Movimiento no relativista de una partcula cargada en un campo magntico utilizando el mtodo de Runge-Kutta

Pablo Balmaceda Rescia[1]

Universidad de Costa Rica, balmacedarescia@gmail.com[1]

Resumen Se va a discutir sobre la solucin del sistema de una partcula cargada en presencia de un campo magntico constante de manera analtica e implementando el mtodo numrico de Runge-Kutta de cuarto orden (RK4). La solucin analtica se emplea para corroborar resultados. Ademas se va a discutir sobre la implementacin del mtodo en el cdigo empleado (programado en C), as como un esquema del mtodo de Runge-Kutta.

Keywords: Runge-Kutta, RK4, partcula cargada, campo magntico, mtodo numrico

1. Introduccin del sistema

El sistema consiste en una partcula con masa m y carga q. Sea $\vec{\boldsymbol{r}}(t)$ la posicin de la partcula en el tiempo t, la velocidad y la aceleracin de la partcula son definidas como sigue:

$$\dot{\vec{\boldsymbol{r}}} = \frac{d\vec{\boldsymbol{r}}}{dt} \quad , \quad \ddot{\vec{\boldsymbol{r}}} = \frac{d\dot{\vec{\boldsymbol{r}}}}{dt} = \frac{\vec{\boldsymbol{F}}}{m}$$

con

$$\vec{\boldsymbol{F}} = q\dot{\vec{\boldsymbol{r}}} \times \vec{\boldsymbol{B}}$$

y las condiciones iniciales de posicin y velocidad, $\vec{\boldsymbol{r}}_0$ y $\dot{\vec{\boldsymbol{r}}}_0$ en t_0, en presencia del campo magntico constante $\vec{\boldsymbol{B}} = B_z\hat{\boldsymbol{k}}$. Se tiene entonces un sistema de ecuaciones diferenciales de segundo orden con valores iniciales. Se tiene entonces que $\dfrac{d\dot{\vec{\boldsymbol{r}}}}{dt}$ es igual a

$$\begin{pmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{pmatrix} = \frac{qB_z}{m}\begin{vmatrix} \hat{\boldsymbol{i}} & \hat{\boldsymbol{j}} & \hat{\boldsymbol{k}} \\ \dot{x} & \dot{y} & \dot{z} \\ 0 & 0 & \hat{\boldsymbol{k}} \end{vmatrix} = \frac{qB_z}{m}\left(\dot{y}\hat{\boldsymbol{i}} - \dot{x}\hat{\boldsymbol{j}}\right) = B_0\left(\dot{y}\hat{\boldsymbol{i}} - \dot{x}\hat{\boldsymbol{j}}\right) \tag{1}$$

Con lo cual se obtiene el sistema de tres ecuaciones diferenciales de segundo orden

$$\ddot{x} = B_0\dot{y} \tag{2}$$

$$\ddot{y} = -B_0\dot{x} \tag{3}$$

$$\ddot{z} = 0 \tag{4}$$

con

$$B_0 = \frac{qB_z}{m}$$

2. Solucin analtica del sistema

Considere la variable compleja dinmica $u(t) = x(t) + iy(t)$, esto implica que $\dot{u} = \dot{x} + i\dot{y}$ y $\ddot{u} = \ddot{x} + i\ddot{y}$, de la suma de (2) mas i (3) se obtiene que

$$\ddot{u} = B_0(\dot{y} - i\dot{x}) = -iB_0\dot{u} \tag{5}$$

lo cual es equivalente a

$$\frac{d\dot{u}}{\dot{u}} = -iB_0 dt \tag{6}$$

integrando (6) se obtiene

$$\int_{\dot{u}_0}^{\dot{u}} \frac{d\dot{u}}{\dot{u}} = ln\left(\frac{\dot{u}}{\dot{u}_0}\right) = -iB_0 \int_{t_0}^{t} dt = -iB_0\left(t - t_0\right) \tag{7}$$

que es equivalente a

$$\dot{u} = \dot{u_0}e^{-iB_0 t} \tag{8}$$

con $t_0 = 0$, luego integrando (8) se obtiene la solucin compleja del sistema

$$u - u_0 = \frac{i\dot{u}_0 e^{-iB_0 t}}{B_0} \tag{9}$$

Realizando la multiplicacin de $\dot{u}_0$ con $e^{-iB_0 t}$ y separando la parte real e imaginaria de todos los trminos se obtienen las ecuaciones

$$x(t) = \frac{\dot{x}_0}{B_0} sen(B_0 t) + \frac{\dot{y}_0(1 - cos(B_0 t))}{B_0} + x_0 \tag{10}$$

$$y(t) = \frac{\dot{y}_0}{B_0} sen(B_0 t) + \frac{\dot{x}_0(1 - cos(B_0 t))}{B_0} + y_0 \tag{11}$$

ademas es sencillo ver que

$$z(t) = \dot{z}_0 \frac{t^2}{2} + z_0 \tag{12}$$

cumple la ecuacin (4).

3. Mtodo de Runge Kutta

En esta seccin se discutir el mtodo numrico de Runge Rutta empleado para solucionar el sistema de ecuaciones diferenciales. El mtodo de Runge Kutta que se usa es el de orden cuatro, en trminos de diferencias esta dado por:

$$\begin{aligned} w_0 &= \alpha \\ w_1 &= hf(t_i, w_i) \\ k_2 &= hf(t_i + \frac{h}{2}, w_i + \frac{1}{2}k_1) \\ k_3 &= hf(t_i + \frac{h}{2}, w_i + \frac{1}{2}k_2) \\ k_4 &= hf(t_i + \frac{h}{2}, w_i + \frac{1}{2}k_2) \\ w_{i+1} &= w_i + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) \end{aligned} \tag{13}$$

para cada $i = 0, 1, ..., N-1$, este mtodo tiene un error de truncamiento de $O(h^4)$, el uso de la terminologa k_1, k_2, k_3, k_4 se introduce con el fin de eliminar la necesidad de encajar sucesivamente la evaluacin de la segunda variable de la funcin $f(t, y)$ en $y' = f(t, y)$. Con h el paso por cada iteracin, definido por el intervalo $[a, b]$ y el numero de subdivicin del intervalo N. Ademas se debe indicar la condicin inicial del sistema α.
Pero el problema es que nuestro sistema esta compuesto de mas ecuaciones, por lo que vamos a hablar sobre sistemas de orden m de ecuaciones de primer orden con condicin inicial, expresados en la forma

$$\begin{aligned} \frac{du_1}{dt} &= f_1(t, u_1, u_2, \ldots, u_m) \\ \frac{du_2}{dt} &= f_2(t, u_1, u_2, \ldots, u_m) \\ &\vdots \\ \frac{du_m}{dt} &= f_m(t, u_1, u_2, \ldots, u_m) \end{aligned} \tag{14}$$

para $a \leq t \leq b$ con condiciones iniciales

$$u_1(a) = \alpha_1 \tag{15}$$
$$u_2(a) = \alpha_2 \tag{16}$$
$$\vdots \tag{17}$$
$$u_m(a) = \alpha_m \tag{18}$$

en este caso se emplea la notacin w_{ij} para denotar la aproximacin $u_i(t_j)$ para $i = 1, 2, \ldots, m$ y $j = 0, 1, \ldots, N$, es decir w_{ij} aproxima la i-sima solucin $u_i(t)$ de (13) en el j-simo punto $t_j = a + jh$. Lo anterior indica que existen $k_{l,i}$ para cada i-sima solucin, con $l = 1, 2, 3, 4$. Por lo que debe calcularse por ejemplo todos los $k_{1,i}$ para poder calcular $k_{2,1}$ lo anterior pues:

$$k_{2,i} = hf_i(t_j + \frac{h}{2}, w_{1,j} + \frac{1}{2}k_{1,1}, \ldots, w_{m,j} + \frac{1}{2}k_{1,m})$$

4. Implementacin del cdigo

El cdigo implementado fue programado en lenguaje C, utilizando programacin modular. Se creo un modulo que resolviera ecuaciones diferenciales vectoriales de segundo orden, es decir el modulo resuelve ecuaciones de la forma:

$$\frac{d^2\vec{\boldsymbol{r}}}{dt^2} = \vec{\boldsymbol{G}}\left(t, \vec{\boldsymbol{r}}, \frac{d\vec{\boldsymbol{r}}}{dt}\right) \tag{19}$$

Pero para poder aplicar el mtodo RK4 las ecuaciones deben ser de primer orden, por lo que se emplea el cambio de variable

$$\frac{d\vec{\boldsymbol{r}}}{dt} = \vec{\boldsymbol{v}} \quad \frac{d\vec{\boldsymbol{r}}}{dt} = F(t, r, \vec{\boldsymbol{v}}) \tag{20}$$

por lo que se obtienen seis ecuaciones diferenciales de primer orden.
Lo primero que se hace en el cdigo es crear una estructura que almacene nmeros reales

```
struct vector {
    double vx;
    double vy;
    double vz;
};
```

Luego una estructura que almacene punteros de funciones que reciben como parmetros double, Vector posicin y velocidad

```
struct vector_funciones {
    double(*tipofunciondif_x)(double t, Vector posicion, Vector velocidad);
    double(*tipofunciondif_y)(double t, Vector posicion, Vector velocidad);
    double(*tipofunciondif_z)(double t, Vector posicion, Vector velocidad);
};
```

Se crean ademas unos arreglos que almacenan las "kz otro que guarde las funciones diferenciales

```
double F[6], SumaK[6];
```

y unas funciones que calculan los parmetros k_1, k_2, k_3, k_4, y sacan el promedio descrito en (13)

```
void Inicializar_Kaas(double t, Vector posicion,
Vector velocidad, VectorF FunDif_1, VectorF FunDif_2, double paso);
void Suma_Kaas();
```

por ultimo este valor se pasa a la funcin que soluciona y guarda cada punto en un archivo

```
double solucion(double liminf ,double limsup,Vector posicion_inicial, Vector vel
VectorF FunDif_1, VectorF FunDif_2,
VectorF posicion_exacta, VectorF velocidad_exacta);
```

dentro de esta funcin se crea una variable tipo Vector para asignar la posicin y otra para la velocidad, dado que la velocidad y posicin iniciales se pasan como argumentos de la funcin, entonces se inicializan las variables de la siguiente manera

```
posicion.vx=posicion_inicial.vx;
posicion.vy=posicion_inicial.vy;
posicion.vz=posicion_inicial.vz;

velocidad.vx=velocidad_inicial.vx;
velocidad.vy=velocidad_inicial.vy;
velocidad.vz=velocidad_inicial.vz;
```

luego dentro de un ciclo de la forma

```
for (int i=1; i<N; i++) {
```

dentro del ciclo se llama a las funciones "$Inicializar_Kaas$" y "$Suma_Kaas$"

```
Inicializar_Kaas( t, posicion, velocidad, FunDif_1, FunDif_2, paso);
Suma_Kaas();

inter_posicion.vx=SumaK[0];
inter_posicion.vy=SumaK[1];
inter_posicion.vz=SumaK[2];

inter_velocidad.vx=SumaK[3];
inter_velocidad.vy=SumaK[4];
inter_velocidad.vz=SumaK[5];

// x vx
posicion.vx+=inter_posicion.vx;
velocidad.vx+=inter_velocidad.vx;
// y vy
posicion.vy+=inter_posicion.vy;
velocidad.vy+=inter_velocidad.vy;
//z vz
posicion.vz+=inter_posicion.vz;
velocidad.vz+=inter_velocidad.vz;
```

con lo cual lo que resta es guardar en un archivo los datos.

Por otra parte, en el archivo principal se crean las funciones que representan las ecuaciones diferenciales del sistema (1) y se inicializa una estructura tipo $vector_funciones$ y se crean variables $Vector$ para la posicin y la velocidad inicial, se inicializan con los valores del cuadro (1) y se llama a la funcin del modulo de resolucin de ecuaciones diferenciales vectoriales, como se muestra en el ejemplo

```
posicion_inicial=Inicializar_posicion(posicion_inicial, 0.0, 1.0, 0.0);
velocidad_inicial=Inicializar_velocidad(velocidad_inicial, 0.0, 1.0, 1.0);
solucion(T0, Tf, posicion_inicial, velocidad_inicial, FunDif_1,
FunDif_2, posicion_exacta, velocidad_exacta);
```

B_0	$\vec{\boldsymbol{r}}_0$	$\dot{\vec{\boldsymbol{r}}}_0$
0.1	$(1,1,0)$	$(0,1,0)$
0.5	–	$(1,0,0)$
1	–	$(0,-1,0)$
2	–	$(0,0,1)$
–	–	$(0,-1,1)$
–	–	$(0,1,1)$

Cuadro 1. Constante y Valores iniciales implementados.

5. resultado analticos y numricos

Al graficar las ecuaciones encontradas en la seccin (2.) (10,11,12) junto con las soluciones numricas empleando (13) en el sistema de seis ecuaciones diferenciales de primer orden obtenidas en la seccin anterior se obtiene las grficas (1,2,3).

En la figura (1) se observa como el radio del helicoide se ve perturbado cuando se modifica la constante B_0, para valores pequeos de la constante el radio del helicoide aumenta, mientras que para valores grandes es lo contrario. Esto tiene sentido pues si la partcula se tratase de un electrn se espera que el radio del helicoide sea menor que el de un protn pues $B_0 = \frac{qB_z}{m}$ y ciertamente la masa del protn es mayor que la del electrn.

Por otro lado, si se mantiene la constante igual y se cambian solamente las velocidades iniciales dadas en el cuadro (1) se obtiene la figura (2). En la cual se observa que para valores iniciales de la velocidad en los que $v_{z0} = 0$ el movimiento esta confinado en un plano, esto es sencillo de entender pues de la ecuacin (12) se observa que $z(t)$ es constante. Para valores en los que $v_{x0} = 0 \quad v_{y0} = 0$ es simple ver que la trayectoria de la partcula va a ser una linea en la direccin del eje z pues $x(t) = x_0$ y $y(t) = y_0$ en (10,11) y se comprueba numricamente que la trayectoria es una linea.

Por otra parte, para los casos en los que $\dot{\vec{\boldsymbol{r}}}_0$ es igual a $(0,-1,1)$ y $(0,1,1)$ la nica diferencia es que $v_{y0} = \pm 1$, para $v_{y0} = -1$ la partcula se mueve en direccin a las

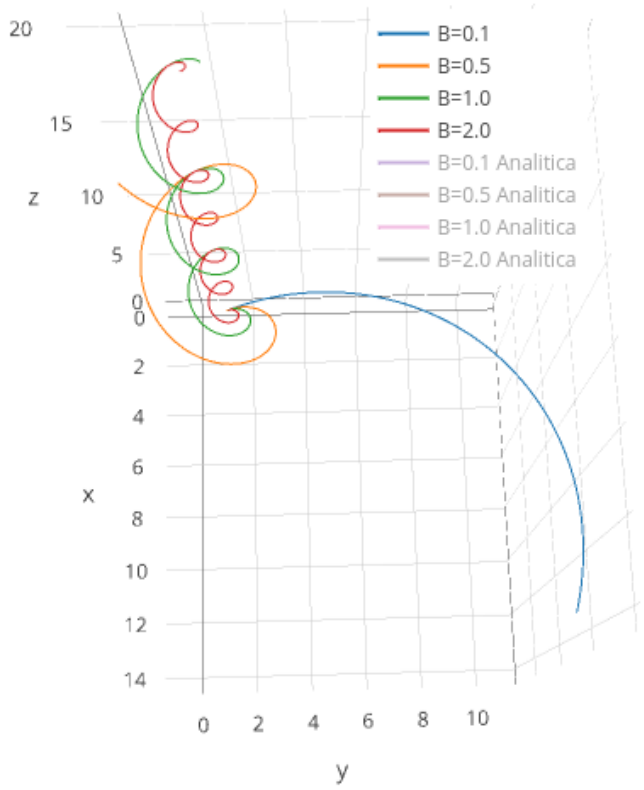

Figura 1. Movimiento de la partcula para diferentes valores de la constante B_0 dados por el cuadro (1), con la posicin y velocidad inicial $(1, 1, 0)$ y $(0, 1, 1)$ respectivamente.

manecillas del reloj y para $v_{y0} = 1$ la partcula sigue la direccin contraria, esto por la simetra del sistema.

6. conclusin

Como se comprueba en las grficas (1)(2) y las ecuaciones (10)(11)(12), los resultados numricos e analticos son en un orden de 7 dgitos iguales, con lo cual se comprueba que los movimientos posibles del sistema son helicoides, crculos o lineas para los diferentes valores iniciales.

Referencias

1. Richard L. Burden, J. Douglas Faires: Anlisis Numrico, 3 ed. , G.E. Iberoamrica, (1985).
2. Brian W. Kernighan, Dennis M. Ritchie: El Lenguaje de Programacion C, 2 ed. , Pretice-Hall, (2005).

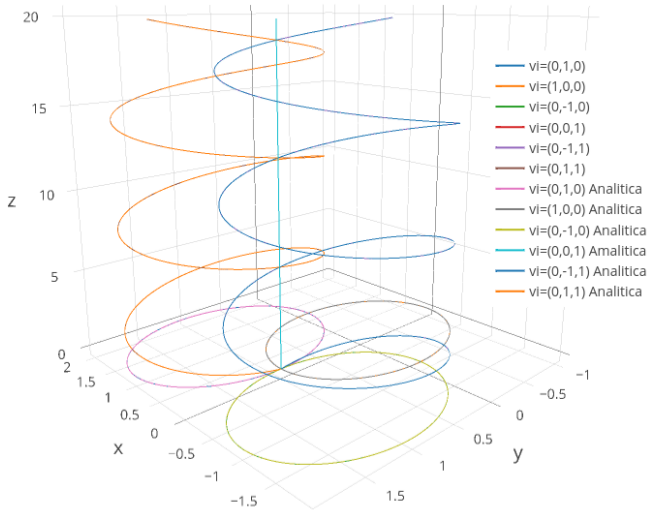

Figura 2. Movimiento de la partcula para $B_0 = 1$ y diferentes velocidades iniciales dados por el cuadro (1), con la posicin inicial $(1, 1, 0)$.

3. D. Halliday, R. Resnick: Fisica parte II, 1 ed. , Continental Editorial, (2009).

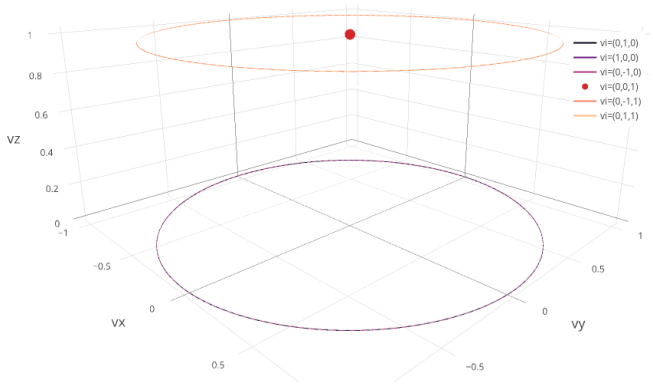

Figura 3. Espacio de velocidades, en el cual los puntos representan movimientos rectilneos y los crculos representan movimientos circulares o helicoidales.