Prueba evaluable de programación con Maxima

Criterios de evaluación

Cada uno de los ejercicios que componen esta prueba evaluable sobre la primera parte de la asignatura Física Computacional 1 se evaluará, de 0 a 10 puntos, de acuerdo a los siguientes criterios de evaluación:

- El código aportado realiza correctamente las tareas que se pedían en el enunciado, cálculos simbólicos y/o numéricos, representaciones gráficas, etc., (sin errores sintácticos): **5 puntos**

- El código está bien estructurado, se entiende claramente lo que se hace en cada parte del mismo, la estructura es lógica y está ordenada: **2 puntos**

- El código realiza las tareas que se piden de manera eficiente: **1.5 puntos**

- El código está documentado con comentarios que facilitan entender qué es lo que se está haciendo en cada parte del mismo, incluyendo descripción del *input* y *output* y la finalidad del código: **1.5 puntos**

La calificación final de esta parte será la media aritmética de las calificaciones obtenidas en todos los ejercicios que forman esta prueba, siempre y cuando se haya obtenido una calificación mínima de 5 puntos en todos ellos. Si uno (o más) de los ejercicios propuestos no alcanzan la calificación mínima de 5 puntos la calificación global de la prueba será *suspenso*, y no se calculará la media.

Nota:

- Es muy importante darse cuenta de que lo que se pide en cada uno de estos ejercicios es la programación de una *función*, no la resolución de un problema concreto.

- En la medida de lo posible ajústese al input y output especificado en cada ejercicio (aunque puede introducir pequeñas modificaciones si lo considera preciso, en ese caso introduzca una breve frase explicando las modificaciones introducidas).

- Muy importante, evite la definición de *variables globales* fuera de estas funciones.

- También le recomendamos que, antes de realizar estos ejercicios, revise la colección de problemas resueltos que puede encontrar en la página de la asignatura, así como las soluciones de las pruebas evaluables anteriores.

Ejercicio 1

La PEC de este curso se va a centrar en el *Método de las Líneas* para la resolución numérica de ecuaciones diferenciales en derivadas parciales con dos variables independientes. Como ya hemos comentado en los apuntes, wxmaxima no dispone de funciones para la integración numérica de ecuaciones diferenciales en derivadas parciales, pero sí dispone de la función `rk`, que permite resolver numéricamente (sistemas de-) ecuaciones diferenciales ordinarias de orden 1. El método de las líneas es un algoritmo numérico muy sencillo que permite re-escribir sistemas de ecuaciones diferenciales en derivadas parciales como sistemas de ecuaciones ordinarias, los cuales se resuelven posteriormente empleando los algoritmos numéricos habituales para EDOs (p. ej. Runge-Kutta).

En esta PEC vamos a emplear dicho algoritmo para resolver la ecuación del calor en una dimensión espacial. Vamos a llamar t a la variable temporal, x a la variable espacial y $f(x,t)$ a la temperatura. Suponemos que hemos escogido las escalas de espacio y tiempo de tal manera que el coeficiente de difusividad es la unidad, en esas condiciones la ecuación que vamos a resolver es

$$\frac{\partial f}{\partial t} = \frac{\partial^2 f}{\partial x^2}$$

Suponemos que queremos resolver esta ecuación para valores del tiempo $t > 0$ y que las condiciones iniciales necesarias para ello ($f(x, t = 0) = f_0(x)$) son conocidas. Por otra parte, suponemos que este problema está planteado para valores de la coordenada x contenidos en un cierto intervalo $[a, b]$ finito y conocido, y que las condiciones de contorno necesarias para realizar el cálculo numérico ($f(x = a, t) = f_a(t)$ y $f(x = b, t) = f_b(t)$) son conocidas. Por supuesto, vamos a suponer que los datos iniciales $f_0(x)$, y las condiciones de contorno $f_a(t)$ y $f_b(t)$ son funciones suaves. En estas condiciones el problema puede resolverse de manera numérica.

El método de las líneas consiste en lo siguiente

- Discretizamos la variable independiente x en una colección de N valores (x_i) equiespaciados en el intervalo $[a, b]$

$$x_i = a + \frac{i-1}{N-1}(b-a), \qquad i = 1, 2, 3, \ldots, N$$

- Discretizamos la variable dependiente $f(x,t)$ en una colección de N funciones de t ($f_i(t)$) resultantes de fijar la variable x de $f(x,t)$ en los valores x_i particulares que acabamos de definir

$$f_i(t) = f(x = x_i, t), \qquad i = 1, 2, 3, \ldots, N$$

  A partir de este momento, para resolver el problema de partida debemos hallar las funciones $f_i(t)$.

De toda la lista de funciones $f_i(t)$, la primera y la última son conocidas, dado que están dadas por las condiciones de contorno del problema

$$f_1(t) = f_a(t), \qquad f_N(t) = f_b(t)$$

Por tanto el problema se reduce a calcular las restantes $f_i(t)$ $(i = 2, 3, \ldots, N-1)$.

- A partir de la discretización realizada para la variable x, aproximamos las derivadas respecto a x por medio de la fórmula de *diferencias finitas* correspondiente (en esta PEC vamos a emplear el algoritmo de *diferencias centradas*). De acuerdo a la anterior discretización de la variable espacial x el espaciado (h) entre valores contiguos de x es constante

$$h = x_{i+1} - x_i = \frac{b-a}{N-1}$$

En estas condiciones, si N es suficientemente grande se cumplirá que $h \ll 1$, y en ese caso las derivadas respecto a x pueden aproximarse por

$$\left.\frac{\partial f}{\partial x}\right|_{x=x_j} = \frac{f_{j+1} - f_{j-1}}{2h}$$

$$\left.\frac{\partial^2 f}{\partial x^2}\right|_{x=x_j} = \frac{f_{j+1} - 2f_j + f_{j-1}}{h^2}$$

donde $f_j = f(x_j, t)$.

- Sustituyendo esta aproximación para el cálculo de las derivadas en la ecuación de partida tenemos que la ecuación en derivadas parciales queda como un conjunto de $N-2$ EDOs

$$\frac{df_i}{dt} = \frac{f_{i+1} - 2f_i + f_{i-1}}{h^2}, \qquad i = 2, 3, 4, \ldots, N-1$$

(recuérdese que f_1 y f_N son datos del problema).

- Resolviendo el anterior conjunto de EDOs con las condiciones iniciales

$$f_i(t=0) = f_0(x_i)$$

por medio de la función `rk` construimos finalmente una solución aproximada a la EDP de partida. Evidentemente, cuanto mayor sea N mayor será el tiempo de cálculo y mejor será la aproximación realizada.

Para el primer ejercicio escriba una función en Maxima que aplique la discretización anterior y genere el conjunto de ecuaciones diferenciales ordinarias que posteriormente resolveremos con `rk`:

input:
- Intervalo de definición de la variable x: $[a, b]$ (suponemos a y b finitos y $b > a$).
- Valor del parámetro N para la discretización de la variable x.
- Condiciones de contorno del problema, dadas por las funciones del tiempo $f_a(t)$ y $f_b(t)$.

output:
- Lista de valores discretizados de x.
- Lista de funciones f_i.
- Lista con los $N-2$ "lados derechos" de las igualdades

$$\frac{df_i}{dt} = \frac{f_{i+1} - 2f_i + f_{i-1}}{h^2}, \qquad i = 2, 3, 4, \ldots, N-1$$

Nota: De esta forma el output de esta función es precisamente el input que necesitamos suministrar a `rk` para proceder a la resolución numérica.

Ejercicio 2

Dado el output de la función anterior, escriba una función en Maxima que resuelva numéricamente dicho sistema de $N-2$ EDOs por medio de la función `rk`:

input:
- Lista de variables dependientes f_i.
- Lista de "lados derechos" de las igualdades $df_i/dt = \ldots$, con $i = 2, \ldots, N-1$).
- Lista de condiciones iniciales $f_i(0) = f_0(x_i)$, con $i = 2, \ldots, N-1$.
- Variable independiente (t) y valores mínimo y máximo de la variable independiente.
- Parámetro de *paso* a emplear en el algoritmo de Runge-Kutta.

output:
- Matriz con los resultados numéricos obtenidos por medio de la función `rk`.

Ejercicio 3

Programe una función en Maxima que permita visualizar el resultado numérico anterior en una gráfica en 3d.

input:
- Lista de valores discretizados de x.
- Variable independiente y valores mínimo y máximo de la variable independiente.
- Matriz con los resultados numéricos obtenidos por medio de la función `rk`.

output:
- Gráfica en 3d de $f_i(t)$ vs. (x, t).

Ejercicio 4

Con mucha frecuencia una gráfica en 2d permite visualizar mejor los resultados que la correspondiente gráfica tridimensional. Para ello vamos a representar $f(x,t)$ frente a x para una serie de valores de t.

input:
- Lista de valores discretizados de x.
- Variable independiente y valores mínimo y máximo de la variable independiente.
- Matriz con los resultados numéricos obtenidos por medio de la función `rk`.
- K: Número de valores de t a emplear en la gráfica bidimensional.

output:
- Gráfica bidimensional con las correspondientes líneas $f(x,t_j)$ vs. x correspondientes a los K valores de tiempo ($t = t_j$, $j = 1, \ldots, K$) que vamos a visualizar.

Observación

Todos los ejercicios que estamos haciendo en esta PEC están planteados para resolver la ecuación de la difusión (ver arriba) con condiciones iniciales ($f_0(x)$) y de contorno ($f_a(t)$, $f_b(t)$) arbitrarias, y también con valores arbitrarios de a y b. Para programar estas funciones y verificar que todo funciona correctamente es necesario asignar valores concretos a todos estos parámetros, de tal forma que podamos llegar a resultados numéricos concretos. Para realizar las pruebas de programación necesarias pueden probar con funciones sencillas, como p. ej.

$$N = 80, \qquad a = -1, \qquad b = +1, \qquad f_0(x) = 0, \qquad f_a(t) = 1, \qquad f_b(t) = 0$$

pero recuerden que lo que estamos pidiendo en esta PEC es un conjunto de funciones que sean capaces de resolver este problema para valores arbitrarios de N, a, b, $f_0(x)$, $f_a(t)$ y $f_b(t)$.

Finalmente, antes de comenzar a programar las funciones de esta PEC es conveniente revisar a conciencia todos los ejemlos resueltos disponibles en el curso virtual.