MATLAB – uppgifter

Förkunskaper

MATLAB är jätte användervänlig. För uppgifterna 1 – 6 behöver man bara skriva en m.file som har 2(!) rader och ge ett MATLAB – kommando (och naturligtvis plot – kommandot). På appendix A finns några exempel och ytterligare information om det.

Uppgifterna 7 – 10 kräver lite mer. Man måste veta hur man kan genomföra rekursiva beräkningar. Det är dock heller inte svårt. Ett exempel för en rekursiv beräkning finns på appendix B. När du har studerat exemplet vet du hur du ska göra de rekursiva beräkningarna vid uppgifterna 7 till 10.

I alla fall är du välkommen om det finns problem. Jag hjälper gärna. Även ytterligare frågor är naturligtvis välkomna.

Mål

Ett syfte är att demonstrera olika intressenta fenomen i samband med ordinära differentialekvationer.

- Vilka fasporträt som finns för linjära system med konstanta koefficienter (uppgift 1; kvasiperiodiska system ska dock inte behandlas).
- Hur friktion och/eller en periodisk yttre kraft påverkar fjäderpendelns amplitud, fasvinkel och period. Bl a ska resonanseffekten diskuteras (uppgift 2).
- Hur perioden beror på begynnelsevärdena vid den icke – linjära pendeln. Linjärisering ska också behandlas (uppgift 3).
- Styva system (uppgift 4).
- Hopfbifurkation (uppgift 6).
- Instabila system (uppgift 10).

Ett annat syfte är att demonstrera olika fenomen i samband med numeriska beräkningar.

- Styvhet och instabilitet (uppgifterna 4 och 10) är naturligtvis också viktiga i samband med numeriska problem.
- Vad som händer när man lämnar stabiltetsområdet demonstreras vid uppgiften 7 (se också 9).
- Fördelerna av de implicita metoderna när det gäller stabilitetsområdets storlek demonstreras av uppgiften 8 (se också 9).
- Uppgift 9 handlar bl a om noggrannhet.

Uppgifter

1. Låt A vara en 2×2 – matris och $\underline{x}$ en lösning till differentialekvationssystemet

$$\underline{x}'(t) = A \cdot \underline{x}(t), \quad t \in \mathbb{R}.$$

Lösningsbanan $\{\underline{x}(t) : t \in \mathbb{R}\}$ är en ellips eller går mot $\underline{0}$ eller mot ∞, då $t \longrightarrow \infty$. Banan kan t ex vara en spiral eller en rakt linje. Hur banan ser ut beror på matrisen A och begynnelsevärdena. Demonstrera de olika möjligheterna.

2. Diskutera differentialekvationen

$$y''(t) + \alpha y'(t) + \omega_0^2 y(t) = k\cos(\omega t), \quad t \in \mathbb{R},$$

för lämpliga $\alpha, k \geq 0$ och $\omega_0, \omega > 0$. Några intressanta frågor och/eller fenomen:

- Vad sker om $\alpha = 0$ och $\omega_0 = \omega$ (resonanseffekten)?
- Hur påverkar friktionskoefficienten $\alpha > 0$ amplituden och perioden i fallet $k = 0$?
- Hur utvecklar sig perioden på lång sikt om $k \neq 0$ och
  a) $\alpha = 0$ b) $\alpha \neq 0$?

- Finns något som “liknar” resonanseffekten om $\alpha \neq 0$?

3. Plot samtidigt lösningen till pendelekvationen

$$y''(t) + \sin(y(t)) = 0, \quad t \in \mathbb{R},$$

och den motsvarande linjära ekvationen

$$x''(t) + x(t) = 0, \quad t \in \mathbb{R}.$$

Som beynnelsevärden tag $y'(0) = x'(0) = 0$ och $y(0) = x(0) = \alpha$ för olika α. En intressant fråga är naturligtvis hur pendelns period beror på begynnelsevinkeln α. Dessutom är det intressant att kolla på linjäriseringseffekten.

4. Låt

$$A := \begin{pmatrix} -45.0550 & 45.0450 \\ -44.9550 & 44.9450 \end{pmatrix}, \quad \underline{y}_0 := \begin{pmatrix} 0.001 \\ -0.001 \end{pmatrix}.$$

Plot lösningen till begynnelsevärdesproblemet

$$\underline{y}'(t) = A \cdot \underline{y}(t), \quad \underline{y}(0) = \underline{y}_0,$$

på intervallet $[0, 25]$. Efter det plot lösningen på intervallet $[0, 500]$. Vilken skillnad finns? Hur kan man tydliggöra skillnaden vid datorskärmen?

5. Plot fasporträtet till Volterra – Lotkas modellen. Kan man styra systemet genom att döda ett visst antal jägare vid en enda tidspunkt?

6. (Hopfbifurkation) Det ordinära differentialekvationssystemet

$$\begin{aligned} y_1'(t) &= \alpha - y_1(t) - \frac{4y_1(t)y_2(t)}{1 + y_1^2(t)}, \\ y_2'(t) &= \beta y_1(t)(1 - \frac{y_2(t)}{1 + y_1(t)^2}), \end{aligned}$$

beskriver ungefärligt en viss kemisk reaktion; α och β är parameter.

Låt

$$\beta_c := \frac{3\alpha}{5} - \frac{25}{\alpha}.$$

Om $\beta > \beta_c$ då avtar lösningarnas amplituder och lösningsbanorna är spiraler som konvergerar mot någon punkt. Om $\beta < \beta_c$ då oscillerar lösningarna utan dämpning och lösningsbanorna konvergerar mot en sluten bana. Det kallas för Hopf bifurkation.

Undersök systemet. Lämpliga parameter är t ex

$$y_1(0) = 0, \quad y_2(0) = 2, \quad 0 \le t \le 20, \quad \alpha = 10$$

och $\beta = 2$ resp. $\beta = 4$. Särskilt intressant är naturligtvis den kritiska parametern $\beta_c = 3.5$.

7. Använd Eulers metoden för att beräkna lösningen till begynnelsevärdesproblemet

$$y'(t) = -5ty(t)^2 + \frac{5}{t} - \frac{1}{t^2}, \quad y(1) = 1,$$

på intervallet $[1, 25]$. Välj steglängderna $h = 0.19$, $h = .21$ och $h = 0.4$. Jämför med den verkliga lösningen

$$y(t) = \frac{1}{t}.$$

8. Betrakta samma begynnelsvärdesproblem som i uppgiften 7. Den här gången använd den implicita Euler metoden. Som steglängder väl $h = 0.19$; $h = 0.21$; $h = 0.4$ (som förut och dessutom) $h = 1$; $h = 4$.

9. Samma uppgift som 7. och 8. Använd den explicita och den implicita trapetsmetoden och "den klassiska Runge Kutta metoden". Välj stegländerna så att betydelse av stabilitetsområdet blir tydlig och/eller skillnaderna i samband med noggrannhet.

10. Det inhomogena linjära begynnelsevärdesproblemet

$$y'(t) = 50(y(t) - \sin(t)) + \cos(t), \quad y(0) = 0,$$

har exakt en lösning, nämligen

$$y(t) = \sin(t), \quad t \in \mathbb{R}.$$

"Beräkna" lösningen på intervallet $[0, 1]$ både med Eulers metod och steglängden 0.01 och helt enkelt med någon MATLAB – ode – lösare.

Appendix A: Ode – lösare

Exempel 1 För att beräkna lösningen till begynnelsevärdesproblemet

$$x'(t) = -x(t), \quad x(0) = 1,$$

på intervallet $[0, 5]$ kan man göra följande:

1:a Skriv en file som heter a.m

(filenamnet får vara vad som helst; bara att det slutar på .m).

Filens innehåll ser så här ut:

```
function b=c(t,x)
b=-x;
```

2:a Spara filen a.m

3:a Använd följande MATLAB kommandon:

```
>> [t,x]=ode45('a',[0 5],1);
>> plot(t,x)
```

Då har du redan lösningen.

Några anmärkningar till kommandot

```
>> [t,x]=ode45('a',[0 5],1);
```

1. a var filenamnet. Glömn inte tecken ' '. MATLAB förstår inte vad a betyder, men MATLAB förstår vad 'a' betyder.

2. $[0, 5]$ är intervallet där lösningen $x(t)$ undersöks. Så man skriver [7 12] om man vill veta hur lösningen ser ut på intervallet $[7, 12]$ istället.

3. 1 var begynnelsevärdet. Så om man vill hitta lösningen som satisfierar $x(0) = 97$ så skriver man 97 istället av 1.

4. Om man tar bort ; då får man detaljerad information. Så om man skriver

```
>> [t,x]=ode45('a',[0 5],1)
```

då får man en lista med alla t – värdena som har tagits och de motsvarande värden $x(t)$ av lösningen x. I exemplet får man

$$
\begin{aligned}
t &= \\
& \quad 0 \\
& \quad 0.0502 \\
& \quad 0.1005 \\
& \quad 0.1507 \\
& \quad \vdots
\end{aligned}
$$

och

$$
\begin{aligned}
x &= \\
& \quad 1.0000 \\
& \quad 0.9510 \\
& \quad 0.9044 \\
& \quad 0.8601 \\
& \quad \vdots
\end{aligned}
$$

och det betyder att $x(0) = 1.0000$; $x(0.0502) = 0.9510$; $x(0.1005) = 0.9044$; $x(0.1507) = 0.8601$; ...

Exempel 2 Kanske man vill jämföra lösningen i exemplet 1 med lösningen till

$$x'(t) = -\cos^2(t)\sin(\frac{x(t)^2}{1+t^2}), \quad x(0) = 1.$$

Det skulle inte vara särskilt intressant men är en bra möjlighet att lära sig några viktiga MATLAB – kommandon.

Man genomför förstås samma programm som i exemplet ovan:

1:a Skapa en file, som heter, t ex, b.m

Filens innehåll ser så här ut:

```
function b=c(t,x)
b=-cos(t).*cos(t).*sin(x.*x./(1+t.*t));
```

2:a Spara filen b.m

3:a Använd följande MATLAB – kommandon:

>> [t,x]=ode45('b',[0 5],1);

>> plot(t,x)

Nu får du lösningskurvan.

Om du vill jämföra med lösningen i exemplet 1 ovan kan du fortsätta som följande:

>> hold on;

Det här kommandot gör att bilden inte kommer att försvinna.

>> [t,x]=ode45('a',[0 5],1);

>> plot(t,x,'r')

Nu får du samtidigt lösningarna till problemen i exemplen 1 och 2. En lösning har blå färg som vanlig, den andra har röd färg p.g.a. ,'r'. Det finns många andra möjligheter istället av 'r' (andra färger, andra markeringer osv.)

Kommandot "hold off" är motsatsen till kommandot "hold on".

Exempel 3 Hur behandlar man ett system? Betrakta problemet

$$\begin{aligned} x_1'(t) &= 5x_1(t) + 2x_2(t), \\ x_2'(t) &= \sin(x_1(t)), \end{aligned}$$

$$x_1(3) = 4, \quad x_2(3) = 2,$$

på intervallet $[3, 4]$. För att hitta lösningen kan du göra så här:

1:a Skriv en file som heter, t ex, c.m och ser så här ut:

```
function b=c(t,x)
b=[5*x(1)+2*x(2);sin(x(1))];
```

2:a Spara filen c.m

3:a Använd kommandot

>> [t,x]=ode45('c',[3 4];[4;2]);

Nu beräkna MATLAB lösningarna x_1 och x_2. Vi har skrivit "[3 4]" eftersom vi skulle vilja få lösningarna på intervallet $[3, 4]$.

"[4; 2]" vid kommandot motsvarar begynnelsevillkoret $x_1(3) = 4$, $x_2(3) = 2$. Om vi ville har begynnelsevillkoret $x_1(3) = 2$, $x_2(3) = 4$ istället då skulle vi skriva

```
>> [t,x]=ode45('c',[3 4];[2;4]);
```

Det finns nu olika möjligheter att rita bilder:

```
>> plot(t,x(:,1))
```

Då får man lösningen x_1.

```
>> plot(t,x(:,2))
```

Då får man naturligtvis lösningen x_2.

```
>> plot(x(:,1),x(:,2))
```

Då får man banan $\{(x_1(t), x_2(t)) : 3 \leq t \leq 4\} \subset \mathbb{R}^2$.

Naturligtvis finns också möjligheten

```
>> plot(t,x(:,1),t,x(:,2),'r')
```

Då får man x_1 med blå färg och x_2 med röd färg samtidigt. (x_2 ses dock nästan inte vid det här exemplet ty den är så liten jämfört med x_1; man får ett tydligare exempel om man betrakter det korta intervallet $[3, 3.1]$ istället).

Appendix B: Rekursiva beräkningar

En möjlighet att genomföra rekursiva beräkningar med hjälp av MATLAB är att använda while – kommandot. While – kommandot gör att något programm repiteras så länge som ett visst villkor är satisfierat. De följande exemplen ska tydliggöra hur det fungerar. Bakom exemplen ligger en metod som utvecklades för några tusental år sedan. Metoden används för att beräkna rutan av 2 approximativt:

$$a_0 := 1, \quad a_{n+1} := \frac{1}{2}(a_n + \frac{2}{a_n}), \quad n = 0, 1, 2, \ldots$$

Man kan bevisa att $\sqrt{2} = \lim_{n \longrightarrow \infty} a_n$ och ge en feluppskattning som visar att följden konvergerar snabbt.

För att beräkna a_1, a_2, ... , a_{20} kan man göra följande:

1:a Skriv en file som heter, t ex, root.m och har följande innehåll:

n=1; z=1;
while $n < 21$
z=(1/2)*(z+(2/z))
$n = n + 1$
end

2:a

>> root

ger på skärmen bl a olika z som motsvarar a_1, a_2, a_3, ..., a_{20}.

Det kan vara lämpligare att skapa en vektor

$$\underline{a} = (a_1, a_2, a_3, \ldots, a_{20}).$$

Man kan skriva en file, t ex root1.m, som ser så här ut:

```
n=1; z=1;
while n < 21
z=(1/2)*(z+(2/z));
a(n)=z;
n=n+1;
end
```

```
>> root1
```

```
>> a
```

ger vektorn $\underline{a}$ på skärmen.

Det kan vara ännu lämpligare att avsluta beräkningarna när resultatet är tillräckligt noggrant. Istället av "while $n < 21$" kan man t ex skriva "while abs$(z * z - 2) > 0.0001$". Som vektor $\underline{a}$ får man

$$a =$$
$$1.5000 \qquad 1.4167 \qquad 1.4142$$

i det här fallet. Kom ihåg att

```
>> abs(y)
```

beräknar y:s belopp.