Métodos Numéricos

IMERL - Facultad de Ingeniería - UDELAR

Solución del examen del 14 de Diciembre de 2006

Ejercicio 1.

Parte 1 Explícitos son aquellos en los que para cualquier EDO la expresión de la solución en el paso k+1 (y_{k+1}) se puede expresar analíticamente en función de los pasos anteriores (y_k, y_{k-1}, etc). Serán implícitos aquellos métodos en los que no se pueda lograr lo anterior.

Parte 2 Ver teórico del curso.

Parte 3 Método implícito.

Parte 4 Una primera manera de iterar con el Método del Trapecio es implementando un método Predictor-Corrector. Eso implica obtener una primera estimación de y_{k+1} con un método explícito (por ejemplo Euler Hacia Adelante) y luego con esa estimación utilizar el paso iterativo del Método del Trapecio.

Para la EDO general y un paso temporal h, el paso iterativo del método Predictor-Corrector con Euler Hacia Adelante y Trapecio sería:

$$\begin{aligned} p_{k+1} &= y_k + hf(y_k, t_k) \\ y_{k+1} &= y_k + \frac{h}{2}(f(p_{k+1}, t_{k+1}) + f(y_k, t_k)) \end{aligned}$$

Una segunda manera de iterar sería notando que en la fórmula implícita del paso iterativo del Método del Trapecio tenemos una ecuación no lineal (en principio), siendo la única incógnita de dicha ecuación y_{k+1}. Por lo tanto para iterar el método del trapecio se podría resolver en cada paso una ecuación no lineal, utilizando por ejemplo el método de Newton-Raphson. Un buen punto de arranque para el método de N-R (o el que se use) es sencillamente y_k dado que asumimos que la solución de la EDO es continua.

Para la EDO general y un paso temporal h, el paso iterativo del método utilizando N-R sería:

Para hallar y_{k+1} a partir de y_k, iterar el siguiente método iterativo (N-R) en j hasta $j = n$ tal que para n se cumpla que $\|s_n - s_{n-1}\| < tol$:

$$\begin{aligned} s_0 &= y_k \\ s_{j+1} &= s_j - \frac{g(s_j)}{g'(s_j)} \end{aligned}$$

Siendo $g(x) = x - y_k - \frac{h}{2}(f(x, t_{k+1}) + f(y_k, t_k))$.

Ejercicio 2.

El problema a resolver es el de ajuste de una función no lineal mediante mínimos cuadrados. Ver teórico de Mínimos Cuadrados: Método de Gauss-Newton y deducción de las ecuaciones normales en el problema de mínimos cuadrados lineales.

Ejercicio 3.

Parte 1 Tenemos un cierto valor $F \in \mathbb{R}$ del cual tenemos una expresión $T(h)$ que nos da un valor aproximado de F, más precisamente,

$$T(h) = F + \textstyle\sum_{i\in\mathbb{N}} c_i h^{p_i}, \text{ y } c_i \neq 0 \text{ con } i = 1, 2, ...$$

La sumatoria de la derecha es el error de truncamiento al calcular F como $T(h)$. Entonces buscamos combinar la función $T(h)$ en dos valores distintos para obtener un error de truncamiento de mayor orden del que tenemos con $T(h)$.

Calculamos con $q > 0$,

$$T(h/q)q^{p_1} - T(h) = (F + \sum_{i\in\mathbb{N}} c_i\,(h/q)^{p_i})q^{p_1} - (F + \sum_{i\in\mathbb{N}} c_i h^{p_i})$$
$$T(h/q)q^{p_1} - T(h) = (q^{p_1} - 1)F + \sum_{i\in\mathbb{N}\setminus 1} d_i h^{p_i}$$

Por lo tanto podemos escribir,

$$\frac{T(h/q)q^{p_1} - T(h)}{(q^{p_1} - 1)} = F + \sum_{i\in\mathbb{N}\setminus 1} \frac{d_i}{(q^{p_1} - 1)} h^{p_i}$$

Y entonces esta nueva aproximación de F tiene orden $O(h^{p_2})$ en vez de $O(h^{p_1})$

Parte 2 Programa para la regla de Simpson en Matlab:

Suponiendo que tenemos una función de Matlab llamada f que evalúa $e^{-\frac{x^2}{2}}$ en vectores. Suponemos que el paso cumple $h = \frac{b}{2m}$ para algún m.

```
function T = Simpson(h,b)
x = linspace(0,b,h);
F = f(x)
f0 = F(1);
S1 = F(2:2:end-1);
S2 = F(3:2:end-2);
f2m = F(end);
T = (f0 + 4*S1 + 2*S2 + f2m)*h/3;
```

Parte 3 Dado que el error de trucamiento para la aproximación de la integral por la regla de Simpson es $O(h^4)$, entonces en este caso $p_1 = 4$, llevando el orden de truncamiento a por lo menos $O(h^5)$.

El siguiente programa pide el valor del paso h y del parámetro de la extrapolación de Richardson q. Además hay que especificar el extremo del intervalo de integración b.

```
function T = SimpsonRichardson(q,h,b)
T1 = Simpson(h/q,b);
T2 = Simpson(h,b);
T = (T1*q^ 4 - T2)/(q^ 4 - 1);
```
