

# Actividad 6

Edwin Herrera  
Departamento de Fisica  
Universidad de Sonora

30 de Noviembre de 2017

# **1 Sistema Sol-Tierra**

La traslación de la Tierra es el movimiento de este planeta alrededor del Sol, que es la estrella central del sistema solar. La Tierra describe a su alrededor como una órbita elíptica.

Si se toma como referencia la específica posición de una estrella, la Tierra realiza una vuelta completa en un año sidéreo, cuya duración es de 365 días, 5 horas, 45 minutos y 46 segundos. El año sidéreo es de poca importancia práctica. Para las actividades terrestres es más importante la medición del tiempo según las estaciones.

## **1.1 Sistema Tierra-Luna**

La Luna es el único satélite natural de la Tierra. Con un diámetro ecuatorial de 3474 km es el quinto satélite más grande del Sistema Solar, mientras que en cuanto al tamaño proporcional respecto de su planeta es el satélite más grande: un cuarto del diámetro de la Tierra y  $1/81$  de su masa.

Al desplazarse en torno del Sol, la Tierra arrastra a su satélite y la forma de la trayectoria que esta describe es una curva de tal naturaleza que dirige siempre su concavidad hacia el Sol. La velocidad con que la Luna se desplaza en su órbita alrededor de la Tierra es de 1 km/s.

## **1.2 Trabajo Sistema Sol-Tierra-Luna**

En este trabajo tuvimos que hacer un modelo Sol-Tierra-Luna en el cual la órbita de estas eran una circunferencia. Teniamos que hacer un programa en el cual la órbita de la tierra alrededor del sol y a su vez la órbita de la luna alrededor de la tierra. Teniamos que encontrar la posición en formas polares para dar el resultado en coordenadas cartesianas.

El código que utilice para programar en Fortran 90 fue el siguiente:

```
function solx(thetaL) result (x)
    double precision, intent(in) :: thetaL
    double precision :: x
    double precision, parameter :: Rsol = 1.49d8
    x = Rsol * dcos(thetaL)
end function solx

function soly(thetaL) result (y)
    double precision, intent(in) :: thetaL
    double precision :: y
    double precision, parameter :: Rsol = 1.49d8
    y = Rsol * dsin(thetaL)
end function soly

subroutine luna(Rsol, Rluna, Px, Py, thetaL, thetaS)
    double precision, intent (in) :: Rsol, thetaL, thetaS
    double precision, intent (out) :: Px, Py
    double precision :: Rluna
    Rluna = Rsol / 4.0d0
    Px = (Rsol * dcos(thetaS)) + (Rluna * dcos(thetaL))
    Py = (Rsol * dsin(thetaS)) + (Rluna * dsin(thetaL))

end subroutine luna

program begin

    implicit none
    double precision :: fi, fj, Rsol, Rluna, Px, Py, thetaL, rd, dia
    double precision :: Vluna, Vsol, solx, soly, thetaS
    double precision, parameter :: pi=3.1416d0, Tsol = 360, Tluna = 28
    integer :: j
    double precision, dimension(360) :: xtotal, ytotal
    double precision, dimension(360) :: x, y

    !Convertimos los angulos a radianes.
    rd = pi / 180d0
    !Definimos el radio del sol.
    Rsol = 1.496d8
    !Los días que pasan por 1 radian.
    dia = 365.26d0 / (360d0*rd)
    !Velocidad de la Luna.
    Vluna = 2d0 * (pi / Tluna)
    !Velocidad del Sol.
    Vsol = 2d0 * (pi / Tsol)
```

```

open (unit=1, file = 'Luna-Tierra.dat', status = 'unknown')
open (unit=2, file = 'Sol-Tierra.dat', status = 'unknown')

do j=1, 360, 1
  fj = dble(j)
  thetaS = fj * Vsol
  thetaL = fj * Vluna

  x(j)= solx(thetaS)
  y(j)= soly(thetaS)

  call luna(Rsol, Rluna, Px, Py, thetaL, thetaS)
  xtotal(j) = Px
  ytotal(j) = Py

  write (1,*) xtotal(j), ytotal(j)
  write (1,*) ''
  write (2,*) x(j), y(j)
  write (2,*) ''

end do

close (unit=1)
close (unit=2)

end program

```

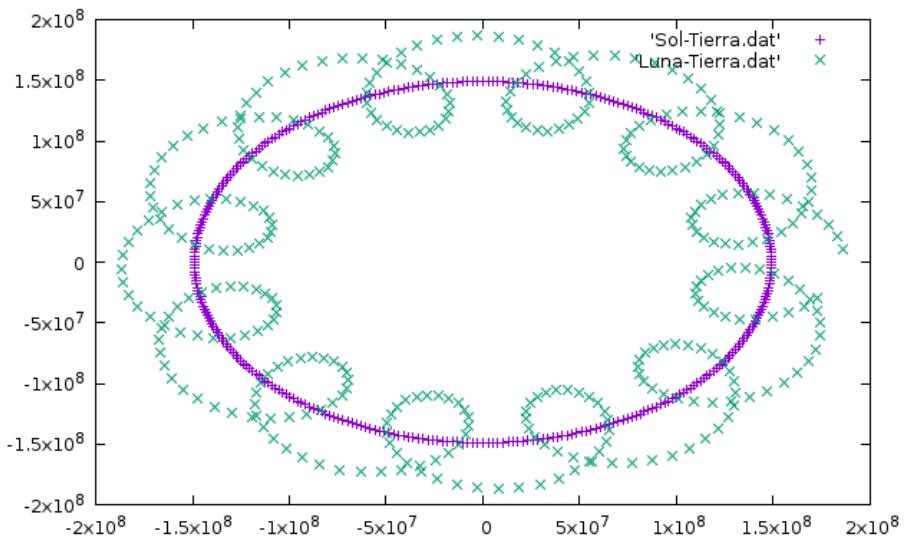


Figure 1: Grafica de posiciones de las orbitas