Métodos Computacionales

Tarea 4 - Ecuaciones Diferenciales Parciales

24-10-2016

La solución a este taller debe subirse por SICUA antes de las 8:00AM del jueves 10 de noviembre del 2016.
Los códigos deben encontrarse en un unico repositorio de `github` con el nombre `NombreApellido_hw4`. Por ejemplo yo debería subir crear un repositorio con el nombre `JaimeForero_hw4`. El último commit de ese repositorio debe ser anterior a las 8:00AM del jueves 10 de noviembre del 2016.
Adentro de ese repositorio deben existir dos carpetas con los nombres `punto_1` y `punto_2` por cada uno de los puntos de esta tarea.
Dentro de cada carpeta deben estar los siguientes elementos.

- (30 puntos) Un código fuente en C que resuelve las ecuaciones diferenciales y produce datos.
- (10 puntos) Un código en Python que lee los datos producidos por el código en C y produce visualizaciones.
- (10 puntos) Un makefile que sique la estructura lógica del punto para compilar y ejecutar el código en C, ejecutar el codigo en Python y borrar todos los archivos auxiliares.

1. **Condensador de placas paralelas**

   Considere un condensador de placas paralelas ubicado en una región bidimensional del espacio, de dimensiones $L \times L$. Las placas tienen un largo l y una separación d entre ellas, como se muestra en la figura 1.

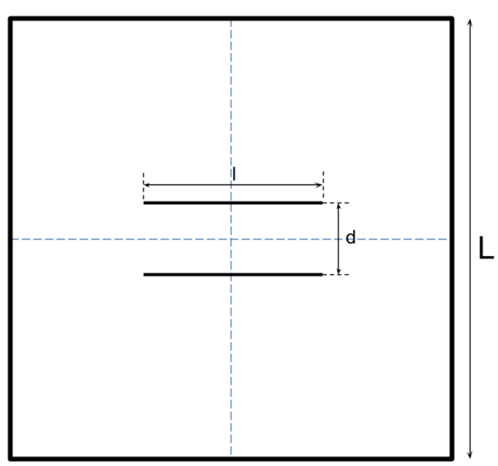

Figura 1: Condensador de placas paralelas.

   Supongamos que existe una diferencia de potencial constante V_0 entre las placas (una de las placas se encuentra a $-V_0/2$ y la otra a $V_0/2$). Además, en el borde de la región tomemos el potencial fijo en 0.

El potencial eléctrico $V(x, y)$ en la región debe cumplir la ecuación de Laplace

$$\nabla^2 V(x,y) = \frac{\partial^2 V(x,y)}{\partial x^2} + \frac{\partial^2 V(x,y)}{\partial y^2} = 0.$$

Así mismo el campo eléctrico está dado por.

$$\mathbf{E}(x,y) = -\nabla V(x,y) = \left(-\frac{\partial V(x,y)}{\partial x}, -\frac{\partial V(x,y)}{\partial y} \right)$$

Para calcular el potencial numéricamente se debe discretizar la región como una matriz de tamaño $L/h \times L/h$, donde h es la separación entre los puntos y utilizar el esquema de diferencias finitas.

Escriba un programa en C (`placas.c`) que encuentre el potencial eléctrico con el método de relajación usando N iteraciones. El mismo programa debe calcular el campo eléctrico a partir de la solución final del potencial. Escriba un código en Python `grafica.py` que grafique el potencial y el campo eléctrico usando `imshow` y `streamplot` en una grafica llamada `placas.pdf`.

Utilice $L = 5$ cm, $l = 2$ cm, $d = 1$ cm, $h = 0{,}02$ cm, $V_0 = 100$ V y $N = 100$ iteraciones.

2. **Cuerda Vibrando**

Considere una cuerda de longitud L descrita por la función $u(x, t)$ que corresponde al desplazamiento con respecto a su posición de equilibrio. Después de una perturbación inicial, la evolución de u está dada por

$$\frac{\partial^2 u(x,t)}{\partial x^2} = \frac{1}{c^2} \frac{\partial^2 u}{\partial t^2} \tag{1}$$

donde $c = \sqrt{T/\rho}$ es una velocidad de propagación con T la tensión de la cuerda y ρ su densidad.

Las condiciones de contorno corresponden a puntos fijos. Como condición inicial se tiene que la cuerda está estirada de forma triangular, con el máximo ubicado a 8/10 de la longitud total de la cuerda con una altura 1. Es decir,

$$u(x, t = 0) = \begin{cases} 1{,}25x/L & x \leq 0{,}8L \\ 5 - 5x/L & x > 0{,}8L \end{cases} \tag{2}$$

Escriba un programa en C (`cuerda.c`) que resuelva esta ecuación y encuentre $u(x, t)$. Escriba un código en Python (`animacion.py`). que produzca un gif animado (`cuerda.gif`) con el movimiento resultante de la cuerda.

Utilice $T = 40$, $\rho = 10$ y $L = 100$ para $0 < t < 200$.