Control 2. Computación I

Jose M. Soler
Universidad Autónoma de Madrid

12 de febrero de 2018

1. El cuadro 1 contiene las coordenadas $\mathbf{r}_i$ de los centros de $n = 5$ esferas metálicas, así como sus radios R_i y potenciales eléctricos V_i. Suponiendo que la carga q_i de cada esfera se distribuye uniformemente sobre su superficie, su potencial puede aproximarse por

$$V_i = k\frac{q_i}{R_i} + \sum_{j\neq i} k\frac{q_j}{r_{ij}} \tag{1}$$

siendo k la constante de Coulomb y r_{ij} la distancia entre los centros de las esferas i y j. Escribir un programa *chargedSpheres.m* que

- Calcule una matriz U tal que las ecuaciones anteriores puedan escribirse como $V = Uq$, siendo V y q dos vectores columna con los potenciales y las cargas. (2 puntos)
- Resuelva dichas ecuaciones y escriba las cargas de las n esferas. (1 punto)
- Calcule y escriba el vector campo eléctrico en el centro de las esferas, así como la fuerza sobre cada esfera. (1 punto)
- Dibuje en tres dimensiones las esferas, coloreadas según su potencial (usar las funciones *sphere* y *surf* de Matlab) y los campos eléctricos (con la función *quiver3*). (1 punto)

Esfera	x	y	z	R	V
1	0.5	1.2	-0.1	12	7
2	-0.3	0.8	-0.4	15	-12
3	0.4	-0.4	0.1	18	15
4	-0.8	0.1	1.1	20	-15
5	-0.9	-0.7	0.5	10	10

Cuadro 1: Coordenadas x, y, z de los centros (en m), radios R (en cm) y potenciales (en V) de $n = 5$ esferas metálicas.

2. Un collar de $n = 10$ partículas (de masa $m = 0{,}1$ kg, unidas por muelles de constante $k = 100$ N/m y longitud en reposo $d_0 = 5$ cm), cuelga verticalmente del extremo de un brazo horizontal de longitud $R = 15$ cm, orientado inicialmente según el eje x. En el instante $t = 0$, el brazo comienza a girar con velocidad angular $\omega = 2\pi$ rad/s alrededor de un eje vertical que pasa por su otro extremo. La fuerza del muelle j será $k(d_j - d_0)\mathbf{u}_j$, siendo d_j su longitud y $\mathbf{u}_j = (\mathbf{r}_{j+1} - \mathbf{r}_j)/d_j$ el vector unitario de su dirección. $\mathbf{r}_{n+1} = (R\cos(\omega t), R\sin(\omega t), 0)$ es la punta del brazo.

   - Escribir una función con la siguiente interfaz (2 puntos)

   ```
   function f = rotatingStringForces(r,t,R,w,d0,k,m)
   % Forces on a string of particles connected by springs
   % and pulled by a rotating arm
   % Input:
   %   r(3,n) : particle positions (m)
   %   t      : time (s)
   %   R      : pulling arm length (m)
   %   w      : pulling arm angular velocity (rad/s)
   %   d0     : unstretched spring length (m)
   %   k      : springs constant (N/m)
   %   m      : particles mass (kg)
   % Output:
   %   f(3,n) : force on each particle (N)
   ```

   - Escribir un programa *rotatingString.m* que inicialice las posiciones de las masas y llame a la función *rungeKutta* de clase, para hallar sus posiciones durante 10 s, a intervalos de 10^{-2} s. Supóngase que inicialmente las masas cuelgan verticalmente a las distancias de equilibrio, $d_j = d_0 + M_j g/k$, siendo $M_j = mj$ la masa total colgada debajo del muelle j. (2 puntos)
   - Representar la posición de las masas como un video en tres dimensiones, usando un intervalo de 0,2 s entre imágenes. (1 punto)

Notas:

- Los programas deben estar documentados en inglés y no deben escribir nada que no se pida.
- Las figuras deben incluir etiquetas en cada eje.
- Enviar los ficheros *chargedSpheres.m, rotatingStringForces.m* y *rotatingString.m* a jose.soler@uam.es antes de terminar el examen.
- Para cada función y programa, se valorará:
  1. Que tenga la interfaz que se pide, y haga lo que se pide sin errores.
  2. Que esté bien documentado.
  3. Que esté programado de forma sencilla y fácil de entender.