Simulación y Modelización
$2^{do.}$ cuatrimestre 2018

Guía Práctica 3
Modelos Físicos: Método de diferencias finitas

Ingeniería Informática
Universidad Nacional de Avellaneda

1. Introducción

1.1. Método de diferencias finitas

Este método permite resolver sistemas de ecuaciones diferenciales (ordinarias y en derivadas parciales). En esta práctica el trabajo se va a centrar en un tipo muy particular de ecuación denominada "*caída exponencial*". Es un tipo de ecuación ordinaria de orden 1.

La referencia bibliográfica es el libro:

- *Finite Difference Computing with Exponential Decay Models (Hans Peter Langtangen, Springer, 2016)*

2. Ejercicios

2.1. Ejercicio 1

1. Discretización del dominio temporal: escribir una función que reciba como parámetros T y dt (ambos de tipos float) y devuelva t (un array de numpy) que contenga la subdivisión del intervalo $[0, T]$ en intervalos de longitud dt (o sea: $t[i] - t[i-1] = dt$)

2. Considerando el problema completo:
   - Ecuación: $u'(t) = -au(t)$ (con $t \in (0, T]$)
   - Condición inicial: $u(0) = I$

   Implementar la regla θ: escribir una función que reciba los parámetros I, a, T, dt, θ (todos de tipo float), y devuelva dos arrays de numpy: t (el vector del punto anterior) u (los valores aproximados de la función en los puntos determinados por t)

2.2. Ejercicio 2

1. Solución exacta: escribir una función que calcule el valor de la solución exacta: $u_e(t) = Ie^{-at}$ para los puntos dados por la discretizacion temporal (ej. 1.1). Sugerencia: utilizar la función *exp* de numpy.

2. Comparar los resultados de las soluciones discretas con la solución exacta:
   - Calcular la solución exacta para $T = 5$, $dt = 0{,}001$
   - Para cada conjunto de parámetros $T = 5$, $I = 1$, $a = 2$, $\theta = \{0; 0{,}5; 1\}$, $dt = \{0{,}1; 0{,}5; 0{,}7; 1{,}25\}$
     - Calcular la solución aproximada
     - Graficar juntas la solución exacta y la solución aproximada
     - Grabar el gráfico en un archivo *png* o *jpg*.
     - Comparar los gráficos con la primera sección del capítulo 2 del libro.

Referencia: `https://matplotlib.org/users/pyplot_tutorial.html`

2.3. Ejercicio 3

La ecuación de *caída exponencial* permite modelar la Ley de Enfriamiento de Newton cuando la temperatura ambiente (T_s) es de 0^o (ver pag. 95 del libro).

Adaptar el programa de ejemplo "circ_color.py"para visualizar el enfriamiento de una partícula:

- Calcular una aproximación de la solución por el método de Crank-Nicholson ($\theta = 0{,}5$), donde $I = 100$, $a = \{0{,}5; 1; 2\}$, $dt = 0{,}1$, $T_s = 0$ (Obs.: notar que "k" es lo mismo que "a"); el intervalo temporal a considerar es: $[0, 100]$
- En la visualización se tienen que ver dos circunferencias (ubicadas en lugares distintos):
  - Una de color constantemente azul que va a representar la temperatura ambiente.
  - Otra que va a ir desplazandose del color rojo al azul y que va a representar el enfriamiento de la partícula a lo largo del tiempo

2.4. Ejercicio 4 (para profundizar)

De acuerdo al modelo de crecimiento poblacional descripto en la pag. 93:

- Construir la regla θ del modelo
- En base al ejemplo "*ecosistema.py*" realizar una visualización del crecimiento poblacional