

DEPARTAMENTO DE MATEMATICA APLICADA
UNIVERSIDAD COMPLUTENSE DE MADRID
Análisis Numérico
PRACTICAS. Hoja 5
Problemas de Contorno

Práctica 24. 1. Crear el fichero tipo función **midisplin.m** que tomando como datos los coeficientes $p(t)$, $q(t)$, $r(t)$ de la ecuación diferencial implemente el método de disparo lineal para cada una de las condiciones de contorno siguientes:

$$x(t_0) = a, \quad x(T) = b$$

$$x'(t_0) = a, \quad x(T) = b$$

$$x(t_0) = a, \quad x'(T) = b$$

$$x'(t_0) = a, \quad x'(T) = b$$

Indicación: utiliza dos variables de entrada adicionales $C1$ y $C2$ respectivamente para indicar cual es la condición de contorno prescrita en t_0 y T respectivamente. Utilizar el convenio de que cuando uno de estos parámetros es 0, entonces estamos prefijando el valor de la función, mientras que si es 1 entonces estamos prefijando el valor de la derivada. Como método de resolución de los problemas de valor inicial, emplea el método de Runge–Kutta de orden 4.

Notación obligatoria: Los ficheros conteniendo las funciones de las dos EDOs necesarias para el método de disparo deben llamarse **fundisplin1.m** y **fundisplin2.m**.

Sugerencia: Se pueden definir las funciones $p(t)$, $q(t)$, $r(t)$ como una *lista de funciones* de nombre **pqr**. Si por ejemplo $p(t) = 0$, $q(t) = 4$ y $r(t) = -4t$, en el fichero de datos **mispracticasm** se escribe:
`pqr=@(t) 0*t , @(t) 4+0*t , @(t) -4*t ;`

y en el fichero **fundisplin1.m**:

```
function dx=fundisplin1(t,x,pqr)
dx=zeros(2,1);
dx(1)=x(2);
dx(2)=feval(pqr{1},t)* x(2)+feval(pqr{2},t)* x(1) +feval(pqr{3},t);
```

2. Crear el fichero tipo script **testmidisplin.m** que ejecute **midisplin.m** y pinte la solución del problema de contorno y su derivada en una misma ventana.

Código obligatorio de colores: Las componentes han de pintarse **en rojo y verde**, respectivamente.

Práctica 25. Utilizar los programas anteriores para resolver los siguientes problemas de contorno. Explorar las soluciones modificando las condiciones de contorno.

1. $x''(t) = 4(x(t) - t), \quad 0 \leq t \leq 1, \quad x(0) = 0, x(1) = 2.$

2. $x''(t) = 3x'(t) + 2x(t) + 3\cos(t), \quad 0 \leq t \leq 5, x(0) = -2, x'(5) = 1.$

3. $x''(t) = \cos(t)x(t) + t, \quad 0 \leq t \leq 10, x'(0) = -2, x(10) = -1.$

4. $x''(t) = (1 - \sin(t))x'(t) + \cos(t)x(t) + \sin(t), \quad 0 \leq t \leq 10, x'(0) = -2, x'(10) = -1.$

Práctica 26. 1. Crear el fichero tipo función **midispnolin.m** que implemente el método de disparo para resolver un problema no lineal con condiciones de contorno

$$x(t_0) = a, \quad x(T) = b.$$

Como método de resolución de los problemas de valor inicial, emplea el método de Runge–Kutta de orden 4 **para cada** valor del parámetro de disparo s_k . Interrumpir el algoritmo cuando el error cometido al disparar con pendiente s_k

$$F(s_k) = x(s_k, T) - b,$$

sea menor que una cierta tolerancia dada.

- Utilizar el **Método de Newton** para generar la iteración de s_k y calcular un cero de $F(s) = 0$. **Sugerencia:** Se pueden definir las funciones $f(t, x, y)$, $\frac{\partial f}{\partial x}(t, x, y)$, $\frac{\partial f}{\partial y}(t, x, y)$ como una *lista de funciones* de nombre `ffxfy`.
- *Opcionalmente, incorporar el **Método de la Secante** para generar la iteración de s_k y calcular un cero de $F(s) = 0$, dados s_0 y s_1

$$s_{n+1} = s_n - \frac{F(s_n)}{F(s_n) - F(s_{n-1})}(s_n - s_{n-1}).$$

Usar como pendiente inicial $s_0 = (b - a)/(T - t_0)$ y $s_1 = s_0 + (b - x(s_0, T))/(T - t_0)$.

- Crear el fichero tipo script **testmidispnolin.m** que ejecute **midispnolin.m** y pinte la gráfica de la solución.

Notación obligatoria: El fichero conteniendo la función de la EDO debe llamarse **funcdispnolin.m**.

Práctica 27. Emplear el programa anterior para los problemas de contorno siguientes

- $x''(t) = 2x^3(t)$, $1 \leq t \leq 2$, $x(1) = 1/2$, $x(2) = 1/3$. Comparar con la solución exacta $x(t) = 1/(1+t)$.
- $x''(t) = 8x^3(t)$, $0 \leq t \leq 1$, $x(0) = 1/3$, $x'(1) = 2$. Comparar con la solución exacta $x(t) = 1/(3+2t)$.
- $x''(t) = (-tx'(t) + x(t) + t)^3 + \frac{1}{t}$, $1 \leq t \leq e$, $x'(1) = 1$, $x'(e) = 2$. Comparar con la solución exacta $x(t) = t \ln(t)$.
- $x''(t) = -(x'(t))^2 + x(t) + t$, $1 \leq t \leq 2$, $x'(1) = 0$, $x'(2) = 2$.

Práctica 28. Emplear el programa **midispnolin.m** para el siguiente problema de contorno

$$\begin{aligned} x''(t) &= \left[2 \sin(t^2) + 8t^2 \right] x(t) - 4t^2 x(t) \log(x(t)) + 2t \sin(t^2) x'(t) \\ x(0) &= e, \quad x(\sqrt{10,5\pi}) = e^2. \end{aligned} \quad (1)$$

Sabiendo que la solución exacta es $solex = \exp(2 - \cos(t^2))$, dibujar la diferencia entre la solución exacta y la aproximada.

Práctica 29. 1. Crear el fichero tipo función **midiffin.m** que tomando como datos los coeficientes $p(t)$, $q(t)$, $r(t)$ de la ecuación diferencial implemente el método de diferencias finitas para el problema de contorno con condiciones

$$x(t_0) = a, \quad x(T) = b$$

Indicación: No almacenar la matriz cuadrada que aparece con este método.

- Crear el fichero tipo script **testmidiffin.m** que ejecute **midiffin.m** y pinte la solución del problema de contorno.
- (Opcional) Modifica el programa principal **midiffin.m**, de manera que pueda resolver problemas con las mismas condiciones de contorno que con el método de disparo.

Práctica 30. Emplear el programa anterior para resolver los problemas de contorno con las ecuaciones lineales de una práctica anterior.