

Simulación de la trayectoria de un cohete mediante la resolución de un sistema de ecuaciones diferenciales usando el método de Runge-Kutta de orden 4

Rodrigo P.,Eduardo C.,Carmen R.

24 de mayo de 2020

Resumen

En el presente artículo se muestra la resolución de un problema de mecánica planetaria mediante la resolución numérica de un sistema de cuatro ecuaciones diferenciales haciendo uso de un programa en C++ y el método de Runge-Kutta de cuarto orden. Con la resolución de este sistema de ecuaciones se pretende encontrar la trayectoria que sigue una nave espacial en un viaje translunar bajo la influencia gravitatoria de la tierra y la luna. Las variables involucradas en el sistema de ecuaciones son el radio, el ángulo polar y los momentos generalizados de la nave en un instante del tiempo dado, se programó una función contenedora para el sistema de ecuaciones haciendo uso de estas variables y posteriormente se evaluó haciendo uso del método RK4, los resultados se guardaron en un archivo de texto. Los resultados arrojaron que el tiempo necesario para intersectar la trayectoria lunar es aproximadamente un día con una velocidad cercana a la velocidad de escape de la tierra y un ángulo de 65 grados, estos resultados son congruentes con la realidad por lo que la resolución del sistema de ecuaciones fue exitosa

1. Planteamiento

El problema consiste en modelar la trayectoria de un cohete bajo la influencia gravitacional de la luna y la tierra, para ello es necesario ubicar a la nave en el espacio, esto se hace con el uso de las coordenadas r y ϕ . La luna se mueve en una órbita circular de radio r_1 con frecuencia constante ω al rededor de la tierra.

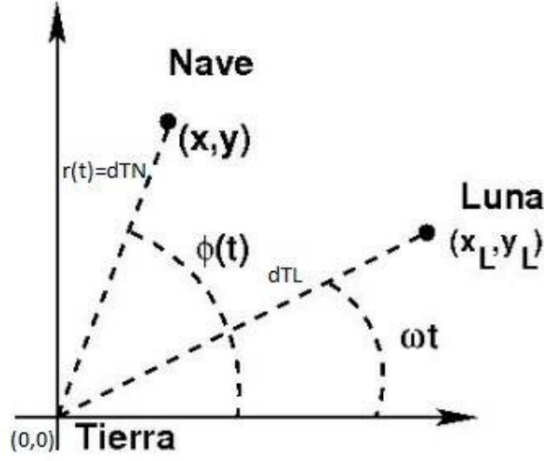


Figura 1: Ubicación de la nave y la luna en el espacio.

El movimiento de este sistema está descrito por un sistema de cuatro ecuaciones diferenciales de primer orden^[1] donde las variables son: \bar{r} , ϕ , \bar{p}_r y \bar{p}_ϕ

$$\frac{d\bar{r}}{dt} = \bar{p}_r \quad (1)$$

$$\frac{d\phi}{dt} = \frac{\bar{p}_\phi}{\bar{r}^2} \quad (2)$$

$$\frac{d\bar{p}_r}{dt} = \frac{\bar{p}_\phi^2}{\bar{r}^3} - \Delta \left[\frac{1}{\bar{r}^2} + \frac{\mu}{\bar{r}^3} [\bar{r} - \cos(\phi - \omega t)] \right] \quad (3)$$

$$\frac{d\bar{p}_\phi}{dt} = -\frac{\Delta\mu\bar{r}}{\bar{r}^3} \sin(\phi - \omega t) \quad (4)$$

Donde:

$$\Delta = \frac{GM_T}{r_l^3}$$

$$\mu = \frac{M_T}{M_l}$$

$$\bar{r}' = \sqrt{1 + \bar{r}^2 - 2\bar{r}\cos(\phi - \omega t)}$$

Las variables del sistema de ecuaciones están ajustadas para ser adimensionales o tener unidades de 1/s, de esta forma, \bar{r} está relacionada a la distancia radial de la nave al centro del sistema coordenado, ϕ es el ángulo de desfase entre la luna y la tierra y \bar{p}_r , \bar{p}_ϕ están relacionadas con los momentos generalizados de la nave en la dirección radial y angular, en términos de variables físicas se

definen de la siguiente forma:

$$\begin{aligned}\bar{r} &= r/r_l \\ \phi &= \phi \\ \bar{p}_r &= \frac{v}{r_l} \cos(\theta - \phi) \\ \bar{p}_\phi &= \frac{rv}{r_l^2} \sin(\theta - \phi)\end{aligned}$$

Donde θ es el ángulo que forma el vector momento con el vector radial de la posición de la nave

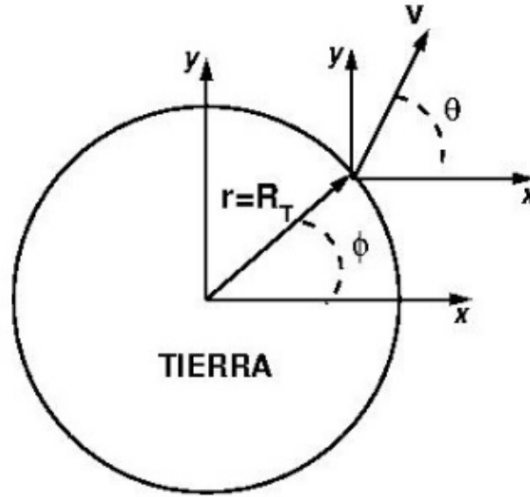


Figura 2: Ilustración de la ubicación de los ángulos ϕ y θ

2. Metodología

La resolución numérica del sistema de ecuaciones diferenciales se hizo mediante el método de Runge-Kutta de orden 4, para ello se proporcionaron 4 condiciones iniciales siendo estas las siguientes:

$$\begin{aligned}
\bar{r}_0 &= r_T/r_l = 0,01659 \\
\theta &= 65 \\
\phi &= 0 \\
\bar{v}_0 &= v_0/r_l = 2,9338 \times 10^{-5} 1/s \\
\bar{p}_r &= \bar{v}_0 \cos(\theta - \phi) \\
\bar{p}_\phi &= \frac{r_0 v_0}{r_l^2} \sin(\theta - \phi)
\end{aligned}$$

Se usó un programa para evaluar el método continuamente 43,200 veces con un paso de 2s, haciendo que la simulación evaluara un tiempo total de 24 horas de movimiento, el programa fué hecho en C++ y evaluado para este sistema de ecuaciones en particular.

Las funciones del sistema de ecuaciones fueron condensadas en una sola función contenedora que devolvía una evaluación distinta dependiendo del parámetro que se le indicase

```

1 double f(double t, double* xi, int n){
2 double fr;
3 const double om =2.6617e-6, delta =7.014e-12, mu =0.0123;
4 double r = xi[0], phi = xi[1], pr = xi[2], pp = xi[3];
5 double rpr = sqrt(1+r*r-2*r*cos(phi - om*t));
6 switch (n){
7 case 0:
8 fr = pr;
9 break;
10 case 1:
11 fr= pp/(r*r);
12 break;
13 case 2:
14 fr = (pr*pr)/(r*r*r)*delta*(1.0/(r*r)+(r-cos(phi - om*t))*mu/(rpr*
    rpr*rpr));
15 break;
16 case 3:
17 fr=-sin(phi - om*t)*delta*mu*r/(rpr*rpr*rpr);
18 break;
19 }
20 return fr;
21 }

```

Listing 1: código de la función condensadora del sistema de ecuaciones

Esta función se usó para evaluar el método RK4 para sistemas de ecuaciones dadas las condiciones iniciales y los resultados se escribieron en un archivo de texto:

```

1 void RK4sys(double ti,double* xi,double f(double t, double* xi, int
    n),double h, const int n, const int m){
2 double x[n]={};
3 double t = ti;
4 for (int i = 0;i<n;i++){
5 x[i]=xi[i];
6 }
7
8 ofstream file("datos.dat");
9 file << setprecision(8);
10 for (int j = 0;j<m;j++){
11 file << t<<" ";
12 for (int i = 0;i<n;i++){
13 file << x[i]<<" ";
14 x[i]=xvnext(t,x,f,h,n)[i];
15 }
16 t = t +h;
17 file << endl;
18 }
19 }

```

Listing 2: Aplicación del método RK4 y escritura en un archivo de texto

El método RK4 en sí se encuentra en la función "xvnext", esta toma un estado y una función y devuelve el siguiente estado al avanzar un paso.

```

1 double* xvnext(double t,double* xi,double f(double t, double* xi,
    int n),double h, const int n){
2 double k1[n]={};

```

```

3 double k2[n]={};
4 double k3[n]={};
5 double k4[n]={};
6 double* xs = new double[n];
7
8 for (int i = 0;i<n;i++){
9 k1[i]=f(t,xi,i);
10 xs[i] = xi[i] + k1[i]*h/2.0;
11 }
12 for (int i = 0;i<n;i++){
13 k2[i] = f(t+h/2.0,xs,i);
14 xs[i] = xi[i] + k2[i]*h/2.0;
15 }
16 for (int i = 0;i<n;i++){
17 k3[i] = f(t+h/2.0,xs,i);
18 xs[i] = xi[i] + k3[i]*h;
19 }
20 for (int i = 0;i<n;i++){
21 k4[i] = f(t+h,xs,i);
22 }
23 for (int i = 0;i<n;i++){
24 xs[i] = xi[i]+h*(k1[i]+2*k2[i]+2*k3[i]+k4[i])/6.0;
25 }
26 return xs;
27 }

```

Listing 3: Método RK4 para sistemas de ecuaciones diferenciales

3. Resultados

Haciendo uso del programa antes descrito se obtuvo una lista de 43,200 estados del sistema, es decir, los valores de cada variable en un dado instante de tiempo, las variables \bar{r} y ϕ dictan la trayectoria del cohete, estas se graficaron de forma polar junto a la trayectoria de la luna, se obtuvo lo siguiente:

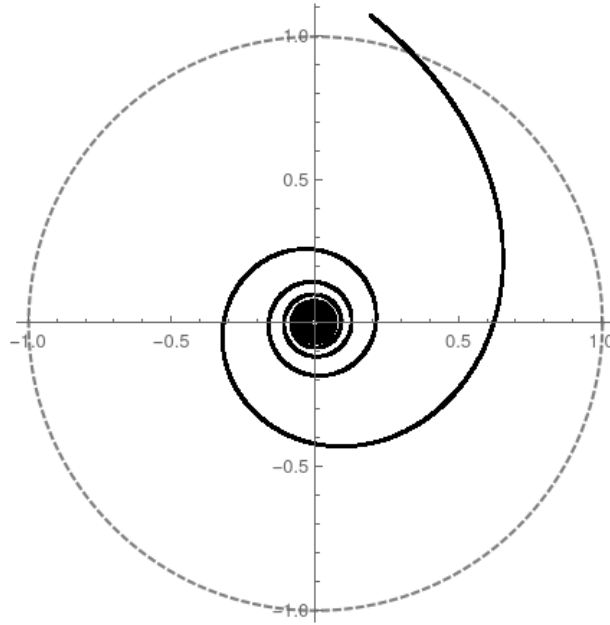


Figura 3: Trayectoria obtenida de la nave (negro) y trayectoria de la luna (gris)

Esta gráfica está normalizada respecto al radio de la luna, se puede notar que la trayectoria del cohete intersecta a la de la luna en aproximadamente 1 día y la trayectoria es espiraloide.

4. Conclusiones

Se obtuvo satisfactoriamente la solución al sistema de ecuaciones propuesto y por consiguiente la trayectoria de la nave bajo la influencia de la gravedad de la tierra y la luna. Se puede notar que el tiempo para intersectar la trayectoria lunar es aproximadamente un día con una velocidad inicial cercana a la velocidad de escape de la tierra, este resultado coincide con la realidad y es congruente con la duración de los viajes reales a la luna. La resolución de este método puede ser aplicado para encontrar trayectorias reales de vuelo translunar y ser adaptado para encontrar trayectorias de viaje transplanetario de igual forma.

Referencias

- [1] Ivón Fabian, *Modelado y simulación del movimiento de un cohete mediante un sistema de ecuaciones diferenciales*. Universidad autónoma del estado de México, Toluca, 2019.