Hans Thunberg
KTH Matematik

Maple och Ordinära Differentialekvationer; några användbara kommandon

Här nedan finns en kortfattad beskrivning av några Maple-kommandon som är användbara när man studerar ordinära differentialekvationer. Vissa av kommandona finns illustrerade i ett separat dokument, ett Maple-worksheet, som du kan välja att läsa som statisk grafisk fil (mapleinstr_ode_ex.pdf) eller som en körbar Maplefil (mapleinstr_ode_ex.mws). I texten härnedan hänvisar **Exempel n** till exempel n i denna Maplefil. Mer hjälp finner du i Maples on-line hjälp.

1. Ordinära differentialekvationer, symbolisk lösning

- `diff(y(x),x)` definierar derivatan av funktionen $y(x)$ Andraderivatan fås med kommandot `diff(y(x),x$2)` osv. (**Exempel 1**).
- Om $y(x)$ är ett explicit uttryck beräknas $y'(x)$. (**Exempel 2**).
- `dsolve` används för att söka lösningar till ordinära differentialekvationer. Vill vi lösa ekvationen

  $$\frac{dy}{dx} - y = x^2$$

  skriver vi (**Exempel 3**)

  ```
  dsolve(diff(y(x),x)-y(x)=x^2,y(x));
  ```

  `dsolve` används här med två argument, det första anger den ekvation vi vill lösa, det andra talar om att vi söker y som funktion av x. *Observera att $y(x)$ måste skrivas som* `y(x)` *även i ekvationen, man kan inte skriva bara* `y`.
  - Man kan också ge initialvillkor eller randvillkor. Dessa ingår då i det första argumentet. Ekvationerna åtskiljs med komma-tecken (,), och hela första argumentet avgränsas av krull-parenteser ({ }). (**Exempel 4**):

    ```
    dsolve({diff(y(x),x$2)-y(x)=x^2,y(0)=0,D(y)(0)=1},y(x));
    ```

    Observera hur villkor på första-derivatan skrivs; t.ex. skrivs $y'(0) = 1$ som `D(y)(0)=1`.
  - Med ett extra sista argument `implicit` (eller `explicit`) kan man ibland tvinga `dsolve` till att att svara på implicit (explicit) form om den explicita (implicita) är svårbegriplig, (**Exempel 5**). I Maple version V.4 eller äldre anger man istället `implicit=true` (`explicit=true`)
- I **Exempel 5** har vi utnyttjat tekniken att först lagra differentialekvationen i en variabel med kommandot

  ```
  deq:=diff(y(x),x)=1/((y(x)-1)*(y(x)-2)*(y(x)-3));
  ```

  och sedan anropa denna variabel som ett argument till `dsolve`

  ```
  dsolve(deq,y(x));
  ```

  Se vidare nedan.

2. Att plotta riktningsfält och lösningskurvor till ODE

- `DEplot` används för att plotta riktningsfält och för att numeriskt bestämma lösningskurvor. Detta kommando finns i paketet `DEtools` som laddas med `with(DEtools):`
- Använd `DEplot` med följande fem argument:
  1. Differentialekvationen 2. Variabler 3. Oberoende variabels interval 4. En lista av listor med initialvillkor 5. Beroende variabels interval.
- Första ordningens ekvationer. Vi exemplifierar med ekvationen ovan (**Exempel 6**):

  ```
  DEplot(diff(y(x),x)-y(x)=x^2,y(x),x=-1..1,[[y(0)=0],[y(1)=2]],y=-2..2);
  ```

  Varje initialvillkor ger en lösningskurva som uppfyller initialvillkoret (om någon sådan finns). Man kan också utelämna initialvillkoren, då ritas bara riktningsfältet. Vill man undertrycka riktningsfältet lägger man till ett sista argument `arrows=NONE` (**Exempel 7**). Den beroende variabelns interval kan utelämnas om man angivit initialvärden, men måste anges om man bara vill rita riktningsfältet.
- Andra ordningens ekvationer. Här krävs ju som bekant normalt extra villkor för att en lösning skall vara bestämd. Dessa anges parvis i de inre listorna, exempelvis

  ```
  ...,[[y(0)=0,D(y)(0)=1],[y(1)=2,D(y)(0)=0]], ...
  ```

- Man också ändra färgerna i plotten. Exempel: `color=BLUE` ger blå färg åt riktningsfältet, medan `linecolor=BLACK` ger lösnininingskurvorna i svart. Som alla andra extra-argument ges de på slutet:

  ```
  DEplot(diff(y(x),x)-y(x)=x^2,y(x),x=-1..1,[[y(0)=0]],linecolor=BLACK);
  ```

  Se **Exempel 7**.
- `DEplot` arbetar med en steglängd som ges som 1/20 av längden av det interval som angivits för den oberoende variabeln. Ibland är detta alldels för grovt. Man kan då korta ner intervallet för den oberoende variabeln, eller tvinga fram en kortare steglängd genom att ge ett sista extra argument `stepsize=l`, där `l` är ett tal som anger den steglängd man önskar (**Exempel 8**).

3. Några tips om hantering av ekvationer och plottar

- Ofta är det enklare och överskådligare att först lagra ekvationer och initialvärden i varsin Maple-variabel, och sedan anropa dessa variabler som argument till `dsolve` respektive `DEplot` (**Exempel 9**):

  ```
  deq:=diff(y(x),x)-y(x)=x^2;
  init:=y(0)=0;
  inits:=[[y(0)=0],[y(1)=2]];
  dsolve({deq,init},y(x));
  DEplot(deq,y(x),x=-1..1,inits);
  ```

- Man kan lagra en plot i Maple i en variabel (**Exempel 10**):
  `plot1:= plot( f(x), x=a..b ):`
  Observera kolon, ej semi-kolon på slutet!
  Vill man sedan titta på plotten använder man kommandot `display`.
  Detta kommando finns i paketet `plots` som laddas med kommandot
  `with(plots);`.
  Man ger sedan kommandot
  `display(plot1);`
  Man kan även titta på flera grafer i samma figur:
  `display({plot1,plot2});`
  Observera krull-paranteserna { } *!*

4. Att rita fasportätt för system av 1:a ordningens autonoma ekvationer

Först laddar vi `DEtools` paketet med
`with(DEtools):`
Vi använder nu kommandot `DEplot` för att rita vektorfält och fasporträtt. Här följer ett exempel:

`eq1:=diff(x(t),t)=2*sin(y(t)^3);`

(1:a autonoma ekvationen)

`eq2:=diff(y(t),t)=-sin(x(t)^3);`

(2:a autonoma ekvationen)

`inits:=[[x(0)=0,y(0)=1],[x(0)=1,y(0)=0]];` (initialvillkor till två banor)

```
DEplot({eq1,eq2},{x(t),y(t)},t=0..10,inits,stepsize=0.2,linecolor=blue,
title=`Ett exempel`);
```

De tre sista argumenten till `DEplot` är inte obligatoriska. `stepsize` avgör hur små steg som ska tas vid beräkning av lösningskurvorna. Ju mindre värden desto bättre precision, men desto längre beräkningstid. `linecolor` sätter färgen på lösningskurvorna (vektorfältets färg kan ändras med `color`). Med `title` kan man ange en rubrik till sin plot. Observera att argumentet till `title` ska stå inom bakåtlutand citat-tecken (‘ ‘). Se **Exempel 11**.