



KTH Engineering Sciences

## Projekt-uppgifter

---

1	Instruktioner och tips	2
2	Röret	5
3	Kretsen	7
4	Ljudvågor	10
5	Linjalen	12

---

Uppgifterna är baserade på äldre uppgifter konstruerade av Gerd Eriksson, Henrik Eriksson och Jesper Oppelstrup med flera.

## 1 Instruktioner och tips

### Effektiva metoder

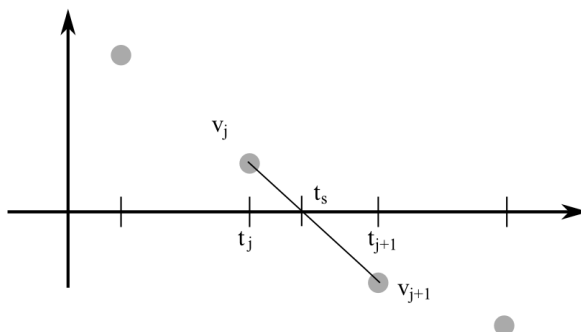
Uppgifterna ska lösas med *effektiva* numeriska metoder i meningen att tidsåtgången för att uppnå en viss noggrannhet inte ska öka för fort när noggrannhetskravet skärps. Mer precist skall metoder för ODE, integration, derivering, interpolation etc. ha en noggrannhetsordning som är minst två, och metoder för ekvationslösning ska ha bättre än linjär konvergensordning. Till effektiva metoder räknas då tex Runge-Kutta 4, trapetsregeln, Simpsons metod, centradifferenser, linjär interpolation, Newtons metod (och modifierad Newton) och sekantmetoden. Till metoder som inte är effektiva räknar vi tex Framåt Euler för ODE, styckvis konstant interpolation (dvs välja det närmaste värdet i en tabell), framåt/bakåt-differenser, generiska fixpunktiterationer, intervallhalvering och linjär sökning efter nollställen eller extrempunkter.

Nedan finns ytterligare kommentarer på vanligt förekommande metoder.

#### *Interpolation av lösningar från ODE-metoder*

När man löser en ODE numeriskt får man approximerade lösningsvärden i diskreta tidpunkter. Flera uppgifter går ut på att avgöra när lösningen antar ett givet värde, vilket typiskt inträffar *mellan* de beräknade tidpunkterna. Det kan tex handla om att beräkna en nedslagsplats för ett kast, eller en maxpunkt. (För maxpunkter söker man punkten där derivatan blir noll; derivatan ingår ofta i ODE-lösningen.)

En typisk situation visas schematiskt i bilden till höger. Värdena  $\{v_n\}$  (grå punkter) är beräknade i tidpunkterna  $\{t_n\}$ , dvs  $v_n \approx v(t_n)$ , och vi ska bestämma vid vilket  $t = t_s$  som den exakta lösningen är noll,  $v(t_s) = 0$ . Vi har hittat det index  $j$  där lösningen byter tecken,  $v_j > 0 > v_{j+1}$ . Här duger det inte att använda den första negativa punkten, dvs  $t_s \approx t_{j+1}$  eller den sista positiva punkten  $t_s \approx t_j$ . Detta ger en för dålig approximation. (Noggrannhetsordningen är ett.)



För att få ett tillräckligt noggrant värde behöver man *interpolera* den numeriska lösningen. Man ska använda åtminstone styckvis linjär interpolation, men det är ofta bättre att använda högre ordning, tex kvadratisk eller kubisk interpolation. Det sökta värdet  $t_s$  approximeras med nollstället till den interpolerande funktionen, som i bilden. Om man vill hitta andra komponenter i ODE-lösningen vid det beräknade  $t_s$ -värdet får man även interpolera dessa på samma sätt som ovan och evaluera interpolanten i  $t = t_s$ .

#### *Modifierad Newton*

I flera av uppgifterna måste man lösa ett icke-linjärt ekvationssystem där man inte på ett enkelt sätt kan använda Newtons metod för system eftersom funktionens derivator inte är direkt tillgängliga. Man kan då använda sig av en *modifierad* Newton-metod, där derivatorna approximeras numeriskt.

Antag att vi har två obekanta  $(x, y)$  och vill lösa  $\mathbf{F}(x, y) = 0$ , där  $\mathbf{F} : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ . Normalt behöver vi beräkna Jakobian-matrisen  $J(x, y)$  för att använda Newtons metod. Nu approximerar

vi istället

$$J(x, y) \approx \tilde{J}(x, y) = \begin{pmatrix} \frac{\mathbf{F}(x+h, y) - \mathbf{F}(x, y)}{h} & \frac{\mathbf{F}(x, y+h) - \mathbf{F}(x, y)}{h} \end{pmatrix} \in \mathbb{R}^{2 \times 2}, \quad h \ll 1,$$

och itererar med  $\tilde{J}$  istället för  $J$ ,

$$\begin{pmatrix} x_{n+1} \\ y_{n+1} \end{pmatrix} = \begin{pmatrix} x_n \\ y_n \end{pmatrix} - \tilde{J}(x_n, y_n)^{-1} \mathbf{F}(x_n, y_n).$$

I varje iteration behöver man alltså evaluera  $\mathbf{F}$  tre gånger istället för bara en gång. Konvergensordningen blir formellt linjär men den asymptotiska konvergenshastigheten är  $O(h)$ , varför konvergensordningen nästan blir kvadratisk när  $h \ll 1$ .

### Linjär sökning

Linjär sökning är ett exempel på en mycket **ineffektiv metod** för ekvationslösning, som **inte är en acceptabel lösningsmetod**. Idén är mycket enkel. Antag att vi har en komplicerad funktion som tar ett argument  $a$ . Vår uppgift är att hitta det  $a$  som gör att funktionen får värdet 20. Vi testar oss först fram och ser att det sökta  $a$ -värdet måste ligga i intervallet  $5 \leq a \leq 10$ , samt att funktionens värde för  $a = 5$  är mindre än 20. Sedan loopar vi:

```
for a=5:0.0001:10
    y = KompliceradFunktion(a);
    if (y>20)
        disp('Rätt värde hittat!')
        disp(a);
        break;
    end
end
```

Detta ger förvisso rätt värde, med noggrannheten  $\pm 0.0001$ , men tar väldigt lång tid. Om noggrannhetskravet skulle ha varit större, säg  $10^{-14}$ , hade man överhuvudtaget inte kunnat hitta lösningen på detta sätt.

I dessa situationer ska man notera att det egentligen är ekvationen  $f(x) = 0$  man vill lösa, där  $f(x) = \text{Kompliceradfunktion}(x) - 20$ . Man löser sedan  $f(x) = 0$  med en metod där derivatan inte behövs, tex sekantmetoden, intervallhalvering, eller modifierad Newton.

### Numerisk tillförlitlighetsbedömning

För alla numeriska svar i uppgifterna ska man göra en *numerisk tillförlitlighetsbedömning*. Det innebär att man uppskattar hur stora de numeriska felen är i svaren för att få en uppfattning om hur noggranna resultaten är. Feluppskattningen ska anges i svaren tillsammans med en kort redovisning av hur uppskattningen gjorts.

Feluppskattningen gäller alltså just de *numeriska* felen, dvs felen i er numeriska approximation jämfört med en exakt lösning av det matematiska problemet. Den gäller inte modellfel, indatafel, etc. Uppskattningen kan göras på olika sätt, tex:

- I metoder där man har en steglängd  $h$  som enda numerisk parameter, som tex i numerisk integration och ODE-lösare, beräknar man sina numeriska värden för både  $h$  och  $h/2$ . Skillnaderna mellan de två svaren ger en god uppskattning av felet i det bästa av värdena, förutsatt att  $h$  är litet. För säkerhets skull bör man undersöka svaren även för  $h/4$ ,  $h/8$ ,

etc. De siffror i svaret som inte ändrar sig när man gör denna steglängdshalvering kan anses pålitliga. Läs mer om detta i föreläsningssanteckningarna om noggrannhetsordning, som finns på hemsidan i Canvas.

- Vid ekvationslösning med iterativa metoder har man oftast en toleransparameter  $\tau$  satt som ingår i avbrottskriteriet för iterationerna. Denna ger direkt en uppskattning av felet i lösningen. För säkerhets skull kan man även här variera  $\tau$  och undersöka hur svaret ändras.
- MATLABs inbyggda funktioner, tex `ode45` och `integral` använder sig också av toleransparametrar. Ni behöver känna till värdena på dessa för att uppskatta felet. Defaultvärdena står angivna i metodernas hjälptexter; gör tex `help ode45`. Man kan också själv sätta toleransparametrarna. Hur man gör detta står i hjälptexterna.
- För sammansatta problem där flera olika numeriska metoder är inblandade kommer man ha flera olika numeriska parametrar som påverkar felet, typiskt en steglängd  $h$  och en toleransparameter  $\tau$ . Man måste då variera *båda* parametrarna som ovan, samtidigt, och undersöka hur lösningen varierar.

Notera att tillförlitlighetsbedömning ska ge ett *kvantitativt* värde på felen. Det duger inte att ge svar av typen "Simpson har noggrannhetsordning 4, vilket är mycket bra, så svaret är noggrant". För steglängdsmetoder är det kritiskt att man jämför åtminstone två olika lösningar för att kunna säga något om felet. En ensam lösning ger ingen information om felet.

## Osäkerhet i modellen

Med osäkerhet i modellen menar vi att (differential)ekvationen i sig innehåller osäkra element, tex parametrar, begynnelsevärden, högerled, etc. Detta ger upphov till en osäkerhet i lösningen. Det har alltså inget att göra med osäkerheten i svaren pga numeriska fel. Osäkerheten finns där även om man skulle lösa problemet exakt!

I de flesta av uppgifterna har man en osäkerhet i ett par parametrar och man ska uppskatta den resulterande osäkerheten i lösningen. Det enklaste sättet att göra detta på är med hjälp av experimentell störningsanalys:

- Beräkna störda lösningar, där parametrarna störs en i taget.
- Summera ihop absolutbeloppen av lösningarnas störningar.

Eftersom man är ute efter osäkerheten i den exakta modellen ska man se till att de numeriska felen är betydligt mindre än de framräknade osäkerheterna. Den numeriska tillförlitlighetsbedömningen ger lämpliga val av steglängd, tolerans, etc. för att uppnå detta.

## 2 Röret

Genom ett tjockvägigt cylindriskt rör strömmar en het vätska med den konstanta temperaturen  $T_i = 350$  [°C]. Cylinderväggen har innerradien 1 längdenhet och ytterradien 2 längdenheter. Temperaturfördelningen  $u(r)$  i röret bestäms av differentialekvationen

$$r \frac{d^2 u}{dr^2} + \frac{du}{dr} = 0.$$

Vid  $r = 1$  har röret samma temperatur som vätskan, vilket ger randvillkoret

$$u = T_i \quad \text{vid } r = 1.$$

Yttersidan vid  $r = 2$  kyls av den omgivande luften som har temperaturen  $T_e = 20$  [C°]. Det medför att värmeflödet  $-k du/dr$  är proportionellt mot temperaturdifferensen  $u - T_e$ , där  $k$  är rörmaterialets värmekonduktivitet. Randvillkoret vid  $r = 2$  blir då

$$k \frac{du}{dr} = -\alpha \cdot (u - T_e) \quad \text{vid } r = 2.$$

Koefficienten  $\alpha$  kallas värmeöverföringstalet mellan röret och luften.

- Antag att  $k = 50$  och  $\alpha = 15$ . Använd finita differensmetoden för att bestämma  $u(r)$ . Dela in intervallet  $1 \leq r \leq 2$  i  $N$  delintervall. Definiera  $r_j = 1 + jh$ ,  $h = 1/N$ , och inför approximationen  $u_j \approx u(r_j)$ . Diskretisera det yttre randvillkoret med en andra ordnings approximation vid  $r_N = 2$ ,

$$\left. \frac{du}{dr} \right|_{r=r_N=2} = \frac{3u_N - 4u_{N-1} + u_{N-2}}{2h} + O(h^2).$$

Visa hur randvärdesproblemet approximeras av ett matrisproblem  $Au = b$  med denna diskretisering.

- Lös problemet först för  $N = 25$ . Gör successiva fördubblingar av  $N$  till önskad precision erhålls — t.ex. fem korrekta siffror i temperaturvärdet vid rörets utsida,  $u(2)$ . Ange detta värde (med tillförlitlighetsbedömning) och rita upp temperaturfördelningen i röret.
- Verifiera att approximationen har noggrannhetsordning två genom att studera hur  $u(2)$ , varierar när  $N$  ökas. Redovisa din metod och resultat.
- Man tillåter inte att rörets utsida blir varmare än  $100$  [C°]. Vi antar att man med hjälp av en bättre kylanordning kan öka värmeöverföringstalet  $\alpha$ . Beräkna det kritiska  $\alpha$ -värdet  $\alpha_{100}$  för vilket temperaturkravet precis uppnås. Använd en effektiv algoritm baserad på tex sekantmetoden eller modifierad Newton för att beräkna  $\alpha_{100}$  med hög noggrannhet.
- Undersök hur känsligt det kritiska  $\alpha$ -värdet är för variationer i vätskans temperatur  $T_i$  och osäkerheter i materialets värmekonduktivitet  $k$ . Antag att dessa värden kan variera med 5%. Vad blir osäkerheten i  $\alpha_{100}$ ?

### Utvidgning

- Istället för att ändra  $\alpha$  ska röret nu kylas genom att lägga på ett värmeisolerande skikt utanpå röret med ett material som har en lägre värmekonduktivitet. Den totala tjockleken,

inklusive skiktet, antas nu vara  $L > 2$ . Värmeöverföringstalet  $\alpha$  hålls konstant  $\alpha = 15$ , men  $k$  varierar i rummet,  $k = k(r)$ , enligt formeln

$$k(r) = k_0 + (k_1 - k_0)\chi_\delta(r - 2), \quad \chi_\delta(r) = \frac{1}{2} + \frac{1}{\pi} \arctan(r/\delta),$$

där  $k_0 = 50$  och  $k_1 = 3$  är rörets respektive isoleringsskiktets värmekonduktivitet. Funktionen  $\chi_\delta$  approximerar en stegfunktion: när  $\delta \ll 1$  är  $k(r)$  approximativt konstant inne i röret ( $=k_0$ ) och i skiktet ( $=k_1$ ), men har en snabb övergång mellan dessa värden vid  $r \approx 2$ . Välj  $\delta = 1/50$ .

Med den variabla koefficienten  $k(r)$  får differentialekvationen formen

$$r \frac{d^2 u}{dr^2} + \left(1 + \frac{rk'(r)}{k(r)}\right) \frac{du}{dr} = 0, \quad 1 < r < L,$$

och randvillkoren

$$u = T_i \text{ vid } r = 1 \quad k(r) \frac{du}{dr} = -\alpha \cdot (u - T_e) \text{ vid } r = L.$$

Modifiera finita differensmetoden från ursprungsproblemet så att det löser detta problem. Beräkna temperaturfördelningen  $u(r)$  i röret och isoleringsskiktet när  $L = 2.3$  (dvs skiktets tjocklek är 0.3). Variera  $N$  så att yttemperaturen  $u(L)$  har ett fel  $< 10^{-2}$ . Ange denna temperatur och rita upp temperaturfördelningen i röret och skiktet.

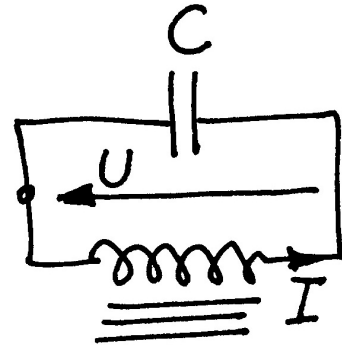
- Bestäm den kritiska skiktjockleken som ger en yttemperatur på precis  $100 \text{ [C}^\circ\text{]}$ . Använd en effektiv algoritm som tidigare.

### 3 Kretsen

En enkel strömkrets består av en kondensator och en spole. Kondensatorn är uppladdad till spänningen  $U_0$ . Vid tiden  $t = 0$  sluts kretsen. Strömmen  $I$  och spänningen  $U$  bestäms sedan av två samband:

$$\text{Spänningen över induktansen:} \quad L \frac{dI}{dt} = U, \quad (1)$$

$$\text{Strömmen genom kondensatorn:} \quad C \frac{dU}{dt} = -I. \quad (2)$$



Spolen innehåller järn och har strömberoende induktans:

$$L = L(I) = L_0 \frac{I_0^2}{I_0^2 + I^2}. \quad (3)$$

Parametrarna har värdena  $L_0 = 0.7$  [H],  $I_0 = 1$  [A],  $C = 0.5 \cdot 10^{-6}$  [F]. Begynnelsevärden vid  $t = 0$  är  $I = 0$  och  $U = U_0$ .

Lösningen  $I(t)$  till differentialekvationen är en periodisk funktion som är mer eller mindre sinusliknande beroende på hur  $U_0$ -värdet väljs.

- Före den numeriska behandlingen kan det vara bra att bedöma storleksordningen på svängningstiden så att lämpligt tidssteg och simuleringsintervall kan väljas. Gör det genom att analytiskt räkna ut frekvensen och svängningstiden för en krets med *konstant*  $L = L_0$ .
- Använd Runge–Kutta 4 för att beräkna strömkurvorna. Några olika värden på  $U_0$  ska provas, dels spänningen 220 V då järnkärnans inflytande är nästan försumbart, dels två höga spänningsvärden 1500 V och 2300 V då strömkurvan inte blir särskilt sinuslik längre. Plotta ett par perioder av de tre lösningarna.
- Den totala energin lagrad i kondensatorn och spolen är konstant, dvs

$$E(t) = E_{\text{kond}}(t) + E_{\text{spole}}(t) = \frac{1}{2}CU(t)^2 + \frac{1}{2}L_0I_0^2 \ln(I_0^2 + I(t)^2) = \text{konstant}.$$

Visa detta genom att derivera uttrycket för  $E$  och utnyttja (1), (2) samt (3). Plotta sedan  $E(t)$  för de olika fallen och verifiera att den är konstant, även för en simulering över många perioder. Prova med olika steglängder.

- Använd interpolation på lämpligt sätt för att från era lösningar bestämma strömmens toppvärde  $I_{\text{max}}$  och svängningens periodtid  $T$  med mycket god precision. Ange dessa för de tre olika värdena på  $U_0$  ovan.
- Frekvensen för kretsens oscillationer beror som ni har sett ovan starkt på spänningsvärdet  $U_0$ . Man vill bestämma vilken spänning  $U_0 = U_0^*$  som ger en frekvens på exakt 400 Hz, dvs  $1/T = 400$ . Beräkna ett noggrant värde på  $U_0^*$  med en effektiv lösningsmetod, tex sekantmetoden eller modifierad Newton. Ange också det motsvarande värdet på strömmens toppvärde  $I_{\text{max}}^*$ .
- Antag att  $L_0$  och  $C$  båda har en osäkerhet på 5%. Vad ger detta för osäkerhet i dina värden på  $U_0^*$  och  $I_{\text{max}}^*$ ?

## Utvidgning

När man ändrar spänningen  $U_0$  ändras inte bara frekvensen utan också formen på strömkurvan. Ni ska här studera formen i mer detalj med hjälp av Fourieranalys. Börja med att definiera den omskalade strömkurvan  $v(t) := I(tT/2\pi)/I_{\max}$ , där  $T$  är periodtiden från tidigare. Funktionen  $v$  kommer per definition ha periodtiden  $2\pi$  och maxvärdet ett. Plotta  $v(t)$  för  $t \in [0, 2\pi]$  och  $U_0 = 220, 1500, 2300$  i samma figur så att man tydligt ser skillnaderna i vågformen. *Fourierutvecklingen* av  $v(t)$  ges av:

$$v(t) = a_1 \sin t + a_2 \sin 2t + a_3 \sin 3t + \cdots \quad a_k = \frac{1}{\pi} \int_0^{2\pi} v(t) \sin(kt) dt.$$

Att det inte blir några cosinustermer i utvecklingen följer av att funktionen  $v$  är udda.

- Använd trapetsregeln för att beräkna de 10 första Fourierkoefficienterna  $\{a_k\}$  med minst fyra siffrors noggrannhet. Om strömmen är nästan sinusformad bör alla koefficienter utom den första vara mycket små. Det symmetriska utseendet hos strömkurvan gör också att Fourierkoefficienterna är noll för jämna  $k$ . Kontrollera detta — stämmer teori och praktik? Plotta i samma figur strömkurvan  $v$  samt resultatet av Fourierutvecklingen, dels då bara de tre första termerna tas med, dels då alla tio finns med. Gör en figur per spänningsvärde  $U_0$ . Visa också koefficienterna i en tabell med en kolumn per  $U_0$ -värde.

Fourierkoefficienternas avtagande med  $k$  ger ofta intressant information om funktionen. Om de avtar fort är funktionen mer sinuslik än om avtagandet är långsamt.<sup>1</sup>

- Plotta  $|a_k|$  som funktion av  $k$  i en **semilogy**-plot, en kurva per  $U_0$ -värde. (Tag bara med udda koefficienter.) Stämmer det att de mer sinuslika formerna har snabbare avtagande?

Svängningskretsar av typen i denna uppgift används ofta (i lite mer avancerad form) för att generera ljud, t.ex. i forna tiders analoga synthesizers. De olika vågformerna ni plottat ovan motsvarar då toner med olika klangbild.

- Använd MATLABs kommando `sound(y,fs)` för att lyssna på ljudet. Vektorn  $y$  innehåller samplevärdena för ljudet. Skapa den genom att repetera den framräknade funktionen  $v$  400 gånger. Parametern `fs` är samplingsfrekvensen. Välj den så att ljudet spelas upp med 400 Hz. (Den kommer bero på vilken steglängd ni använt för att beräkna  $v$ .) Det ger er 1 sekund ljud. Notera hur ljudet förändras när vågformen blir mindre sinuslik, och Fourierkoefficienterna avtar långsammare. Skicka in ljuden för de olika  $U_0$ -värdena tillsammans med rapporten. Ljudfiler skapar ni med kommandot `audiowrite(filnamn,y,fs)`.

Man spelar in en ljudsnutt som svängningskretsen genererat. Samplevärdena för inspelningen (efter omskalning) finns i filerna `mysterysound1.mat` (en period) och `mysterysoundlong.mat` (400 perioder) på hemsidan i Canvas. Detta ljud innehåller även en del brus från inspelningen. Plotta ljudet och lyssna på det.

Kvoten  $a_3/a_1$  är ett mycket enkelt mått på Fourierkoefficienternas avtagande, som här kan användas för att karaktärisera ljudet. Beräkna  $a_3/a_1$  för det inspelade ljudet och jämför det med  $a_3/a_1$  för de simulerade ljuden.

- Bestäm med en effektiv metod vilken spänning  $U_0$  som användes för att generera ljudet. Ange en uppskattning av felet i ert framräknade  $U_0$ -värde.

<sup>1</sup>Man kan tex också visa att om funktionen är deriverbar  $p$  gånger avtar  $a_k$  som  $O(k^{-p})$ .



*Frivilligt:* Testa att generera mer intressanta ljud. Ni kan tex multiplicera ihop strömkurvorna (utan omskalning) från två (eller flera) olika simuleringar med olika, gärna närliggande,  $U_0$ . Resultatet blir en signal där summan och differensen av insignalernas frekvenser kommer att dominera. Detta kallas *ringmodulering* av signalerna och är en klassisk byggsten i synthesizers. Ni kan också skapa ljud direkt från givna vågformer (fyrkant-, sågtand- och triangelvågor är tex lätta att generera), eller från egna val av Fourierkoefficienter; vad händer tex om ni slumpar fram koefficienterna, och/eller låter dem variera (långsamt) mellan de 400 repetitionerna?

## 4 Ljudvågor

Modifierad version av uppgift P8-15 i Kahaner-Moler-Nash.

The speed of sound in ocean water depends on pressure, temperature and salinity, all of which vary with depth in fairly complicated ways. Let  $z$  denote depth in feet under the ocean surface (so that the positive  $z$  axis points down) and let  $c(z)$  denote the speed of sound at depth  $z$ . We shall ignore the changes in sound speed observed in horizontal directions. It is possible to measure  $c(z)$  at discrete values of  $z$ ; typical results can be found in the table. We need  $c(z)$  and also  $c'(z)$  between data points.

$z$	$c(z)$
0	5050
500	4980
1000	4930
1500	4890
2000	4870
2500	4865
3000	4860
3500	4860
4000	4865
5000	4875
6000	4885
7000	4905
8000	4920
9000	4935
10000	4950
11000	4970
12000	4990

- Fit the data in a least-squares sense with the non-linear model function

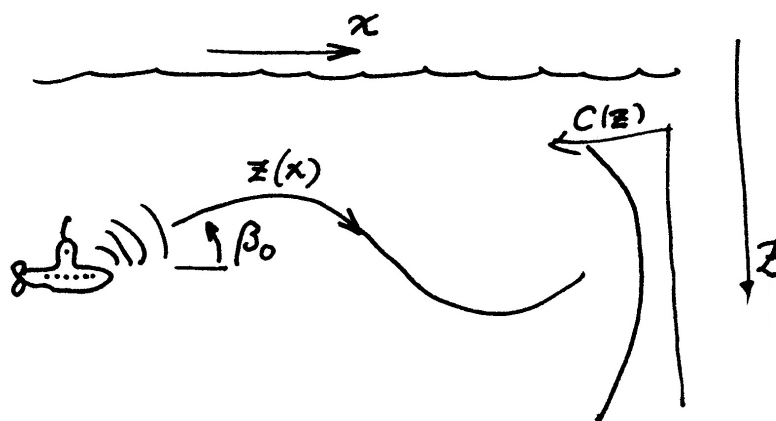
$$c(z) = 4800 + p_1 + p_2 \frac{z}{1000} + p_3 e^{-p_4 z/1000}.$$

Use the Gauss–Newton method to do this. To obtain startguesses, solve the linear least squares problem where  $p_4 = 1$ . Make a plot over the data points and the received model curve  $c(z)$ .

Since the sound speed varies with depth, sound rays will travel in curved paths. A fixed underwater point emits rays in all directions. Given a particular point and initial direction we would like to follow the ray path. Thus letting  $x$  be the horizontal coordinate we know the initial values:  $x = 0$ ,  $z = z_0$ ,  $dz/dx = \tan \beta_0$ , where  $\beta_0$  denotes the angle between the horizontal line  $z = z_0$  and the ray in the start point. The ray path  $z(x)$  is described by the following second order differential equation

$$\frac{d^2 z}{dx^2} = -q_0 \frac{c'(z)}{c(z)^3}, \quad (4)$$

where  $q_0 = (c(z_0)/\cos \beta_0)^2$ . (Note that  $q_0$  depends on the initial data angle  $\beta_0$ .)



- Use the Runge-Kutta method (or ode45) to trace the ray beginning at  $z_0 = 5000$  feet and  $\beta_0 = 13.5$  degrees. Follow the ray for 30 nautical miles (1 nautical mile is 6076 feet). Plot the curve  $z(x)$ . You should find that the depth at  $x_f = 30$  nautical miles is close to 4000 feet.

- Now suppose that a sound source at a depth of 5000 feet transmits to a receiver 30 miles away at a depth of 4000 feet. The above calculation shows that one of the rays from the source to the receiver leaves the source at an angle close to 13.5 degrees. Because of the nonlinearity of the equation there may be other rays leaving at different angles that reach the same receiver. Run your program for  $\beta_0$  in the range from  $-15$  up to  $20$  degrees. Plot the ray paths for each angle, on top of each other. Plot in a separate figure the end depth  $z(x_f)$  as a function of initial angle  $\beta_0$ .
- We are interested in finding values of  $\beta_0$  for which  $z(x_f) = 4000$ . Use an efficient algorithm, like the secant method or modified Newton, to determine the rays which pass through the receiver. Compute the angle values with high numerical accuracy.
- The function  $c(x)$  is clearly not given with high certainty. Model this by assuming that your coefficients  $p_1, \dots, p_4$  all have a 1% uncertainty (after the least squares fit used above). Visualize the uncertainty in  $c(x)$  by plotting a couple of instances of  $c(x)$ , where the  $p$  coefficients are perturbed by 1%, on top of each other, together with the data points. Estimate the resulting uncertainty in  $\beta_0$ .

## Utvidgning

To compute the time it takes for sound to propagate along a ray we add another ODE for the time  $T(x)$ ,

$$\frac{dT}{dx} = \frac{\sqrt{q_0}}{c(z)^2}.$$

Note that this ODE is coupled to the one for  $z$  in (4).

- Compute the time it takes for a sound pulse to travel from the source to the receiver along the different rays you found above. A listener at the receiver will hear the same sound pulse arrive several times with some distinct intervals between. Sort the travel times and determine the length of these intervals. Plot the different rays and indicate along which ray the pulse arrives first, second, etc.
- At a listening station for the receiver an operator detects a noise from a submarine. The noise is repeated four times. The interval between the first two noises is 0.382 seconds. Then there is another 0.0077 seconds delay till the next one. The last signal is lost. From this data you can determine the location of the submarine. This is your task. As a help, the submarine is not too far from the source used above.

This problem boils down to solving a nonlinear system of equations. The unknowns in the equations are the  $x$ - and  $z$ -coordinates of the submarine, i.e. the initial point of the rays. Given any such pair, you can use your solution methods above to compute the time interval  $\Delta t_1$  between the first and second ray hitting the receiver, and  $\Delta t_2$  between the second and third ray. In this way, a vector valued function  $\mathbf{F}(x, z)$  is defined,

$$\mathbf{F}(x, z) := \begin{pmatrix} \Delta t_1 \\ \Delta t_2 \end{pmatrix}.$$

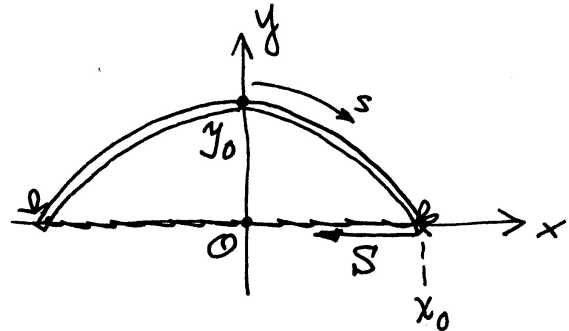
(Note that for each  $(x, z)$  you first need to find the four correct angles that produce rays hitting the receiver.) Next, you use the modified Newton's method to solve

$$\mathbf{F}(x, z) := \begin{pmatrix} 0.382 \\ 0.0077 \end{pmatrix}.$$

As initial guess, use the location of the source  $(x_0, z_0) = (0, 4000)$ .

## 5 Linjalen

Busiga Bertil upptäcker att skolsalens linjal är så böjlig att den borde duga till pilbåge. Han spänner ett snöre mellan linjaländarna — det råkar finnas små hål just vid nollstrecket och vid en-metersmarkeringen — och drar åt snöret så att bågen buktar ut vid mittpunkten som på figuren nedan. Uppgiften är att bestämma formen på pilbågen och kraften  $S$  i bågsträngen när utbuktningen är  $y_0 = 30$  cm och linjalens längd är  $L = 1$  m.



Formen blir symmetrisk kring mittpunkten så det räcker att behandla högra halvan av intervallet  $-x_0 \leq x \leq x_0$ . Låt  $\mathbf{r}(s) = (x(s), y(s))$  vara en båglängdsparameterisering av linjalen där  $\mathbf{r}(0)$  är dess mittpunkt. Funktionerna  $x(s)$  och  $y(s)$  uppfyller då differentialekvationerna<sup>2</sup>

$$\frac{d^2y}{ds^2} + qy\sqrt{1 - \left(\frac{dy}{ds}\right)^2} = 0, \quad \frac{dx}{ds} = \sqrt{1 - \left(\frac{dy}{ds}\right)^2},$$

med randvillkoren

$$y(0) = y_0, \quad y'(0) = 0, \quad y(L/2) = 0, \quad x(0) = 0.$$

Parametern  $q$  i differentialekvationen är relaterad till den sökta kraften enligt  $q = S/EI$ , där  $E$  och  $I$  är linjalens elasticitetsmodul respektive yttröghetsmoment. Vi antar att linjalen är gjord av lönn med  $E = 7.6 \cdot 10^9$  [N/m<sup>2</sup>] och att dess tvärsnitt är rektangulärt med bredden  $b = 3$  cm och tjockleken  $h = 3$  mm, vilket ger  $I = bh^3/12 = 6.75 \cdot 10^{-11}$  [m<sup>4</sup>].

Det blir fråga om ett ickeinjärt randvärdesproblem med den okända parametern  $q$ . Inskjutningsmetoden ska användas för att iterativt hitta rätt värde på  $q$ . Då  $q$  är bestämt får vi den korrekta pilbågsformen  $\mathbf{r}(s)$  samt kraften  $S = qEI$ .

- Goda startvärden behövs för att  $q$  ska kunna bestämmas iterativt. Börja därför med att lösa det förenklade problemet då  $y'$ -termen försummas, för att sedan kunna utnyttja resultatet som startgissning till det ickeinjära problemet. För små utböjningar kommer  $y' \approx 0$ . Den andra ekvationen ger då att  $x' \approx 1$ , vilket insatt i den första ger den förenklade differentialekvationen  $y'' + qy = 0$ , med samma randvillkoren som ovan. Visa att lösningen till denna kan skrivas  $y(s) = y_0 \cos(\sqrt{q}s)$ , och använd detta för att bestämma  $q$ . (Notera att detta approximativa värde är oberoende av  $y_0$ .)
- Implementera sedan inskjutningsmetoden. Skriv om differentialekvationerna till ett system av tre första ordningens ODE, inklusive begynnelsevärden. Implementera en funktion som med hjälp av Runge–Kutta 4 löser begynnelsevärdesproblemet med de tre villkoren vid  $s = 0$ , för ett givet  $q$ . Använd startgissningen på  $q$  från förra deluppgiften, och en effektiv iterativ metod baserad på sekantmetoden eller modifierad Newton för att hitta det värde på  $q$  som ger  $y(L/2) = 0$ .

- Rita upp pilbågen!
- Hur stor blir kraften  $S$  i bågsträngen?
- Vad blir bågsträngens längd  $\ell$ ?
- Flera parametrar är osäkra i modellen. Antag att  $y_0 = 0.30 \pm 0.01$ ,  $L = 1.00 \pm 0.01$  och  $E = 7.6 \cdot 10^9 \pm 0.3 \cdot 10^9$ . Hur stor blir då osäkerheten i  $S$  och  $\ell$ ?

<sup>2</sup>Så länge  $x'(s) > 0$ .

## Utvidgning

Vi vill nu bestämma hur mycket linjalen kan böjas innan den knäcks och går sönder. Villkoret för detta är att den maximala böjspänningen i linjalen

$$\sigma_{\max} = \frac{y_0 S h}{2I} = \frac{1}{2} y_0 E q h, \quad (5)$$

överstiger materialets *modulus of rupture* (MOR), vilket för lönn ligger på  $5.1 \cdot 10^7$  [N/m<sup>2</sup>]. Det kritiska värdet för  $y_0$  där linjalen knäcks betecknar vi  $y_0^*$  och motsvarande  $q$  kallar vi  $q^*$ . Ekvation (5) ger ett samband mellan  $q^*$  och  $y_0^*$ , som ni ska inkorporera i lösningsalgoritmen ovan.

- Hur stora blir  $y_0^*$  och  $q^*$ ?
- Rita upp pilbågen vid detta fall i samma figur som pilbågen ovan. (Använd `hold on` och `axis equal`.)
- Hur stor blir kraften  $S$  i bågsträngen?

Betrakta nu  $y_0^*$  som en funktion av linjalens längd  $L$  och höjd  $h$ , dvs  $y_0^* = y_0^*(L, h)$ . Fixera  $h = h_0 = 3$  mm och beräkna  $y_0^*(L, h_0)$  för ett antal olika linjal-längder  $L$  i intervallet  $[0.1, 1]$ . Anpassa polynomet  $P(x) = c_1 x + c_2 x^2$  till värdena i minstakvadratmening, så att  $y_0^*(L, h_0) \approx P(L)$ . Använd detta polynom tillsammans med sambandet

$$y_0^*(L, h) = y_0^*\left(\frac{Lh_0}{h}, h_0\right) \frac{h}{h_0} \approx P\left(\frac{Lh_0}{h}\right) \frac{h}{h_0},$$

för att ställa upp en ingenjörsmässig tumregel: en enkel formel,

$$y_0^*(L, h) \approx d_1 L + d_2 \frac{L^2}{h},$$

för hur mycket en linjal av längden  $L$  och tjockleken  $h$  kan bukta ut innan den knäcks. Ange  $d_1$  och  $d_2$ . Lägg gärna till en felgräns/säkerhetsfaktor till formeln. Bättre, men mer komplicerade formler ges genom att använda högre ordnings polynom.