

SF1692, numeriskt projekt

Pär Kurlberg

September 12, 2023

Contents

1 Språk: python	1
2 "Slumptalet" N	1
3 Existens men inte entydighet	1
4 PANG!	2
5 Inte PANG	2
6 Vad är π?	2
7 Newton!	3
8 Newton revisited (something for nothing)	3

1 Språk: python

All numerisk kod skall skrivas i python.

2 "Slumptalet" N

Låt N vara de två sista siffrorna i ditt personnummer.

3 Existens men inte entydighet

Betrakta BVP

$$y' = \sqrt{|y|}, \quad y(0) = 0$$

- Värm upp med att lösa det analytiskt.
- Lös problemet numeriskt. Kommentarer?
- Hur länge kan du numeriskt få en lösning att "fastna" på $y = 0$ givet att $y(-1) = -1$? (Dvs, låt $y(-1) = -1 + \epsilon$ med ϵ "pyttelitet", och vi säger att y "fastnar" så länge som $|y| \leq 10^{-4}$.)

4 PANG!

Betrakta BVP

$$y' = y^2, \quad y(0) = N/100$$

- Lös problemet analytiskt. När smäller det?
- Prova att lösa problemet numeriskt. Kan du "numeriskt visa" att det smäller? Kan du uppskatta tiden till singulariteten?
- Övertyga dig om (visa!?) att om $y(0) < 0$ så måste $y(t) < 0$ för alla $t > 0$. (Ledtråd: Picard.)
- Kan du hitta numeriska exempel då $y(0) = \epsilon < 0$, men $y(t) > 0$ för något $t > 0$? (Ev. måste du ändra ekvationen lite och ta $y' = Cy^2$ för "stort" $C > 0$.) Att filosofera över: om du skulle skriva en "automatlösare", hur skall denna kunna undvika/detektera djupt felaktiga lösningar av denna typ?

5 Inte PANG

Betrakta BVP

$$y' = y, \quad y(0) = N$$

Lös problemet numeriskt och finn $y(2^k)$ med två korrekta decimaler, för $k = 1, 2, 3, \dots$ (tortera din dator tills den ger upp!)

6 Vad är π ?

Läs början av kapitel 4.24. Betrakta BVP

$$y'' + y = 0, \quad y(0) = 1, \quad y'(0) = 0$$

- Lös analytiskt.

- Definiera $\pi/2$ som första värdet på $t > 0$ så att $y(t) = 0$. Skatta π genom att lösa BVP numeriskt (och skatta första nollstället). Uppvärmning: hur många korrekta siffror kan du få innan din dator smälter? (Prova Euler samt bättre saker.) Efter uppvärmning: finn 20 korrekta decimaler. Ledtråd: du kan be om ledtråd.
- Om vi istället väljer att skatta π genom att hitta roten som ligger närmast 10^3 , hur mycket svårare blir problemet?

7 Newton!

Läs kapitel 3.21. Sätt solen i origo (antag att solen inte rör sig.) Tag en planet P som rör sig i (x, y) -planet, och låt $(x(t), y(t))$ vara planetens position vid tid t .

- Härled ett andra ordningens system av differentialekvationer som beskriver planetens rörelse (byt **inte** till polära koordinater.) Du kan välja massor så att ekvationerna blir så enkla som möjligt.
- Simulera systemet en kort tid genom att lösa numeriskt för lite olika val av initialvärden. (Plotta!)
- Simulera systemet en lång tid genom att lösa numeriskt för lite olika val av initialvärden. Kan du upptäcka något konstigt? (Du kan be om ledtråd.)

Ledtråd: i vårt universum samt i den här kursen bevaras energin.

- Kan du hitta sätt att återställa ordningen i vår simulering? Du kan be om ledtråd.

Ledtråd: sök på semi-implicit Euler (eller symplectic Euler.)

8 Newton revisited (something for nothing)

Tag planetsystem med sol, Jupiter (dvs tung planet) samt jorden (punktformad). Givet ett rymdskepp nära jorden med "delta-v" som är så litet att skeppet ej kan lämna solsystemet (ledtråd: "escape velocity") designa en bana nära Jupiter som funkar som en "gravitational slingshot" så att skeppet kan lämna solsystemet.

- Utforska numeriskt och hitta massor, planetbanor etc så att ovanstående funkar. Ledtrådar: använda astronomiska enheter (jorden ligger då ca 1 AU från solen.) Om problem att pricka Jupiter: använd program för att söka fram initialvillkor. Du kan be om ledtrådar.
- Hur vet du säkert att skeppet verkligen lämnar solsystemet?