

Hur man skaffar sig tillgång till programmen som ritar fasporträtten

Om du inte redan har ett MATLAB-bibliotek i ditt eget rotbibliotek skapa ett sådant med hjälp av kommandona (varje kommando avslutas med Return)

```
cd
```

```
mkdir matlab
```

Gå sedan ner i ditt MATLAB-biblioteket med kommandot

```
cd matlab
```

Kopiera slutligen in fem förberedda filer till ditt MATLAB-bibliotek med hjälp av kommandot

```
cp ~/jorgen/odefiles/* .
```

OBS : Kommandot slutar med mellanslag och punkt.

Du bör då att ha kopierat över följande filer till ditt MATLAB-bibliotek :

“ODE” , “fjader.m” , “linear2d.m” , “pars.m” , “grafik.m”

Du kan kontrollera det genom att ge kommandot *ls*. Om allt är rätt skall du se dessa fem filers namn uppräknade.

Den första filen sätter MATLAB's sökvägar, så att fasporträttprogrammen kan hittas. Den andra och tredje filen är exempel på filer som definierar ett system av ordinära differentialekvationer. Den fjärde filen , alltså “pars.m” , innehåller parametrar som behövs för att rita fasporträtt. Den femte innehåller parametrar för att påverka grafiken.

Hur man startar programmen

Varje gång du vill använda fasporträttprogrammen skall du gå till ditt MATLAB-bibliotek (*cd matlab*) och ge kommandot

```
ODE
```

OBS : Stora bokstäver !

Det första du gör när du kommit in i MATLAB är att ge kommandot :

```
» startode
```

i MATLAB-fönstret.

Hur man använder programmen

Du måste använda en editor för att skriva och ändra textfiler. Om du skriver

```
» emacsinfo
```

i MATLAB-fönstret får du en enkel information om editorn EMACS.

Öppna två fönster med din editor. I det ena fönstret öppnar du den fil i ditt MATLAB-bibliotek som heter “pars.m”, i det andra öppnar du den aktuella ekvationen. Jag kallar den senare filen för ekvations-filen. Exempel på sådan filer är “linear2d.m” eller “fjader.m”. Ekvations-filen måste också ligga i ditt MATLAB-bibliotek och ha ett namn som slutar på “.m”.

Du får en översikt över program och parametrar om du i MATLAB-fönstret ger kommandot

```
» odeinfo
```

Kommandona fas, forts, pilar, tid

Ge följande kommandon i MATLAB-fönstret och studera resultatet.

» `fas`

Computing

Plotting

Kommentar : `fas` ritnar banor med de startpunkter som återfinns i "pars.m"

» `forts`

Kommentar : `forts` ritnar banor med start från senaste ritade banors slutpunkter.

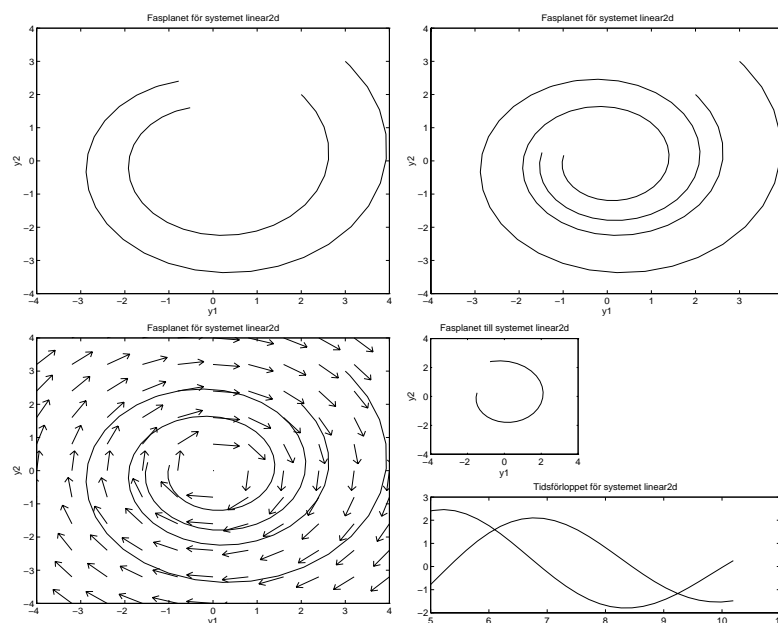
» `pilar`

Kommentar : `pilar` ritnar riktningsfält. (Man kan reglera hur tätt pilarna ritas med hjälp av parametrarna `var1_gitter` och `var2_gitter` i filen "grafik.m")

» `tid`

Kommentar : `tid` ritnar lösningskurvor och banor för den första av de senast valda startvärdena.

Nu bör man ha fått följande figurer :



Kommandona fasplanet, pilar2, clr

Med kommandot `fasplanet` återvänder man till fasplanet och den senast ritade banan. Programmet `pilar2` ritnar riktningsfältet så att pilarnas längd antyder fältets relativa styrka.

Kommandot `clr` tömmer grafikfönstret.

Att ändra koordinataxlar, startpunkter och andra parametrar

Låt oss ändra startvärden och tidsintervall. Vi går då till filen "pars.m" och ändrar i den genom att skriver in nya värden :

```
stoppTid = 10;
```

```
startPunkter = [4 -2];
```

(Ändra inget annat än dessa två värden!) Spara!

Gå sedan tillbaka till MATLAB-fönstret och ge kommandot :

```
» clr;fas
```

Låt oss nu ändra koefficienterna i ekvationen.

Gå in i ekvations-filens fönster (alltså filen "linear2d.m") och skriv in nya parametervärden, till exempel

```
p1=0.1; ( GLÖM INTE SEMI-KOLON! )
```

```
p2=3.1;
```

```
p3=-0.7;
```

```
p4=-0.3;
```

Spara!

```
» fas
```

Eftersom man inte ser hela banan kan det finnas anledning att ändra axeldimensionerna.

```
» axdim
```

Vänster ändpunkt på den horisontella axeln : -8

Höger ändpunkt på den horisontella axeln : 8

Nedre ändpunkt på den lodräta axeln : -5

Övre ändpunkt på den lodräta axeln : 5

Att skriva in en ny ekvation

Låt oss skriva in det så kallade Lorenz-systemet

$$y_1' = \sigma \cdot (-y_1 + y_2),$$

$$y_2' = r \cdot y_1 - y_2 - y_1 \cdot y_3,$$

$$y_3' = -b \cdot y_3 + y_1 \cdot y_2$$

med parametervärdena $\sigma = 10$, $r = 28$, $b = 8/3$

och startvärdena $y_1(0) = 10$, $y_2(0) = 2$, $y_3(0) = 40$.

Vi skall rita banan i y_1y_3 -planet.

Spara först filen "linera2d.m" under nytt namn. Den nya filen skall heta "lorenzekv.m". (GLÖM INTE .m !) Skriv in och spara följande innehåll :

```
function yprim = lorenzekv(t,y)
```

```
sigma=10; ( SEMI-KOLON! )
```

```
r=28;
```

```
b=8/3;
```

```
yprim=[ sigma*(-y(1)+y(2));  
        r*y(1)-y(2)-y(1)*y(3);  
        -b*y(3)+y(1)*y(2)];
```

OBS: Skriv inga mellanslag mellan symbolerna.

Sedan ändrar vi i filen "pars.m" så att :

```
Namn='lorenzekv';
```

```
startTid = 0;
```

```
stoppTid = 5;
```

```
startPunkter = [10 2 40];
```

```
var1 = 1;
```

```
var2 = 3;
```

Spara! För att banan skall synas ändrar vi axeldimensionerna:

» **axdim**

och matar successivt in värdena -20, 20, 0, 50

» **fas**

Kommandot **komet** ritar ett "komet"-huvud som följer den senast ritade banan med rätt tidsdynamik.

Prova också kommandot **tredim** som ritar banor med tre rumskoordinater tredimensionellt så att tidsförloppet syns.

Sammanställning av kommandon

Avbryt pågående beräkningar:

tryck Ctrl - C (dvs tryck på knapparna Ctrl och C samtidigt).

Översikt över fasporträtt-kommandon

fas	Beräknar lösningar och "stämplar in" banor
forts	Fortsätter rita från senaste slutpunkter
komet	Ett komethuvud följer en redan ritad bana med rätt hastighet
tid	Ritar tidsförlopp och banan för den senast beräknade lösningen
tredim	Ritar den senast beräknade lösningens bana i tre dimensioner med korrekt tidsförlopp (förutsätter tre rumsdimensioner)
fasplanet	Återvänder till fasplanet och ritar den senaste banan
pilar	Ritar riktningsfält med lika långa pilar
pilar2	Ritar riktningsfält med pilar vars längd antyder fältets styrka
clr	Tömmer grafikfönstret
axdim	Ändrar grafikfönstrets axeldimensioner
steg	För att ändra beräkningsnoggrannhet vid beräkning av banor och tidssteg vid beräkning av kometrörelsen
pars	Aktiverar den parametersättning som skrivits in i filen "pars.m"
grafik	Aktiverar grafikparametrarna i filen "grafik.m"
startode	Aktiverar programmets olika variabler
bytinst	Väljer programinställning för kontinuerliga eller diskreta system
odeinfo	Beskriver kortfattat hur programmet fungerar
emacsinfo	Ger rudimentär hjälp om editorn EMACS
quit	För att lämna MATLAB

Kommandot **help** följt av kommandonamn ger mer eller mindre omfattande information om vad kommandot uträttar. Exempel : **help fas**.

Kommandot **tips** ger tips för den datorvane om bekväma arbetssätt.

I biblioteket *~jorgen/odelib/* finns ett par ytterligare ekvationer.

Några nyttiga kommandon:

print	Skriver ut grafikfönstret på skrivaren
print -deps filnamn	Sparar grafikfönstret i s.k. .eps-format
what	Ger en lista över MATLAB-filer i aktuellt bibliotek
zoom	För att zooma in och ut. För mer info skriv » help zoom
figure	För att rita flera separata bilder.

En varning: Använd inte globala variabler! Det fungerar inte tillsammans med fasporträtt-kommandona.