Unidad 4

Ecuaciones diferenciales parciales. (c) Ecuaciones hiperbólicas.

La pulsación de una cuerda vibrante de longitud L con sus extemos fijos conduce al siguiente *problema de Cauchy para la ecuación de ondas*: Hallar la función $u(x, t)$ que satisface la ecuación

$$u_{tt} - \alpha^2 u_{xx} = 0,$$

para $0 \le x \le L$, $t \ge 0$, y que cumpla las condiciones homogéneas en los bordes:

$$u(x = 0, t) = u(x = L, t) = 0, \quad t \ge 0.$$

y las condiciones iniciales, determinadas por la desviación inicial de la cuerda en forma de triángulo de altura h en el punto $x = c$ y liberada a partir del reposo:

$$u(x, t = 0) = f(x) = \begin{cases} \dfrac{h}{c}x, & 0 \le x \le c, \\ h\dfrac{(L-x)}{(L-c)}, & c < x \le L, \end{cases}$$

$$\frac{\partial u}{\partial t}(x, t = 0) = 0.$$

Ejercicio 1. Mostrar que la solución *analítica* del problema, obtenida por el *método de separación de variables*, está dada por

$$u(x, t) = \sum_{n=1}^{\infty} \frac{2hL^2}{\pi^2 n^2 c(L-c)} \operatorname{sen}\left(\frac{n\pi c}{L}\right) \times \operatorname{sen}\left(\frac{n\pi x}{L}\right) \cos\left(\frac{n\pi}{L}\alpha t\right).$$

Ejercicio 2. Una solución *numérica* del problema es obtenida por el *método de diferencias finitas*. Considerando una grilla espacio-temporal de paso Δx y Δt, tenemos la red de puntos:

$$x_i = i\Delta x, \quad i = 0, 1, \ldots, n = L/\Delta x,$$
$$t_j = j\Delta t, \quad j = 0, 1, \ldots.$$

Mostrar que el algoritmo de *diferencia finita centrada* determina la aproximación $u_{i,j+1}$ de $u(x_i, y_{j+1})$ según la fórmula:

$$u_{i,j+1} = 2(1-\lambda^2)u_{i,j} + \lambda^2(u_{i+1,j} + u_{i-1,j}) - u_{i,j-1},$$

donde $\lambda = \alpha\Delta t/\Delta x$, con las condiciones iniciales:

$$u_{i,0} = f(x_i), \quad u_{i,1} = 0, \quad i = 1, 2, \ldots, n-1,$$

y las condiciones de contorno:

$$u_{0,j} = u_{n,j} = 0, \quad j = 0, 1, \ldots$$

Ejercicio 3. El programa, cuyo código se muestra más adelante, implementa la determinación de la solución numérica del problema planteado para u/h, tomando $L = 1$, $\alpha = 1$ y $c = 0.8$.

a) Determine la solución numérica con $n = 100$ y un paso Δt que satisfaga la *condición de estabilidad*: $\alpha\Delta t/\Delta x \le 1$. Grafique la solución.

b) Compare la solución numérica con la solución analítica.

c) Repita el proceso para otros tamaños de paso, en particular investigue como se comporta la solución numérica cuando no se satisface la condición de estabilidad.

```
      program ondas
*     ------------------------------------------------------
*     Problema de la ecuación de ondas sobre el dominio
*     0 <= x <= L, t >= 0
*     ------------------------------------------------------
*     Declaración de tipo
*     ------------------------------------------------------
      implicit none
      integer maxpoints ! Máximo tamaño de la red
      parameter (maxpoints = 1000)
      double precision u(0:maxpoints,0:2)
      double precision long,deltax,deltat
      double precision alpha2,lambda2
      integer n,tpasos,twrite
      integer i,j
*     ------------------------------------------------------
*     Parametros del problema
*     ------------------------------------------------------
      long    = 1.0d0  ! Longitud de la cuerda
      alpha2  = 1.0d0  ! Coeficiente de la ec. diferencial
      n       = 100    ! N+1 puntos de la malla en x
      deltax  = long/n ! Paso de la malla en el espacio
      deltat  = 0.01d0 ! Paso de la malla en el tiempo
      tpasos  = 100    ! Número de pasos en el tiempo
      twrite  = 50     ! Imprimir cada twrite pasos temporales
      lambda2 = (alpha2*deltat/deltax)**2
*     ------------------------------------------------------
*     Asignar condiciones de contorno
*     ------------------------------------------------------
      u(0,0) = 0.0d0  ! Nivel 0 del tiempo
      u(n,0) = 0.0d0
      u(0,1) = 0.0d0  ! Nivel 1 del tiempo
      u(n,1) = 0.0d0
      u(0,2) = 0.0d0  ! Nivel 2 del tiempo
      u(n,2) = 0.0d0
*     ------------------------------------------------------
*     Asignar condiciones iniciales
*     ------------------------------------------------------
      do i=1,int(0.8*n)
         u(i,0) = 1.25*deltax*i/long
      enddo
      do i=int(0.8*n)+1,n-1
         u(i,0) = 5.0*(1.0d0-deltax*i/long)
      enddo
      do i=1,n-1
         u(i,1) = u(i,0)
      enddo
*     ------------------------------------------------------
*     Imprimir condición inicial
*     ------------------------------------------------------
      do i=0,n
         write(*,*) i*deltax,0.0d0,u(i,0)
      enddo
      write(*,*)
*     ------------------------------------------------------
*     Imprimir primer paso temporal
*     ------------------------------------------------------
      do i=0,n
```

```
            write(*,*) i*deltax,deltat,u(i,1)
        enddo
        write(*,*)
*       ------------------------------------------------------
*       Computar la solucion en los sucesivos tiempos sobre
*       los puntos interiores de la grilla (extremos fijos)
*       ------------------------------------------------------
        do j = 1,tpasos
            do i=1,n-1
                u(i,2) = 2.0d0*(1.0d0-lambda2)*u(i,1)+
     *                   lambda2*(u(i+1,1)+u(i-1,1))-u(i,0)
            enddo
            ! Imprimir solución cada twrite pasos en el tiempo
            if (mod(j,twrite).eq.0) then
                do i=0,n
                    write(*,*) i*deltax,j*deltat,u(i,2)
                enddo
                write(*,*)
            endif
            ! Reasignar niveles 0 y 1 del tiempo
            do i=0,n
                u(i,0) = u(i,1)
                u(i,1) = u(i,2)
            enddo
        enddo
        stop
        end
```