

## Laboration 2

Laborationen görs i grupper om två, men examineras individuellt genom datorprov. Vid datorprovet kommer ni att ha tillgång till den kod ni skrivit för uppgifterna nedan. Provet kommer även att innehålla teorifrågor om de koncept som arbetas med i laborationen. För mer information, se kurshemsidan.

Utnyttja gärna alla schemalagda handledningstillfällen för att ställa frågor. Häftet "Matlab 8 i korthet", liksom den interaktiva "Matlab Onramp", ger en stegvis introduktion till MATLAB som kan vara användbar (se Canvas). Flitigt utnyttjande av MATLABs `help`-kommando rekommenderas också.

För var och en av uppgifterna nedan laddas en Matlab fil upp till Canvas, med angivet namn. Lägg i funktioner som används i samma fil, så att allt inga andra filer behövs för att lösa uppgiften. Lösningar till teoriuppgifter och svar på frågor behöver inte laddas upp, men se till att du gör och förstår dem, dessa koncept testas på datorprovet.

### 1. Dollarkurser – minstakvadratmetoden

På kurshemsidan finns datafilen `dollarkurs.mat` som innehåller dagsnoteringar för dollarkursen under två år, från 1a januari 2009 till 31a december 2010. Läs in filen i MATLAB med kommandot `load`. Kursen för dag  $i$  betecknas  $X_i$ , där  $i = 1, \dots, N$  och  $N = 730$ .

#### UPPGIFT 1. MINSTAKVADRATMETODEN

Skriv ett MATLAB-program `MKV.m` som löser uppgifterna nedan.

- a) Anpassa en linjär modell,  $f(t) = c_0 + c_1 t$ , i minstakvadratmening till dollarkursen och bestäm koefficienterna  $c_0$  och  $c_1$ . Plotta data  $X_i$  tillsammans med den anpassade kurvan. Skriv ut värdena på koefficienterna och medelkvadratfelet

$$E = \frac{1}{N} \sum_{i=1}^N (X_i - f(i))^2,$$

Plotta även felet mellan den linjära modellen och data.

- b) Du kan beräkna ditt resultat i a), antingen genom att forma normalekvationerna och använda Matlabs backslash på det kvadratiske systemet, eller genom att använda backslash direkt på det överbestämda systemet. Hur stor är skillanden i koefficienterna som du får med de två tillvägagångssätten? Beräkna konditionstalet för matrisen för normalekvationerna (Matlabs `cond` kan användas). Hur är konditionstalet definierat, och vad säger det om din garanterade noggrannhet? Varför använder Matlab en annan metod än att forma normalekvationerna?

- c) Felet i den linjära modellen tycks vara periodiskt. Uppskatta periodlängen  $L$  från plotten i förra deluppgiften. Anpassa därefter följande modell till dollarkursen:

$$f(t) = d_0 + d_1 t + d_2 \sin(2\pi t/L) + d_3 \cos(2\pi t/L).$$

Plotta resultatet och felet, som ovan. Skriv ut värdena på koefficienterna och medelkvadratfelet.

- d) Låt nu även  $L$  vara en okänd parameter. Modellen blir då olinjär. Hitta hela den parameteruppsättningen  $d_0, d_1, d_2, d_3, L$  som ger bäst approximation av data i minstakvadratmening. Använd Gauss-Newtons metod med resultatet från deluppgift (c) som startgissning. Skriv ut värdena på koefficienterna, inklusive  $L$ , och medelkvadratfelet.

Plotta slutligen data  $X_i$  och alla de tre modellernas anpassning i samma figur.

- Hur mycket steg eller föll dollarkursen per dag under perioden enligt den linjära modellen? Ger de andra modellerna väsentligt annorlunda resultat för denna långsiktiga trend?
- Jämför medelkvadratfelen i de olika modellerna. Vilket är störst/minst?

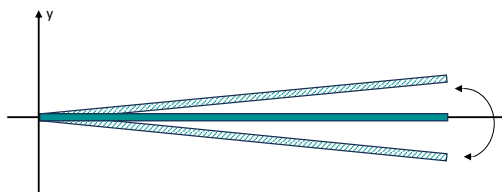
**Skicka in programmet MKV.m**

## 2. Svängande balk

Rörelsen hos en svängande balk, se Figur 1, kan beskrivas av följande ekvation för dämpad svängning

$$y(t) = Ae^{-\alpha t} \cos(\omega t)$$

där  $y$  är förskjutningen från jämviktsläget ( $y = 0$ ) och  $t$  är tiden. Låt  $A = 9$ ,  $\alpha = 1/2$ ,  $\omega = 4$ .



Figur 1: Balk som svänger kring sitt jämviktsstillstånd

## UPPGIFT 2. NUMERISK LÖSNING AV OLINJÄR EKVATION

Nu vill man ta reda på den sista tidpunkten (kalla denna  $t_H$ ) då balkens positiva förskjutning från jämviktsläget blir lika med ett givet värde  $H$ . Skriv ett MATLAB-program `Balk.m` som löser uppgifterna **a)**-**c)** nedan, och svara på fråga **d)**.

- a)** Givet  $H = 0.5$ , bestäm tidpunkten  $t_H$  med Newtons metod. Felet i approximationen till  $t_H$  ska vara mindre än  $10^{-8}$ . Redovisa startvärde, antal iterationer och resultatet. Från resultatet ska det framgå att Newtons metod har kvadratisk konvergens.
- b)** Givet  $H = 0.5$ , bestäm tidpunkten  $t_H$  med sekantmetoden. Felet i approximationen till  $t_H$  ska vara mindre än  $10^{-8}$ . Redovisa startvärden, antal iterationer och resultatet. Behöver vi fler eller färre steg med sekantmetoden för att nå ett fel som är mindre än  $10^{-8}$ . Motivera ditt svar.
- c)** Givet  $H = 4.1355432362$ , bestäm tidpunkten  $t_H$  med Newtons metod. Studera konvergensthastighet och känsligheten vad gäller val av startgissning. Vad beror förändringen jämfört med **a)** på?
- d)** Beteckna approximationen till  $t_H$  från Newtons metod med  $\tilde{t}_H$  och antag att  $|y(\tilde{t}_H) - H| = \delta$ , där  $\delta$  är ett lite tal. Vad kan du då säga om  $|\tilde{t}_H - t_H|$  för de två olika  $H$  värdena i **a)** och **c)** ovan?

Tips! Anteckningen om *Fel och störningsanalys* av Olof Runborg som finns på kurshemsidan.

**Skicka in programmet `Balk.m`**

### 3. Att anlägga en väg

Man vill anlägga en väg från  $P_0$  till  $P_4$ . Men eftersom marken inom det streckade området inte är lämpligt som underlag för en väg måste vägsträckningen gå runt området, se Figur 2.

Uppgiften går ut på att bestämma en vägsträckning som går genom punkterna  $P_0$ ,  $P_1$ ,  $P_2$ ,  $P_3$  och  $P_4$ , se Figur 2 a). Koordinaterna för  $P_0 = (0, 0)$  och för  $P_4 = (1020, 0)$  är kända men koordinaterna för punkterna  $P_1$ ,  $P_2$  och  $P_3$  måste bestämmas.

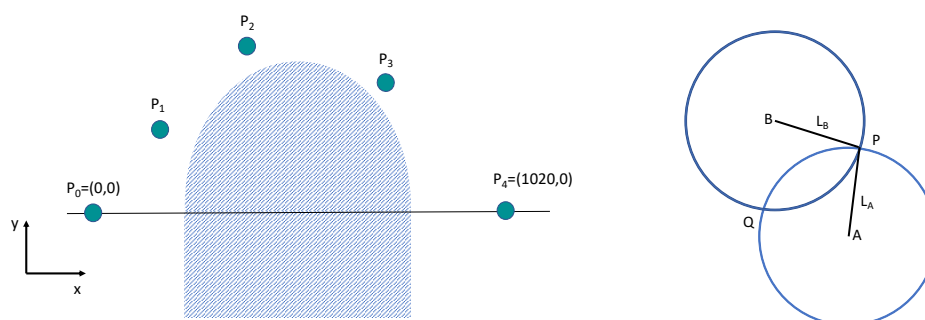
#### Att bestämma koordinaterna - Inbindning

För att bestämma koordinaterna för en punkt  $P$  mäter man avstånden från  $P = (x_P, y_P)$  till två punkter med kända koordinater,  $A = (x_A, y_A)$  och  $B = (x_B, y_B)$ .

Om vi kallar avstånden mellan  $P$  och  $A$  och  $P$  och  $B$  för  $L_A$  respektive  $L_B$  så får vi följande ekvationssystem att lösa för  $x_P$  och  $y_P$ .

$$\begin{aligned}(x_P - x_A)^2 + (y_P - y_A)^2 &= L_A^2 \\ (x_P - x_B)^2 + (y_P - y_B)^2 &= L_B^2.\end{aligned}$$

Ekvationerna beskriver två cirklar med centrum i  $A$  och  $B$  och med radier som ges av  $L_A$  och  $L_B$ , se Figur 2 b). Detta sätt att bestämma en punkts okända koordinater kallas inom



Figur 2: Vänster: Skiss av vägbygget. Höger: Bestämning av koordinater för punkten  $P$  med inbindning.

geodesin för inbindning och är det som satellitnavigeringssystemet GPS utnyttjar.

Notera att ekvationssystemet kommer att ha två lösningar (skärningspunkter) eftersom det finns två punkter,  $P$  och  $Q$ , som båda ligger på samma avstånd från punkterna  $A$  och  $B$ . I vårt fall är det punkten  $P$ , det vill säga den högra skärningspunkten vi vill beräkna.

I tabellen nedan finns för var och en av punkterna  $P_1$ ,  $P_2$  och  $P_3$  punkter  $A$  och  $B$  med kända koordinater samt uppmätta avstånd  $L_A$  och  $L_B$ .

$P$	$A = (x_A, y_A)$	$B = (x_B, y_B)$	$L_A$ [m]	$L_B$ [m]
$P_1$	(170, 950)	(160, 1008)	60	45
$P_2$	(420, 2400)	(370, 2500)	75	88
$P_3$	(670, 1730)	(640, 1760)	42	57

Koordinaterna är angivna i meter från punkten  $(0, 0)$ .

### UPPGIFT 3. NEWTON FÖR SYSTEM OCH INTERPOLATION

Skriv ett MATLAB-program `Road.m` som löser uppgifterna nedan.

- a) Börja med att bestämma koordinaterna (med lämplig tolerans) för punkterna  $P_1$ ,  $P_2$  och  $P_3$  med hjälp av inbindning. För varje punkt måste ett olinjärt ekvationssystem

motsvarande (1)-(2) lösas med Newtons metod. Startvärden bestäms genom att rita upp cirklarna.

Programmet ska inte använda kodupprepning. Använd en `for`-slinga för de tre punkterna. Redovisa koordinaterna samt att Newtons metod konvergerar kvadratisk.

- b) Bestäm det fjärdegradspolynom,  $p(x)$ , som går genom de fem punkterna  $P_0, P_1, P_2, P_3$  och  $P_4$ . Rita upp vägen (graf för polynomet) för  $x \in (0, 1020)$ . Rita även in de fem interpolationspunkterna och markera dem med 'o'. Redovisa koefficienterna i polynomet.
- c) För en annan vägsträcka har koordinaterna i filen `roadcoord.mat` tagits fram. Bestäm det interpolerande polynom som går igenom samtliga koordinater  $(x, y)$ . Plotta koordinaterna och visa det interpolerande polynomet i samma graf. Ser vägen rimlig ut? Förklara vad som händer om du istället använder Matlabs inbyggda funktion `interp1`, med

```
xx = linspace(-1,1,200); v = interp1(x,y,xx); plot(xx,v,'r');
```

- d) Alternativa koordinater för samma vägsträcka återfinns i filen `roadcoord2.mat`. Bestäm det interpolerande polynom som går igenom samtliga koordinater  $(x_2, y_2)$ . Plotta koordinaterna och visa det interpolerande polynomet i samma figur som i uppgift c). Förklara varför vägen framtagen genom interpolation genom  $(x_2, y_2)$  ser annorlunda (bättre?) ut än den från den första delen av uppgift c).

**Skicka in programmen `Road_ab.m` och `Road_bc.m` (för uppgift a-b respektive c-d).**

## 4. Specialfunktioner

Många funktioner som har ett namn går inte att evaluera direkt med hjälp av elementära funktioner (inklusive trigonometriska funktioner och exponentialfunktioner), men de kan definieras med hjälp av en integral. Exempel är Bessel och Hankel funktioner av olika slag.

Ett enkelt exempel på detta är felfunktionen (*the error function*) som definieras som

$$\operatorname{erf}(x) = \int_0^x g(t) dt, \quad g(t) = \frac{2}{\sqrt{\pi}} e^{-t^2}.$$

Vi vill nu studera noggrannheten i att evaluera  $\operatorname{erf}(x)$  genom att approximera integralen med hjälp av trapetsregeln. För att få fram ett noggrant värde att jämföra med kan Matlabs `erf(x)` användas.

Givet ett heltal  $N$ , låt steglängden för trapetsregeln vara  $h = (b-a)/N$ , med punkter uniformt spridda i intervallet  $[a, b]$ , och låt  $T_N[g]$  beteckna approximationen med trapetsregeln

till integralen av  $g$  från  $a$  till  $b$ . Vi vet då att följande gäller

$$E_N[g] = \int_a^b g(t) dt - T_N[g] = -h^2 \frac{b-a}{12} g''(\xi)$$

för något  $\xi \in [a, b]$ , så att en övre gräns av felet vid integration mellan 0 och  $x$  ges av

$$|E_N[g]| \leq \frac{h^2}{12} x \max_{\xi \in [0, x]} |g''(\xi)| = C \frac{x^3}{N^2}, \quad (1)$$

för en konstant  $C$  som kan bestämmas.

#### UPPGIFT 4. INTEGRATION MED TRAPETSREGELN

Skriv en Matlabfunktion som givet  $a$ ,  $b$ ,  $g$  och  $N$  evaluerar  $T_N[g]$ , och använd den där det behövs nedan.

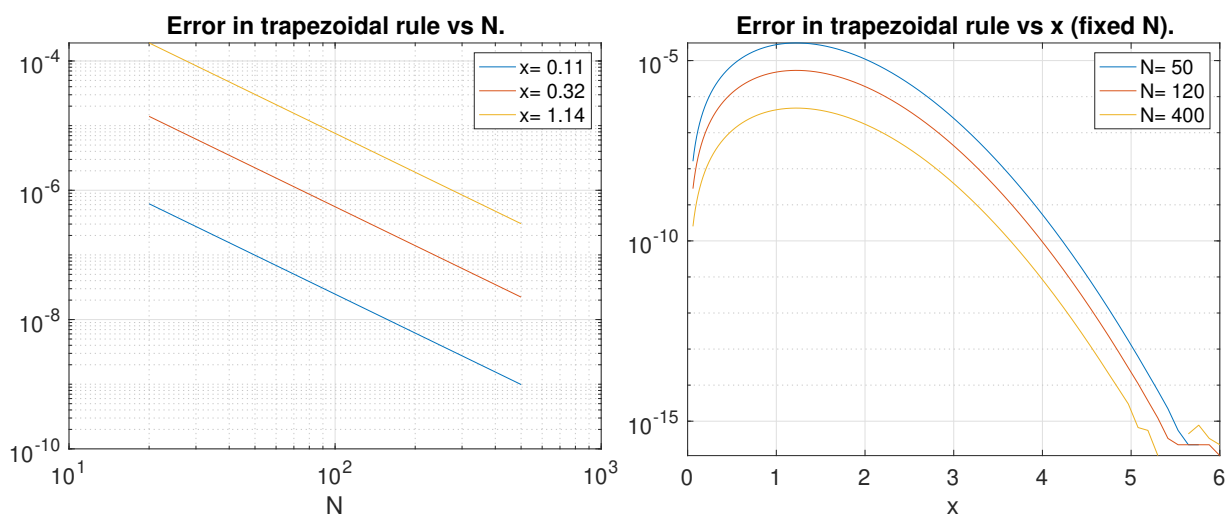
- a) Skriv ett Matlabprogram som med hjälp av funktionen för trapetsregeln återskapar resultaten i figur 3a. Bestäm (för hand)  $C$  i (1), och plotta felgränsen som en funktion av  $N$  i samma plot för samma tre värden på  $x$ , med streckade linjer, så att du kan jämföra med de uppmätta felen. Vad noterar du?
- b) För  $N = 50, 120, 400$ , evaluera nu felet för olika värden på  $x$ , och plotta felet som en funktion av  $x$ . Säkerställ att dina resultat överensstämmer med figur 3 b. När  $x$  ökar från ett litet värde, så ökar felet. Varför blir det så, hur stämmer det med teorin? När sedan  $x$  fortätter att öka så minskar felet igen, och går ner mot avrundningsnivå. Vad händer här, varför blir felet så litet?

*Tips: för en jämn funktion  $g$  så gäller,*

$$\int_0^x g(t) dt = \frac{1}{2} \int_{-x}^x g(t) dt$$

*och trapetsregeln tillämpad på den ena intervalen är lika med trapetsregeln tillämpad på den andra integralen med samma steglängd  $h$ .*

**Skicka in programmet Trapz.m.**



Figur 3: Vänster: Uppmätt fel  $|E_N[g]|$  plottat mot  $N$  för tre olika värden på  $x$ . Höger: Uppmätt fel  $|E_N[g]|$  plottat mot  $x$  för tre olika värden på  $N$ .