Mètodes Numèrics i Probabilístics
Grau de MatCAD
Curs 2021-2022

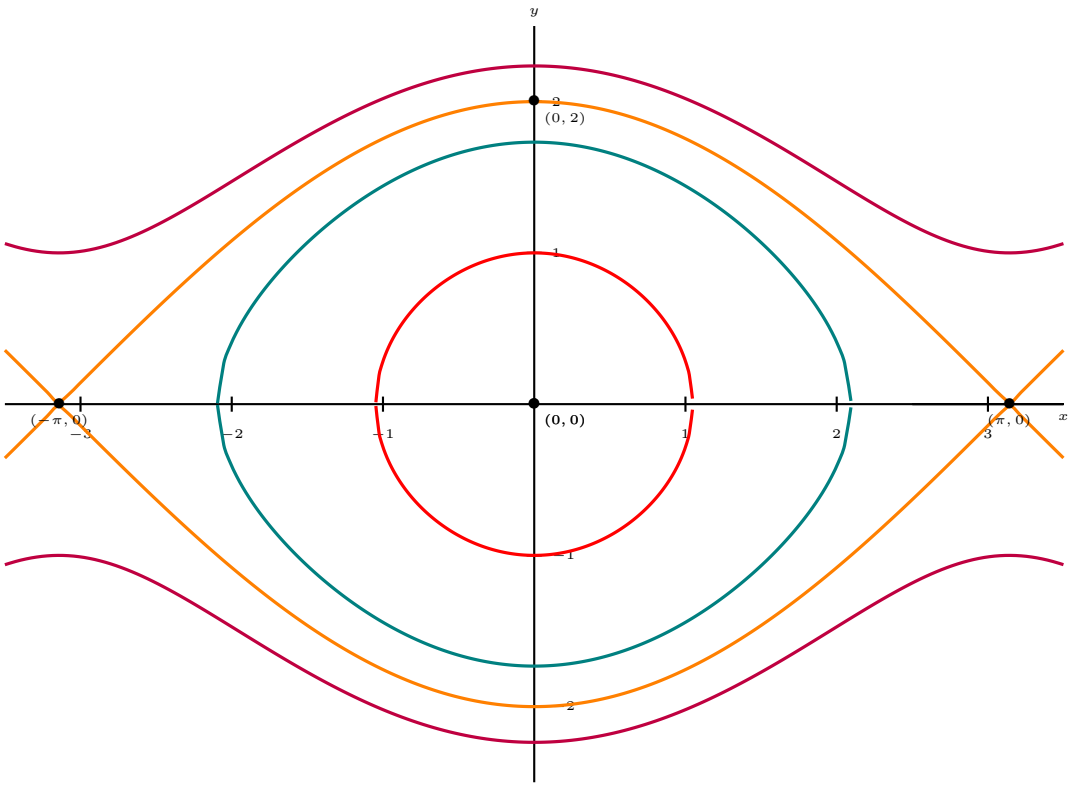
Pràctica 4: Resolució Numèrica d'EDOs

L'equació del pèndol esmorteït és la següent:

$$x'' + \alpha x' + \frac{m}{L}\sin(x) = 0 \tag{1}$$

on m és la massa del pèndol, l la longitud de la corda i $\alpha > 0$ el coeficient de fregament amb l'aire. Com a equació diferencial no té una solució analítica. A més a més, tampoc és un sistema Hamiltonià i per tant no tenim integrals primeres que ens puguin ajudar. Així doncs, per trobar solucions d'aquest problema és molt important l'ús de mètodes numèrics.

1. Programeu un mètode de Runge-Kutta 4 en dues dimensions per resoldre l'equació diferencial. En particular, feu servir el que teniu als apunts.
2. Quin paper juga el paràmetre α?
3. Si $\alpha = 0$, $m = 1$ i $L = 1$, intenteu obtenir el retrat de fase (gràfic en x i y) del pèndol següent:



4. Pel cas $\alpha = 0$, $m = 1$ i $L = 1$, els punts $\big((2k+1)\pi, 0\big)$, $k \in \mathbb{Z}$ són selles, i per tant inestables. Què passa si integreu el sistema amb condició inicial en un d'aquests punts amb amb t_f diferent?

Entrega

1. Fitxers font dels programes realitzats en C:
   (a) Un fitxer anomenat `RK4.c` amb com a mínim una funció amb el següent tipus:
   ```
   void RK4 (double (*f1)(double, double, double, void*),
   double (*f2)(double, double, double, void*), double *x, double *y,
   double x0, double y0, double t0, double tf, int n, void *prm, void *prm2)
   ```

Aquesta funció ha de realitzar el mètode de Runge-Kutta 4 entre t_0 i t_f en el sistema diferencial donat per

$$\begin{cases} \dot{x} = f_1(t; x, y) \\ \dot{y} = f_2(t; x, y) \end{cases}$$

amb n passos i condicions inicials $x(t_0) = x_0$ i $y(t_0) = y_0$. Finalment ha de guardar els resultats de la integració en els punters `*x` i `*y`

(b) Un fitxer anomenat `pendol.c` amb el codi de les funcions a integrar i el main. La crida del programa un cop compilat hauria de ser

`./pendol` α `m L x0 y0 t0 tf n`

El programa hauria d'imprimer per pantalla tres columnes

$$t \quad x \quad y$$

amb la solució trobada de l'equació diferencial. La n és el nombre de passos que voleu que faci el mètode. Filosòficament podeu entendre-la de dues maneres:

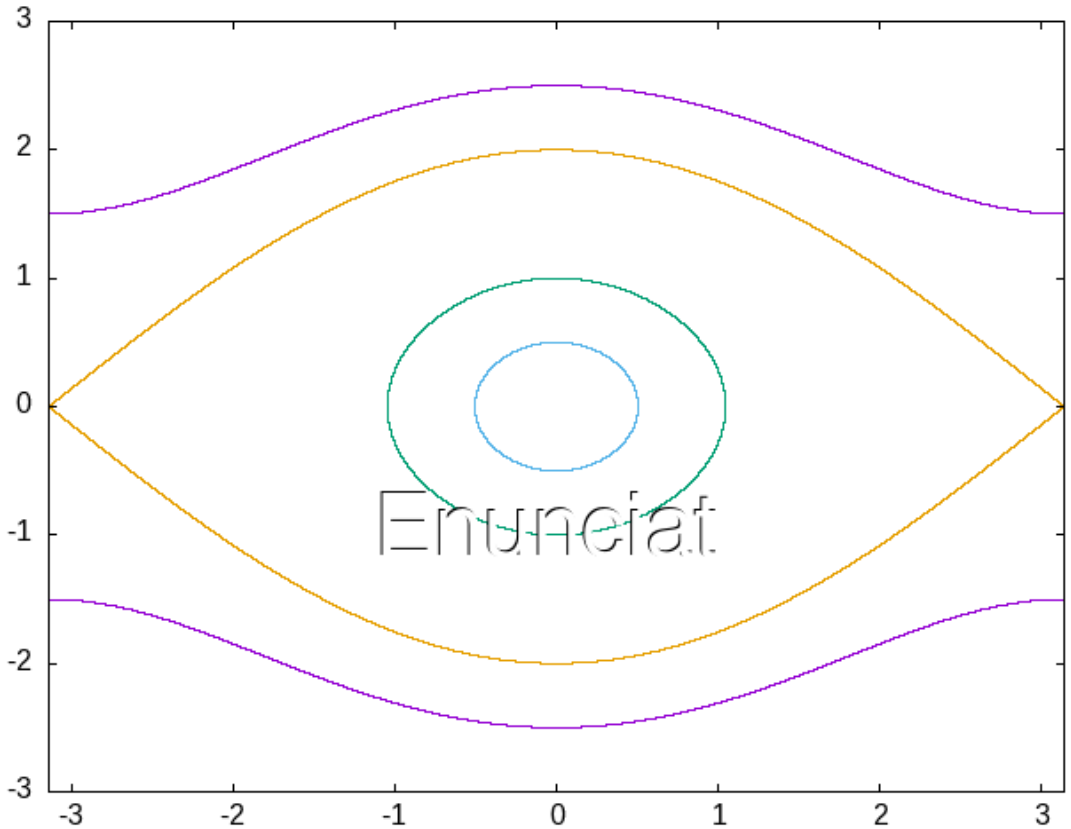
- Fixar un nombre de passos per unitat de temps i llavors fer aquest nombre de crides amb n passos cadascuna.
- Que sigui el nombre total de passos que farà el mètode i cridar-lo n vegades fent només un pas.

2. Informe de pràctiques, on han d'aparèixer una descripció de les funcions del programa, la crida del compilador i exemples concrets per demostrar que les funcions efectivament funcionen (idealment, voleu que jo no tingui necessitat de fer córrer el vostre programa perquè amb l'explicació de l'informe en tinc prou!)

3. L'informe també hauria d'incloure les respostes a les preguntes plantejades al full de pràctiques. En aquest cas, un gràfic val més que mil paraules. (Però no em poseu un gràfic i ja està, expliqueu-lo una mica!)

Comentari

No us espereu que el retrat de fase us surti una cosa tan bonica com la que surt a l'enunciat. La sortida del `gnuplot` és més aviat una cosa com aquesta:



A més a més, algunes comandes de gnuplot que us poden ser útils:

- `plot "foobar.txt" u n:m` - Fa el gràfic utilitzant la columna `n` com a abscissa i la columna `m` com a ordenada.
- `set xrange [a:b]` - El domini del dibuix serà només [a,b]. Això és útil per eliminar coses sobreres i centrar-se en l'important.
- `unset key` - Eliminia la llegenda que surt per defecte
- `set terminal "foobar"` - canvia on dibuixa el gnuplot. Si es posa `postscript` per exemple, ho dibuixa en un fitxer eps. Si es vol dibuixar en un fitxer ha d'anar acompanyat de la comanda `set output`.
- `set output "foo.bar"` - En cas que la terminal dibuixi en un fitxer, determina en quin fitxer haurà de ser.