

Tarea 5 - Problema 1 - Programación avanzada

Nicolás Aylwin

1. Desarrollo Teórico

Primero determinemos el ángulo respectivo de cada carga. Sea este θ_i para la carga q_i . Si inicialmente se tiene una distribución estable, las cargas deben estar equidistantes sobre el anillo, por lo que los ángulos entre ellas deben ser idénticos. Siendo así para $N + 1$ cargas:

$$\theta_i = \frac{2\pi}{N+1}i \quad (1)$$

Con i de 0 hasta N . Para el problema asumiremos que la carga q_0 es retirada, la que se encuentra claramente en $\theta_0 = 0$.

Ahora analicemos la fuerza que siente q_i . Ignorando fuerzas por efecto de campos magnéticos, tenemos que por fuerza de Coulomb, la fuerza que siente q_i dado q_j es:

$$\vec{F} = \frac{Kq_iq_j}{r_{ij}^3}\vec{r}_{ij}$$

Haciendo un poco de algebra:

$$\begin{aligned} \vec{r}_{ij} &= \vec{r}_i - \vec{r}_j = ((\vec{r}_i - \vec{r}_j) \cdot \hat{r}_i)\hat{r}_i + ((\vec{r}_i - \vec{r}_j) \cdot \hat{\theta}_i)\hat{\theta}_i \\ \vec{r}_{ij} &= r((\cos\theta_i - \cos\theta_j)\cos\theta_i + (\sin\theta_i - \sin\theta_j)\sin\theta_i)\hat{r}_i \\ &\quad + r((\cos\theta_i - \cos\theta_j)(-\sin\theta_i) + (\sin\theta_i - \sin\theta_j)\cos\theta_i)\hat{\theta}_i \\ \vec{r}_{ij} &= r(1 - (\cos\theta_i\cos\theta_j + \sin\theta_i\sin\theta_j))\hat{r}_i + r(\sin\theta_i\cos\theta_j - \sin\theta_j\cos\theta_i)\hat{\theta}_i \\ \vec{r}_{ij} &= r(1 - \cos(\theta_i - \theta_j))\hat{r}_i + r(\sin(\theta_i - \theta_j))\hat{\theta}_i \end{aligned}$$

Por otro lado, el módulo al cubo:

$$\begin{aligned} r_{ij}^3 &= (r^2(1 - \cos(\theta_i - \theta_j))^2 + r^2(\sin(\theta_i - \theta_j))^2)^{3/2} \\ r_{ij}^3 &= r^3(2(1 - \cos(\theta_i - \theta_j)))^{3/2} \\ r_{ij}^3 &= 8r^3 \left| \sin^3 \left(\frac{\theta_i - \theta_j}{2} \right) \right| \end{aligned}$$

Esto último dado que estamos calculando un módulo y necesitamos que sea siempre positivo. Con esto la fuerza sobre la carga q_i dadas las otras cargas, contabilizando a todas excepto a ella misma (por no autointeracción) será:

$$\vec{F}_{ij} = \sum_{j=1, j \neq i}^N \frac{Kq_iq_j}{8r^2 \left| \sin^3 \left(\frac{\theta_i - \theta_j}{2} \right) \right|} ((1 - \cos(\theta_i - \theta_j))\hat{r}_i + (\sin(\theta_i - \theta_j))\hat{\theta}_i)$$

Dado que la carga está restringida al anillo, la fuerza en \hat{r}_i es completamente contrarrestada por la que ejerce este mismo (acción y reacción). Siendo así, solo nos interesa la componente en $\hat{\theta}_i$, por lo que la fuerza total en la carga q_i es:

$$F_{ij}\hat{\theta}_i = \sum_{j=1, j \neq i}^N \frac{Kq_iq_j}{8r^2 \left| \sin^3 \left(\frac{\theta_i - \theta_j}{2} \right) \right|} ((\sin(\theta_i - \theta_j))\hat{\theta}_i)$$

$$F_{ij}\hat{\theta}_i = \frac{K}{8r^2} \sum_{j=1, j \neq i}^N \frac{q_iq_j \sin(\theta_i - \theta_j)}{\left| \sin^3 \left(\frac{\theta_i - \theta_j}{2} \right) \right|} \hat{\theta}_i$$

Considerando que las cargas son adimensionales, y tanto estas como el radio valen uno tenemos finalmente:

$$F_{ij}\hat{\theta}_i = \sum_{j=1, j \neq i}^N \frac{\sin(\theta_i - \theta_j)}{8 \left| \sin^3 \left(\frac{\theta_i - \theta_j}{2} \right) \right|} \hat{\theta}_i$$

Dado que las cargas están restringidas, las masas, radios y cargas son uno (y adimensionales) y todo movimiento está restringido a $\hat{\theta}$, la ecuación diferencial que describe la evolución de una carga cualquiera será:

$$f(t) = \theta_i(t)$$

$$v(t) = f'(t) = \dot{\theta}_i(t)$$

$$a(t) = f''(t) = \ddot{\theta}_i(t) = \sum_{j=1, j \neq i}^N \frac{\sin(\theta_i - \theta_j)}{8 \left| \sin^3 \left(\frac{\theta_i - \theta_j}{2} \right) \right|} \quad (2)$$

Por otro lado, la fuerza total que sentirá el anillo es la fuerza de las cargas intentando escapar de este. O sea la componente radial de la fuerza de coulomb de cada una de las cargas mas la fuerza radial que ejerce cada carga sobre el anillo por el hecho de rotar (la reacción de la fuerza centrípeta ejercida por el anillo para mantener en este a las cargas). Tomando en cuenta la fuerza de coulomb radial sobre la carga q_i y el aporte de la reacción de la fuerza centrípeta (con lo ya mencionado sobre las cargas, el radio y la masa):

$$\vec{F} = \dot{\theta}_i^2 \hat{r}_i + \sum_{j=1, j \neq i}^N \frac{(1 - \cos(\theta_i - \theta_j))\hat{r}_i}{8 \left| \sin^3 \left(\frac{\theta_i - \theta_j}{2} \right) \right|}$$

La fuerza total será la suma sobre todas las cargas q_i . Además, como \hat{r}_i depende de θ_i , resulta mejor expresarlo en cartesianas, para realizar la suma vectorial correspondiente. Considerando todo esto se tiene finalmente:

$$\vec{F}_T = \sum_{i=1}^N \dot{\theta}_i^2 \hat{r}_i + \sum_{i=1}^N \sum_{j=1, j \neq i}^N \frac{(1 - \cos(\theta_i - \theta_j))\hat{r}_i}{8 \left| \sin^3 \left(\frac{\theta_i - \theta_j}{2} \right) \right|}$$

$$\vec{F}_T = \sum_{i=1}^N \dot{\theta}_i^2 (\cos(\theta_i)\hat{i} + \sin(\theta_i)\hat{j}) + \sum_{i=1}^N \sum_{j=1, j \neq i}^N \frac{(1 - \cos(\theta_i - \theta_j))}{8 \left| \sin^3 \left(\frac{\theta_i - \theta_j}{2} \right) \right|} (\cos(\theta_i)\hat{i} + \sin(\theta_i)\hat{j}) \quad (3)$$

Lo único que falta determinar es la energía total del sistema. Considerando la energía cinética y potencial de cada partícula, y los valores para masas, cargas y radio:

$$\begin{aligned}
E_T &= \sum_{i=1}^N \frac{mv^2}{2} + \frac{1}{2} \sum_{i=1}^N \sum_{j=1, j \neq i}^N \frac{Kq_i q_j}{r_{ij}} \\
E_T &= \sum_{i=1}^N \frac{\dot{\theta}_i^2}{2} + \frac{1}{2} \sum_{i=1}^N \sum_{j=1, j \neq i}^N \frac{1}{r_{ij}}
\end{aligned} \tag{4}$$

El 1/2 en la energía potencial es por considerar cada distancia dos veces en la sumatoria.

2. Programa

El programa es desarrollado en el archivo “problema1.cc” teniendo dependencia de una clase de integradores (odeint.h y .cc) y de una de matrices (matriz.h). Este comienza con la declaración de 4 funciones.

La primera, “coulomb” es la encargada de devolver dentro de los distintos integradores la aceleración enunciada en la Ec.(2), además escribe la fuerza total sobre el anillo (Ec.(3)) en una matriz auxiliar.

La segunda “pos”, genera una matriz de una columna y n filas (con n la cantidad de partículas que quedan luego de sacar una) con las posiciones en θ según lo expresado en la Ec.(1).

La tercera “energia” se encarga de calcular a cada paso la energía total del sistema (según la Ec.(4)) y entregar este resultado en una matriz de una fila y m columnas (con m cantidad de pasos). Recibe la matriz de matrices generada por los integradores.

Por último, la cuarta función “pasos_20” se encarga de crear los archivos solicitados, recibe el nombre del archivo, la matriz de matrices resultante de cada integrador y el paso (para imprimirlo).

El programa comienza recibiendo los tres argumentos por main exigidos (cantidad de partículas, tiempo final y paso). Se asume por defecto que la cantidad de partículas recibidas es ya habiendo quitado la carga. Luego se invoca a la función “pos”, para crear la matriz de posiciones iniciales “posi”. Se crea una copia de esta pero con ceros para utilizar de velocidades iniciales. También se agrega una tercera matriz de n filas y 2 columnas, donde se guardará a cada paso la fuerza sobre el anillo en \hat{i} y \hat{j} . Estas tres matrices junto al tiempo inicial (como cero), y el paso se usan para crear el objeto tipo odeint llamado “anillo”.

Luego se usa cada uno de los integradores, definidos como métodos de este objeto. Estas reciben la función “coulomb” y el tiempo final (ingresado en los argumentos de main). Cada uno de estos entrega una matriz de matrices, donde las columnas son cada paso, y las filas contienen toda la información relevante del sistema. La primera fila contiene el tiempo, la segunda las posiciones de todas las cargas, la tercera todas las velocidades, la cuarta todas las aceleraciones y la quinta la fuerza sobre el anillo (primera columna \hat{i} y segunda \hat{j}).

Finalmente se exportan los datos solicitados a archivos separados con el metodo.dat como nombre. Esto a través de la función “pasos_20”, la cual se encarga en algún punto de llamar a la función “energia” para almacenar esta también.

3. Resultados y compilación

Para realizar todo lo anterior solo basta utilizar make en la terminal. Los archivos están estructurados de tal manera que cada fila es un tiempo, en orden de izquierda a derecha cada fila contiene: tiempo, paso, posiciones, velocidades, energía total, fuerza sobre el anillo en \hat{i} y en \hat{j} . Todos estos separados por una tabulación.

Por otro lado, se tiene una carpeta extra llamada “separados”, donde se encuentran los ejecutables del mismo programa pero solo usando un integrador y evitando la creación del archivo. Así poder comprobar con estos la latencia del integrador utilizado (a través del comando time).

En particular, para un sistema de 20 partículas a un tiempo final de 100 con un paso de 0.1 se tiene para cada método:

Integrador	Tiempo real [s]
Euler-Cromer	0.774
Pefrl	3.080
Rk4	2.507
Velocity-Velver	1.893