*pi/180));
            MAarr= E-coeM(1,2)*sin(E);

            tf21=(Tt/(2*pi))*(MAarr(2)-MAarr(1));%*pi/180;
            tf32=(Tt/(2*pi))*(MAarr(3)-MAarr(2));%*pi/180;
            tf31=(Tt/(2*pi))*(MAarr(3)-MAarr(1));%*pi/180;
            %coeM(i,:)=[h e RAAN inc w TA a];
            %taF21=sqrt(coeM(1,7)^3 /mu)*(E(2)-E(1)-
coeM(1,2)*sin(E(2)-E(1)));
            for kCount=1:3
                t_hGibbs(kCount)=sqrt(coeM(1,7)^3 /mu)*(E(kCount)-
coeM(1,2)*sin(E(kCount)));
            end
            %keyboard

            TAarr=TAarr*pi/180;
```

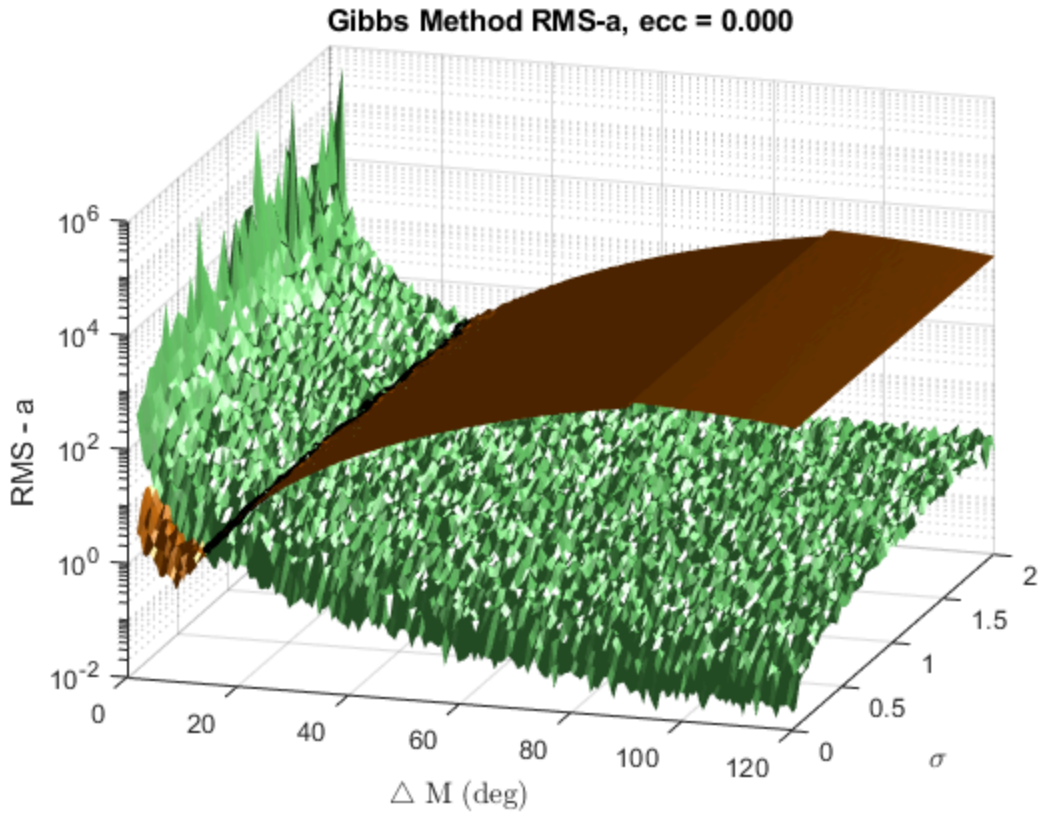
```
            coeLp(:,6)=TAarr;
            for i=1:3
                [r, v] = sv_from_coe(coeLp(i,:),mu);
                rn = normrnd(0,sigma,[numbSamp,3]);

 rRand(1:numbSamp,1:3)=r(1:3)+rn(1:numbSamp,1:3);%/1000;
                 %rRand(1:numbSamp,1:3)=r(1:3)+0.05*ones(numbSamp,3);
%rn(1:numbSamp,1:3);%/1000;
                rMast(:,:,i)=rRand;
            end
            for k=1:numbSamp
               r1=rMast(k,:,1);
               r2=rMast(k,:,2);
               r3=rMast(k,:,3);
               v2HH=-tf32*( 1/(tf21*tf31) + mu/
(12*norm(r1)^3))*r1+(tf32-tf21)*( (1/(tf21 *tf32)) + mu/
(12*norm(r2)^3))*r2+ tf21*( 1/(tf32*tf31) + mu/(12*norm(r3)^3))*r3;

               [r2p,v2p] = gibbs_Fun(r1,r2,r3,mu);
                v2Gm(k,1)=norm(v2p);
               coe = coe_from_sv(r2p,v2p,mu);
               a(k,1)=coe(7);
               coeHH=coe_from_sv(r2,v2HH,mu);
               aHH(k,1)=coeHH(7);
               v2HM(k,1)=norm(v2HH);
            end
            [~, v2] = sv_from_coe(coeLp(2,:),mu);
           aR=ones(length(a),1)*aReal;
           v2R=ones(length(a),1)*norm(v2);
           rmsGv2(sigmaC,TAdistC)=sqrt(mean((v2Gm(:)-v2R).^2));
           rmsHGv2(sigmaC,TAdistC)=sqrt(mean((v2HM(:)-v2R).^2));
           rmsP(sigmaC,TAdistC)=sqrt(mean((a(:)-aR).^2));
           rmsHH(sigmaC,TAdistC)=sqrt(mean((aHH(:)-aR).^2));
          if TAdist>38.2*pi/180&&TAdist<38.6*pi/180
              % keyboard;
           end
           if TAdistC==127
              % keyboard;
           end
        end
        fprintf('%d of %d, for %s,  ',sigmaC,length(sigmaA),OrbType);
        tr=toc;
        pctR=sigmaC/length(sigmaA);
        tTot=tr/pctR;
        tLeft=tTot-tr;
        fprintf('%.1f sec elaps, %.1f est tot, %.1f left
\n',tr,tTot,tLeft);
        tdrw=[tdrw,tLeft];
        addpoints(h,xdrw(xc),tdrw(xc));
        addpoints(hTot,xdrw(xc),tTot);
        addpoints(hCurr,xdrw(xc),tr);
```

```
        xc=xc+1;
        drawnow

    end
    rmsM=rmsP;
    rmsMH=rmsHH;
end


function [rmsGv2,rmsHGv2,rmsME] =
 checkDisc(numbSamp,sigmaA,TAdistA,coeM,mu,OrbType,rp)
    if strcmp(OrbType,'circ')%OrbType=='circ'
        fgid=101;
    else
        fgid=99;
    end
    figure(fgid)

    h = animatedline('Color','r');
    hTot = animatedline('Color','k');
    hCurr= animatedline('Color','b');
    legend('red: timeLeft est ','black: total time est','blue: current
 time')
    grid on
    xdrw=1:1:length(sigmaA);
    tdrw=[];
    xc=1;

    coe=coeM(:,1:6);
    coeLp=coe;
    aReal=coeM(1,7); % set aReal

    tic
    fprintf('%.3f\n',toc);

    tFlightF=2*pi/(2*pi*rp^(3/2) /sqrt(mu));
    tFlightF=1/tFlightF;
    Tt=2*pi/( (mu^2 *(1-coeM(1,2)^2)^(3/2))/ coeM(1,1)^3);

    for sigmaC=1:length(sigmaA)

        sigma=sigmaA(sigmaC);
        for TAdistC=1:length(TAdistA)%120
            TAdist=TAdistA(TAdistC);
            TAarr=(0:TAdist:2*TAdist)';
            E=2*atan( sqrt(1-coeM(1,2))/sqrt(1+coeM(1,2))
 *tan(.5*TAarr*pi/180));
            MAarr= E-coeM(1,2)*sin(E);

            tf21=(Tt/(2*pi))*(MAarr(2)-MAarr(1));%*pi/180;
            tf32=(Tt/(2*pi))*(MAarr(3)-MAarr(2));%*pi/180;
            tf31=(Tt/(2*pi))*(MAarr(3)-MAarr(1));%*pi/180;
            %coeM(i,:)=[h e RAAN inc w TA a];
```

```
            %taF21=sqrt(coeM(1,7)^3 /mu)*(E(2)-E(1)-
coeM(1,2)*sin(E(2)-E(1)));
            for kCount=1:3
                t_hGibbs(kCount)=sqrt(coeM(1,7)^3 /mu)*(E(kCount)-
coeM(1,2)*sin(E(kCount)));
            end
            %keyboard

            TAarr=TAarr*pi/180;



            coeLp(:,6)=TAarr;
            for i=1:3
                [r, v] = sv_from_coe(coeLp(i,:),mu);
                rn = normrnd(0,sigma,[numbSamp,3]);
                %rRand(1:numbSamp,1:3)=r(1:3)+rn(1:numbSamp,1:3);
%/1000;

 rRand(1:numbSamp,1:3)=r(1:3)+0.05*ones(numbSamp,3);%rn(1:numbSamp,1:3);
%/1000;
                rMast(:,:,i)=rRand;
            end
            for k=1:numbSamp
               r1=rMast(k,:,1);
               r2=rMast(k,:,2);
               r3=rMast(k,:,3);
               v2HH=-tf32*( 1/(tf21*tf31) + mu/
(12*norm(r1)^3))*r1+(tf32-tf21)*( (1/(tf21 *tf32)) + mu/
(12*norm(r2)^3))*r2+ tf21*( 1/(tf32*tf31) + mu/(12*norm(r3)^3))*r3;

               [r2p,v2p] = gibbs_Fun(r1,r2,r3,mu);
                v2Gm(k,1)=norm(v2p);
               coe = coe_from_sv(r2p,v2p,mu);
               a(k,1)=coe(7);
               coeHH=coe_from_sv(r2,v2HH,mu);
               aHH(k,1)=coeHH(7);
               v2HM(k,1)=norm(v2HH);
            end
            [~, v2] = sv_from_coe(coeLp(2,:),mu);
           aR=ones(length(a),1)*aReal;
           v2R=ones(length(a),1)*norm(v2);

           rmsGv2(sigmaC,TAdistC)=mean((v2Gm(:)-v2R));
           rmsHGv2(sigmaC,TAdistC)=mean((v2HM(:)-v2R));
           rmsME(sigmaC,TAdistC)=MAarr(2)*180/pi;
           rmsP(sigmaC,TAdistC)=sqrt(mean((a(:)-aR).^2));
           rmsHH(sigmaC,TAdistC)=sqrt(mean((aHH(:)-aR).^2));
           %if TAdist==38.356783919597990
           if TAdist>38.2*pi/180&&TAdist<38.6*pi/180
               keyboard;
           end
           if TAdistC==127
               keyboard;
```

```
            end
        end
        fprintf('%d of %d, for %s,  ',sigmaC,length(sigmaA),OrbType);
        tr=toc;
        pctR=sigmaC/length(sigmaA);
        tTot=tr/pctR;
        tLeft=tTot-tr;
        fprintf('%.1f sec elaps, %.1f est tot, %.1f left
\n',tr,tTot,tLeft);
        tdrw=[tdrw,tLeft];
        addpoints(h,xdrw(xc),tdrw(xc));
        addpoints(hTot,xdrw(xc),tTot);
        addpoints(hCurr,xdrw(xc),tr);

        xc=xc+1;
        drawnow

    end
    rmsM=rmsP;
    rmsMH=rmsHH;
end
```





Published with MATLAB® R2018b