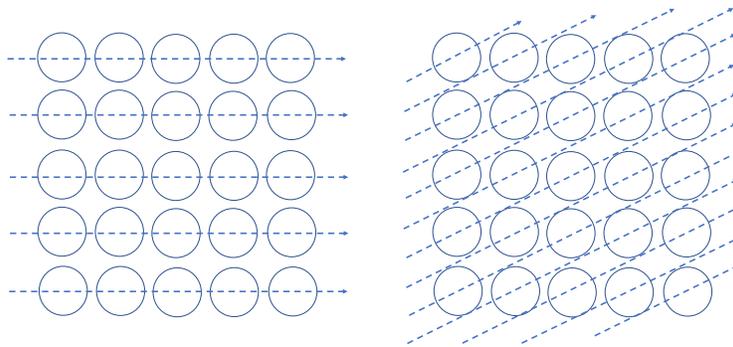


# Control 2. Computación I

Jose M. Soler  
Universidad Autónoma de Madrid

10 de febrero de 2020

## 1. Tomografía



La figura representa esquemáticamente el funcionamiento de la tomografía axial computarizada (TAC). Los círculos representan “píxeles” de densidad y las flechas, rayos X que los atraviesan con cierto ángulo que se va cambiando. Se mide la absorción de cada rayo para cada ángulo. Dicha absorción es proporcional al camino recorrido y a la densidad de los “píxeles” que atraviesa el rayo:

$$b_r = \sum_p A_{rp} \rho_p \quad (1)$$

donde  $r$  numera los rayos,  $p$  numera los píxeles, y  $A_{rp}$  es igual a cero si el rayo  $r$  no atraviesa el píxel  $p$ , e igual a la distancia  $d_r$  entre los píxeles  $p$  que sí atraviesa. El objetivo es reconstruir la densidad  $\rho_p$  de cada píxel a partir de la absorción  $b_r$  de cada rayo. Para ello, se escribirá un programa *tomography.m* que:

- Lea (puede usarse *load*) el fichero *tomography.dat*, previamente descargado de la página web de la asignatura. Dicho fichero contiene la absorción  $b_r$  de los rayos que atraviesan una rodaja de  $n_x \times n_y$  píxeles (con  $n_x = n_y = 9$ ) para  $n_a = 8$  ángulos distintos, dados en la tabla. (1 punto)

$\theta_r$ (rad)	$d_r$ (pixels)	$r$
0	1	$j$
$\tan^{-1}(1/2)$	$\sqrt{5}$	$n_x + n_y - 1 - i + 2j$
$\pi/4$	$\sqrt{2}$	$2n_x + 3n_y - 2 - i + j$
$\tan^{-1}(2)$	$\sqrt{5}$	$4n_x + 4n_y - 3 - 2i + j$
$\pi/2$	1	$4n_x + 5n_y - 5 + i$
$\tan^{-1}(-2)$	$\sqrt{5}$	$5n_x + 5n_y - 7 + 2i + j$
$3\pi/4$	$\sqrt{2}$	$7n_x + 6n_y - 8 + i + j$
$\tan^{-1}-1/2$	$\sqrt{5}$	$8n_x + 7n_y - 10 + i + 2j$

- b) Calcule la matriz  $A_{rp}$ . Para ello, se usarán las distancias  $d_r$  y los índices  $r$  de los rayos que atraviesan cada pixel  $p = (i - 1)n_y + j$  (un rayo para cada ángulo) dados en la tabla. Los índices  $i, j$  se refieren a las coordenadas  $x$  e  $y$  de los pixeles, todos con la misma  $z$ . (Sugerencia: dentro de un bucle en  $i, j$ , calcular el elemento  $A_{r,p}$  para los  $n_a$  rayos que pasan por  $p$ ). (2 puntos)
- c) Obtenga las densidades  $\rho_p$  resolviendo la ecuación (1) y las transforme en una matriz  $\rho_{j,i}$  usando *reshape* (el primer índice es  $j$ , de fila, que corresponde a la coordenada  $y$ ). (1 punto)
- d) Represente dicha densidad usando las funciones *image* o *surf*. (1 punto)

## 2. Reentrada atmosférica

Un satélite de  $m = 5500$  kg comienza su reentrada en la atmósfera desde una órbita equatorial circular a  $h_0 = 150$  km de altura de la superficie de la Tierra. Sobre él actúan la fuerza de gravedad y la resistencia del aire:

$$\mathbf{f} = -G \frac{Mm}{r^3} \mathbf{r} - \frac{1}{2} \rho C_d A v_d \mathbf{v}_d \quad (2)$$

siendo  $G = 6,674 \times 10^{-11}$  Nm<sup>2</sup>/kg<sup>2</sup>,  $M = 5,972 \times 10^{24}$  kg la masa de la Tierra,  $\mathbf{r}$  el vector posición con respecto al centro de la Tierra (y  $r$  su módulo),  $\mathbf{v}_d$  el vector velocidad (y  $v_d$  su módulo) con respecto al aire,  $\mathbf{v}_d = \mathbf{v} - \mathbf{w} \times \mathbf{r}$ , siendo  $\mathbf{w}$  el vector velocidad angular de la Tierra ( $w = 7,292 \times 10^{-5}$  rad/s).  $C_d = 0,4$  es el coeficiente de fricción aerodinámica (“drag” en inglés) y  $A = 12,0$  m<sup>2</sup> la sección transversal del satélite.  $\rho$  es la densidad del aire, que disminuye con la altura  $h = r - R$  como

$$\rho = \frac{m_a P_0}{R_g T} \exp(-m_a g h / R_g T) \quad (3)$$

siendo  $R = 6,371 \times 10^6$  m el radio de la Tierra,  $m_a = 29,0 \times 10^{-3}$  kg/mol la masa molar del aire,  $P_0 = 1,01 \times 10^5$  N/m<sup>2</sup> la presión en la superficie,  $g = 9,82$  m/s<sup>2</sup> la aceleración de la gravedad,  $R_g = 8,314$  J/K/mol la constante de los gases y  $T = 247$  K la temperatura estimada de la atmósfera para el día de la reentrada.

- a) Para simular la entrada, se escribirá una función con la siguiente interfaz

```

function f = reentryForce( r, v, m, A, Cd, T )
% Total force on a satellite re-entering the atmosphere
% Input:
%   r(3,1) : position relative to Earth centre (m)
%   v(3,1) : velocity (m/s)
%   m      : satellite mass (kg)
%   A      : satellite cross sectional area (m^2)
%   Cd     : satellite drag coefficient
%   T      : atmosphere temperature (K)
% Output:
%   f(3,1) : total force, including gravitation and drag (N)

```

Los valores de  $G, M, R, \mathbf{w}, P_0, m_a$  y  $R_g$  se pueden especificar dentro de la función. (2 puntos)

También se escribirá un programa *reentry.m* que

- b) Inicialice la posición y velocidad del satélite, así como los parámetros necesarios para *reentryForce*, defina una función anónima *myforce* adecuada, y llame a la función *rungeKutta* de clase para integrar la trayectoria de reentrada, durante  $t_{max} = 2 \times 10^5$  s, con un intervalo  $\Delta t = 10$  s. La posición inicial puede tomarse sobre el eje  $x$ , siendo  $z$  el eje de rotación de la Tierra. La velocidad inicial será en el mismo sentido de rotación de la Tierra. (1 punto)
- c) Elimine los valores con  $r(t) < R$  y represente la altura en función de la distancia recorrida sobre la superficie (medida en número de vueltas). Para calcular ésta última, puede usarse la función *atan2*, pero deberá después sumarse  $2\pi$  cada vez que el ángulo disminuya. Escriba el tiempo hasta el impacto en horas, el número de vueltas y la distancia recorrida sobre la superficie (en km). (1 punto)
- d) La principal incertidumbre en el punto de caída proviene de la temperatura de la atmósfera, que depende mucho de la actividad solar. Para estimar dicha incertidumbre, se repetirá la simulación para una variación de temperatura de  $\pm 1$  K y se escribirá la diferencia en la distancia horizontal recorrida. (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 *tomography.m*, *reentryForce.m* y *reentry.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.
- La nota máxima será de 10 puntos, aunque la suma de los apartados sea mayor.