Appendice

In appendice vengono riportati i programmi di Matlab utilizzati per ottenere i grafici delle simulazioni numeriche mostrate nel corso di questo lavoro. Gli ultimi due programmi, invece, sono scritti in linguaggio Fortran, e sono un esempio di come sia possibile ottenere delle tabelle con le longitudini del Sole, del pianeta Mercurio, e la sua elongazione.

```
% Programma tolomeo_esempi.m
% Questo programma vuol descrivere la traiettoria di un punto P
% che si muove su un sistema di epicicli e deferenti tipico dei
% modelli tolemaici. Variando le grandezze caratteristiche del
% sistema (raggi dei vari cerchi, velocita' angolari con cui si
% muovono i centri degli epicicli sui deferenti o il punto P
% sull'epiciclo stesso...) e' possibile ottenere diverse traiettorie.

%------------
%Esempio 1
%------------

%Caso (a): le velocita' angolari del deferente e dell'epiciclo
%sono una multipla dell'altra, cioe' la traiettoria e' chiusa.

t=[0:0.01:8.5];
omega1=1;
omega2=3*omega1;
r1=1;
r2=r1*(0.4);
r=sqrt((r1.^2)+(r2.^2)-(2.*r1.*r2.*cos(pi-(omega2.*t))));
f=omega1.*t+ asin(((r2./r).*sin(pi-(omega2.*t))));
X=r.*cos(f);
Y=r.*sin(f);

figure;
plot(X,Y);
axis square;
xlabel('Coordinata X','FontSize',12);
ylabel('Coordinata Y ','FontSize',12);
grid on;

%Caso (b): Variando il valore di omega2 e' possibile ottenere
```

```
%altri tipi di traiettoria. Ad esempio, se omega2 non e' un
%multiplo di omega1 la traiettoria non e' chiusa...

omega2=3.2*omega1;
r=sqrt((r1.^2)+(r2.^2)-(2.*r1.*r2.*cos(pi-(omega2.*t))));
f=omega1.*t+ asin(((r2./r).*sin(pi-(omega2.*t))));
X=r.*cos(f);
Y=r.*sin(f);

figure;
plot(X,Y);
axis square;
xlabel('Coordinata X','FontSize',12);
ylabel('Coordinata Y ','FontSize',12);
box on;
grid on;

%--------------
%ESEMPIO 2 e 3
%--------------

%Anche le ellissi o i cerchi non centrati nell'origine si
%possono ottenere utilizzando soltanto un sistema di epicicli
%e deferenti con raggi e velocita' angolari opportuni.
%Per il moto del sole rispetto alla terra, e in particolare
%per spiegare la non equidistanza temporale tra gli equinozi,
%Tolomeo utilizzo' questo modello...(nella figura la terra si
%trova nell'origine)

omega1=1;
omega2=-omega1;
r1=1;
r2=r1.*(3./50);
r=sqrt((r1.^2)+(r2.^2)-(2.*r1.*r2.*cos(pi-(omega2.*t))));
f=omega1.*t+ asin(((r2./r).*sin(pi-(omega2.*t))));
X=r.*cos(f);
Y=r.*sin(f);

figure;
plot(X,Y);
xlim([-1.2,1.2]);
ylim([-1.2,1.2]);
axis square;
xlabel('Coordinata X','FontSize',12);
```

```
ylabel('Coordinata Y ','FontSize',12);
box on;
grid on;

%Utilizzando una opportuna velocita' angolare del pianeta che
%si trova sull'epiciclo e' possibile ottenere anche un'ellisse

omega1=1;
omega2=-2*omega1;
r1=1;
r2=r1.*(3./50);
r=sqrt((r1.^2)+(r2.^2)-(2.*r1.*r2.*cos(pi-(omega2.*t))));
f=omega1.*t+ asin(((r2./r).*sin(pi-(omega2.*t))));
X=r.*cos(f);
Y=r.*sin(f);

figure;
plot(X,Y);
box on;
xlim([-1.2,1.2]);
ylim([-1.2,1.2]);
axis square;
xlabel('Coordinata X','FontSize',12);
ylabel('Coordinata Y ','FontSize',12);
box on;
grid on;

%------------
%ESEMPIO 4
%------------

%Nel terzo esempio si mostra un sistema piu' complicato, dove
%si combinano un deferente, un epiciclo maggiore al quale si
%associa una velocita' angolare concorde a quella del deferente,
%e un epiciclo minore di velocita' angolare discorde.

omega1=1;
omega2=3.2.*omega1;
r1=1;
r2=r1.*0.4;
r=sqrt((r1.^2)+(r2.^2)-(2.*r1.*r2.*cos(pi-(omega2.*t))));
f=omega1.*t+asin(((r2./r).*sin(pi-(omega2.*t))));
r3=0.1;
omega3=-1.17;
```

```
rho=sqrt(abs((r.^2)+(r3.^2)-(2.*r.*r3.*cos(pi-(omega3.*t)-(omega2.*t-
(asin(((r2./r).*sin(pi-(omega2.*t))))))))));
fi=f+asin((r3./rho).*sin(omega2.*t-(asin((r2./r).*sin(pi-(omega2.*t))))));
[Z,T]=pol2cart(fi,rho);

figure;
plot(Z,T);
axis square;
xlabel('Coordinata X','FontSize',12);
ylabel('Coordinata Y ','FontSize',12);
box on;
grid on;
```

```
% Programma tolomeo_mercurio.m
% Questo programma vuol dare un modello per l'orbita di Mercurio,
% prima secondo un modello semplicissimo (figura 1) e poi (figura 2)
% secondo quanto proposto nell'Almagesto di Tolomeo secondo Paul Tannery.

%------------------------
% Modello semplice
%-----------------------
t=[0:0.01:400];
r2=0.387;
r1=1;
omega1=(2.*pi)./365;
omega2=(2.*pi)./(0.241.*365);
X=(r1.*cos((omega1.*t)))+(r2.*cos((omega2.*t)));
Y=(r1.*sin((omega1.*t)))+(r2.*sin((omega2.*t)));
figure;
plot(X,Y);
axis square;
grid on;

clear all;
```

```
%--------------------------
% Modello piu' complesso
%--------------------------

t=[0:0.01:2000];
r1=0.37083;
r2=1;
r3=1./24;
omega1=(2.*pi)./119;
omega2=(2.*pi)./365;
omega3=-omega2;

%Cerco ora le coordinate X,Y del pianeta (il programma 'divide per zero'
%quando ho D=C)

x3=r3.*cos(omega3.*t);
DCZ=0.5.*(omega3.*t);
DZ=r3;
ZC=r3;
DC=sqrt((DZ.^2)+(ZC.^2)-(2.*DZ.*ZC.*cos(pi-(omega3.*t))));
DG=r2;
DCG=(omega2.*t)+(DCZ);
DGC=asin((sin(DCG).*DC)./DG);
CDG=pi-(DCG+DGC);
CG=(DG.*sin(CDG))./(sin(DCG));
x2=CG.*sin(omega2.*t);
CDZ=DCZ;
ZDG=CDG-CDZ;
K=pi-(CDG+DCZ);
x1=r1.*sin(K+(omega1.*t));
X=x1+x2;

y1=r1.*cos(K+(omega1.*t));
CT=1./20;
y2=(CG.*cos(omega2.*t))+CT;
y3=r3.*sin(omega3.*t);
Y=y2+y1;
figure;
h=plot(X,Y);
set(h,'color','blue')

%Considero adesso l'orbita del Sole sempre secondo Tolomeo, in modo da
%poter comparare i due moti
```

```
r4=0.03;
r=sqrt((r2.^2)+(r4.^2)-(2.*r4.*r2.*cos(pi-(omega3.*t))));
f=(omega2.*t)- (asin(((r4./r).*sin(pi-(omega3.*t)))));
K=r.*cos(f);
W=r.*sin(f);
hold on;
xlim([-1.5,1.5]);
ylim([-1.5,1.5]);
axis square;
grid on;
%title('Traiettoria di Mercurio(blu) e del Sole(magenta)','FontSize',12);
xlabel('Coordinata X','FontSize',12);
ylabel('Coordinata Y ','FontSize',12);
p=plot(W,K);
set(p,'color','magenta')
```

```
program mercurio


      INTEGER T,N
      REAL*8 X(0:3000),Y(0:3000),TG(0:3000),A(0:3000)
      REAL*8 ZX(0:3000),ZY(0:3000),zz(0:3000),dt(0:3000)
      real*8 dcz(0:3000),dcg(0:3000),dc(0:3000),dgc(0:3000)
      real*8 cdg(0:3000),cg(0:3000),zdg(0:3000),f(0:3000)
      real*8 x1(0:3000),x2(0:3000),y1(0:3000),y2(0:3000)
      real*8 d1(0:3000),o1,o2
      n=0
      OPEN
(UNIT=2,FILE='mercury',STATUS='UNKNOWN',FORM='FORMATTED')
      DO 50 T=0,600,10



      ZX(T)=0.017214*T
      ZY(T)=0.052799*T
      zz(t)=-zx(T)
      dcz(t)=0.5*zz(t)
      dcg(t)=zx(t)+dcz(t)
      dc(t)=sqrt(0.0034722-0.0034722*cos(3.1415927-zz(t)))
      dgc(t)=asin(dsin(dcg(t))*dc(t))
      cdg(t)=3.1415927-(dcg(t)+dgc(t))
      cg(t)=dsin(cdg(t))/dsin(dcg(t))
      x2(t)=cg(t)*dsin(zx(t))
```

```
      zdg(t)=cdg(t)-dcz
      f(t)=3.1415927-(cdg(t)+dcz(t))
      x1(t)=0.37083*dsin(f(t)+zy(t))
      X(t)=x1(t)+x2(t)
      y1(t)=0.37083*cos(f(t)+zy(t))
      y2(t)=(cg(t)*cos(zx(t)))+0.05
      Y(t)=y1(t)+y2(t)
      TG(t)=Y(t)/X(t)
      IF ( X(T).LT.0.AND.Y(T).GT.0 ) THEN
        a(t)=(180+atan(tg(t))*57.2957)
      END IF
      IF (X(t).LT.0.and.y(t).lt.0) THEN
        a(t)=(atan(tg(t))*57.2957+180)
      END IF
      IF (x(t).gt.0.and.y(t).lt.0) THEN
        a(t)=(360+atan(tg(t))*57.29578)
      END IF
      IF (x(t).GT.0.and.y(t).gt.0) THEN
        a(t)=(atan(tg(t))*57.29578)
      END IF
      d1(t)=t-10
      dt(t)=(a(t)-a(d1(t)))
      o1=-0.01
      o2=0.01

      if (dc(t).gt.o1.and.dc(t).lt.o2) then
        N=N+1
      end if
      print *, t,a(t),dt(t)

      write (2,45)a(t)
45     FORMAT (5X,F8.4,5X,F8.4,5X,F8.4)
50     CONTINUE
      STOP
      END
```