**Instituto Politécnico Nacional**
**Escuela Superior de Física y Matemáticas**
**Proyecto de Análisis Numérico III**

**El problema de los dos cuerpos**

**Integrantes:**
Guzmán Pérez Yahir Uriel
Fragoso Martínez Issis
García Mendoza Felipe

## Introducción

El sistema físico que consiste en las órbitas de dos masas que interactúan por la aceleración gravitacional puede ser expresado como una ecuación diferencial. Usando las leyes de movimiento de Newton y la fórmula de la fuerza gravitacional, el movimiento de estas dos masas puede ser des- crito como una función del tiempo. Decimos que tal sistema es "analíticamente soluble". Las fórmulas obtenidas demuestran que las masas siguen órbitas elípticas alrededor del centro combi- nado de masa de los dos cuerpos. Sin embargo, un sistema de tres o más masas interactuando exclusivamente por la aceleración gravitacional no es "analíticamente soluble". El así llamado **"problema de los tres cuerpos"**, tiene un espacio de estados 18-dimensional, pues, para resolver el sistema se necesita conocer las tres posiciones $(x, y, z)$ y las tres velocidades $(x', y', z')$ para cada masa, es decir, un total de 18, números. Para este caso no existe una solución explícita del sistema de ecuaciones diferenciales, pero se pueden utilizar métodos computacionales para aproximar las soluciones de las ecuaciones que resultan de las leyes de movimiento de Newton y estas aproximaciones nos dan una idea del comportamiento complicado de las trayectorias de este sistema.

Para hacer el texto más fácil de entender, en esta parte resolvemos numéricamente solamente el problema de los dos cuerpos.

Supongamos que una partícula de masa $m$ (un planeta) es atraída por un cuerpo masivo de masa $M$ (el Sol). Supondremos que la la influencia de la partícula sobre el cuerpo es despreciable, así que el cuerpo masivo está permaneciendo en reposo en el origen.

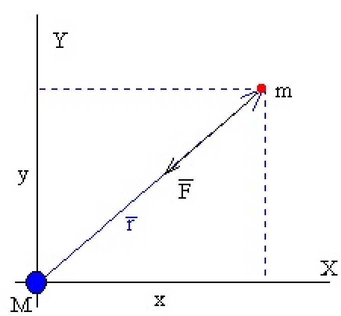


Figura 1: Diagrama de los dos cuerpos

La partícula está sometida a una fuerza atractiva $F$ cuya dirección es radial y apuntando hacia el centro del Sol. El módulo de la fuerza viene dado por la ley de la Gravitación Universal

$$F = \frac{Gm_1m_2}{r^2}u_r,$$

siendo $r = \sqrt{x^2 + y^2}$ la distancia entre la partícula y el centro de fuerzas, y $(x, y)$ su posición respecto del sistema de referencia cuyo origen está situado en el Sol. Las componentes de la fuerza son:

$$F_x = -F\cos(\theta) = -F\frac{x}{r}, \qquad F_y = -F\operatorname{sen}(\theta) = -F\frac{y}{r},$$

Aplicando la segunda ley de Newton, y expresando la aceleración como derivada segunda de la posición, tenemos un sistema de dos ecuaciones diferenciales de segundo orden:

$$m\frac{d^2x}{dt^2} = G\frac{mM}{r^2}\frac{x}{r},$$
$$m\frac{d^2y}{dt^2} = G\frac{mM}{r^2}\frac{y}{r}.$$

Dadas las condiciones iniciales (posición y velocidad inicial), el sistema de dos ecuaciones diferenciales se puede resolver aplicando un procedimiento numérico como el de Runge-Kutta.

## Escalas

Antes de resolver el sistema de ecuaciones diferenciales por procedimientos numéricos, es conveniente prepararlas para que el ordenador no maneje números excesivamente grandes o pequeños.

Establecemos un sistema de unidades en el que la longitud se mide en unidades astronómicas, la distan- cia media entre el Sol y la Tierra. $L = UA = 1{,}496 \times 10^{11}m$ y el tiempo en unidades de año $P = \text{un año} = 365{,}26 \text{ días} = 3{,}156 \times 10^7 s$.

En el nuevo sistema de unidades $x = Lx', t = P \cdot t'$, la primera ecuación diferencial se escribe:

$$\frac{d^2x'}{dt^2}\frac{L}{P^2} = -\frac{GM}{(x'^2 + y'^2)^{3/2}}x'\frac{L}{L^3},$$
$$\frac{d^2x'}{dt^2} = -\frac{GMP^2}{L^3}\frac{x'}{(x'^2 + y'^2)^{3/2}}.$$

Como $L$ es el semieje mayor de la órbita de la Tierra alrededor del Sol, $P$ es el periodo o tiempo que tarda en dar una vuelta completa y $M$ es la masa del Sol. Se sabe que

$$\frac{GMP^2}{L^3} = 4\pi^2$$

Volviendo a la notación $x$ e $y$ para la posición y $t$ para el tiempo en el nuevo sistema de unidades. El sistema de ecuaciones diferenciales se escribe:

$$\frac{d^2x}{dt^2} = -4\pi^2\frac{x}{(x^2 + y^2)^{3/2}},$$
$$\frac{d^2y}{dt^2} = -4\pi^2\frac{y}{(x^2 + y^2)^{3/2}}.$$

Se resuelve por procedimientos numéricos con las condiciones iniciales siguientes: en el instante $t = 0, x = x_0, y = 0, v_x = 0, v_y = v_{0y}$. Debido a que este es un sistema de orden dos, tenemos que hacer la sustitución:

$$u_1(t) = x(t), \quad u_2(t) = x'(t), \quad u_3(t) = y(t), \quad u_4(t) = y'(t),$$

la cual tranforma el sistema de arriba al siguiente sistema de orden uno:

$$u_1'(t) = u_2(t),$$
$$u_2'(t) = -\frac{u_1(t)}{(u_1(t)^2 + u_3(t)^2)^3/2},$$
$$u_3'(t) = u_4(t),$$
$$u_4'(t) = -\frac{u_3(t)}{(u_1(t)^2 + u_3(t)^2)^3/2}.$$

Definimos ahora la siguiente función en MATLAB que evalua el lado derecho de este sistema:

```
function f = satelite(t, u),
   f = zeros(1, 4);
   denom = (u(1) ^ 2 + u(3) ^ 2) ^ 1.5;
   f(1) = u(2);
   f(2) = -u(1) / denom;
   f(3) = u(4);
   f(4) = -u(3) / denom;
end
```

Ahora calculamos y graficamos la solución del problema de valor inicial con la siguiente secuencia de instrucciones en MATLAB. Note que graficamos el conjunto de puntos $(x(t), y(t))$ para los $t's$ generados en lugar de $(t, x(t))$ y $(t, y(t))$. Tenemos pues:

```
function [] = plotsolution2(onestepformula),
   tmin = 0;
   tmax = 10;
   x0 = [0.4, 0, 0.1, 2];
   n = 1000;
   t = linspace(tmin, tmax, n+1)';
   xapprox = onestepmethodvec(@satelite, tmin, tmax, x0, onestepformula, n);
   y = xapprox(:, 1);
   z = xapprox(:, 3);
   plot(y,z);
end
```

El parámetro `onestepformula` denota el método que aplicamos para observar la trayectoria como se muestra en la Figura .

## Un paso de Runge–Kutta

Hemos utilizado el método de Euler y algunos métodos de Runge–Kutta de órdenes 3, 4, 5. Las fórmulas correspondientes se encuentran fácilmente, por eso aquí mostramos solamente el método de orden 5:

```
function [vnext] = rk51step(f,t,v,h),
   k1 = h * f(t, v);
   k2 = h * f(t + h / 4, v + k1 / 4);
   k3 = h * f(t + h / 4, v + (k1 + k2) / 8);
   k4 = h * f(t + h / 2, v + (-k2 + 2 * k3) / 2);
   k5 = h * f(t + 3 * h / 4, v + (3 * k1 + 9 * k4) / 16);
   k6 = h * f(t + h, v + (-3 * k1 + 2 * k2 + 12 * k3 - 12 * k4 + 8 * k5) / 7;
   vnext = v + (7 * k1 + 32 * k3 + 12 * k4 + 32 * k5 + 7 * k6) / 90;
end
```

## Función que aplica el método de un paso

```
function [v] = onestepmethodvec(f, tmin, tmax, x0, onestepformula, n),
   d = length(???); v = zeros(???, d); v(1, :) = ???;
   h = (tmax - tmin) / 2; t = linspace(???, ???, n + 1)';
   for j = 1 : n,
      v(???) = onestepformula(f, ???, ???, ???);
   end
end
```

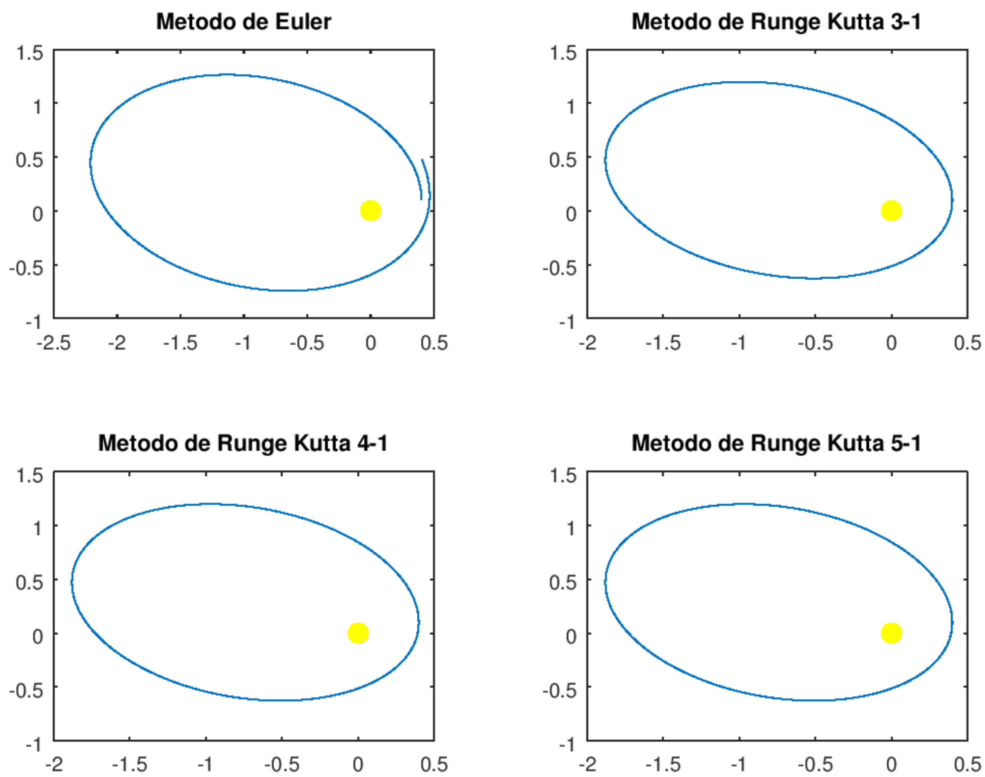


Figura 2: Trayectoria con varios metodos

### Programa que dibuja la solución del sistema

```
function [] = plotsatelite(),
   tmin = 0; tmax = 50; x0 = [0.4, 0, 0.1, 2];
   n = 10000; t = linspace(tmin, tmax, n + 1)';
   approx1 = onestepmethodvec(@satelite, tmin, tmax, x0, @eulerstep,n);
   x1 = approx1(:, 1); y1 = approx1(:, 3);
   approx3 = onestepmethodvec(@satelite, tmin, tmax, x0, @rk31step,n);
   x3 = approx3(:, 1); y3 = approx3(:, 3);
   ...
   subplot(2,2,1),plot(x1,y1,0,0,'.y','markersize',30),title('Metodo de Euler');
   subplot(2,2,4),plot(x3,y3,0,0,'.y','markersize',30),title('Metodo de Runge Kutta 3-1');
   subplot(2,2,6),plot(x4,y4,0,0,'.y','markersize',30),title('Metodo de Runge Kutta 4-1');
   subplot(2,2,8),plot(x5,y5,0,0,'.y','markersize',30),title('Metodo de Runge Kutta 5-1');
   print Solution.png -append
end
```