System och transformer

Datorlaboration 2

av Sven Spanne

Reviderad ht 2012

av Jan Gustavsson och Tomas Carnstam

LUNDS UNIVERSITET
Lunds Tekniska Högskola

Inledning

Programmet för denna datorövning är studium av insignal-utsignalrelationer, dels i tidsområdet, dels i frekvensområdet. Därefter följer en uppgift samt en kort introduktion till Fouriertransformering med Maple. Ha läroboken och övningshäftet tillgängliga. Tag också med din Matlabhandledning.

Förbered dig genom att titta igenom denna handledning och anvisningarna nedan. Läs igenom kapitel 9, speciellt definitionerna av linearitet, tidsinvarians och kausalitet samt systembeskrivning med impulssvar. Läs också om frekvens- och amplitudfunktioner i kapitel 12. En första kontakt (bortsett från föreläsningarna) med kapitel 13 får du också under laborationen.

Tag med teorikompendium, övningshäfte och utdelad transformtabell!

System på insignal-utsignalform

Vi skall experimentellt undersöka system på insignal-utsignalform med hjälp av Matlab. I en uppgift kommer Maple att användas.

2.1 En del behövliga färdigskrivna filer finns att hämta på kursens hemsida.

`http://www.maths.lth.se/matematiklth/personal/jang/sot2012/sot2012lp2.html`

Filerna finns samlade i två filer `Packat.zip` för Windows och `Packat.tar` för Linux. Hämta lämpligt paket och packa upp i någon lämplig egen Matlabkatalog.

Allmänt

Vi skall använda ett paket Matlabfiler för att kunna räkna med funktioner (in- och utsignaler) och system. Detta paket är specialskrivet för övningen och har många begränsningar. Matlab kan endast göra tidsdiskreta räkningar, så att de tidskontinuerliga systemen approximeras genom sampling. Vidare kan Matlab ej visa upp hela tidsaxeln utan signalerna skärs av utanför ett visst tidsintervall ($-10 \leq t \leq 10$).

I paketet representeras funktioner (signaler) med Matlabföljder (radvektorer) och system med Matlabfunktioner (`m`-filer). Titta gärna på de `m`-filer ni kör, med kommandot `type`, och försök förstå hur de fungerar.

2.2 Starta Matlab och initiera övningspaketet med kommandot `startlab2`. Med `whos` får du reda på vilka variabler du har och hur stora de är. Rita upp funktionen `theta` med `plot(t,theta)` och gör motsvarande med `delta` (oändligheten är här tydligen ungefär 13).

Du kan själv definiera nya funktioner som passar in i systemet. Stegfunktionen är t ex tillverkad med `(t>0)`. En kausalt avskuren sinusfunktion fås med `sin(t).*(t>0)` etc.

2.3 Definiera för framtida bruk några funktioner, en kausal rektangelpuls, en triangelpuls och en kausal cosinusfunktion:

```
krekt = (t>0)-(t>1);
triangel = (t+1).*(t>-1)-2*t.*(t>0)+(t-1).*(t>1);
kcos = cos(t).*(t>0);
```

Rita upp funktionerna med `plot`.

Matlabtips: Du kommer snart att få upp många figurfönster från Matlab. När du har fått för många så går det lätt att sudda t ex alla fönster utom de två sista, med kommandot `delete(1:gcf-2)` (`gcf` = get handle to current figure = numret på aktuellt figurfönster).

System

Ett system (i insignal-utsignalform) $\mathcal{S}$ representeras med en Matlabfunktion, t ex definierad i `syst1.m`. Om insignalen finns t ex i Matlabföljden `theta` så beräknas utsignalföljden `y` genom `y=syst1(theta)`.

2.4 Genom filen `lpfilter.m` simuleras lågpassfiltret i kursboken (exempel 10.7 sidan 195) (med $R = 1$ och $C = 1$). Bestäm stegsvaret för filtret med kommandot `y=lpfilter(theta);` och rita upp det med `plot(t,y)`.

För att förenkla för övaren finns i paketet en Matlabfunktion `plotsyst` som ritar upp insignalen och utsignalen. Se nästa uppgift.

2.5 Prova kommandot

```
plotsyst('lpfilter',theta);
```

Observera att kommandot tar systemnamnet som en *sträng*, alltså omgivet av `' '`

2.6 Filen `bil.m` innehåller en realisering av den enhjuliga bilen i exempel 9.10 i läroboken. Testa vad som händer om den kör på en trottoarkant (beskriven med en stegfunktion): `plotsyst('bil',theta);` Jämför övning 11.30 e).

2.7 Filen `brygga.m` innehåller en realisering av nätet i övning 12.23. Rita stegsvaret: `plotsyst('brygga',theta);` Pröva också med `theta` utbytt mot `krekt`, `triangel`, `kcos` och `delta`.

Tror du att det är enkelt att t ex för hand ställa in insignalen så att utsignalen antar ett önskat värde? Svar (motivera): ______________________________

Speciella system

I paketet finns några färdiga praktiskt viktiga speciella system.

- fördröjning: en fördröjning tiden a fås med `utsignal=delay(insignal,a);` (exempel 9.19)
- derivation: `utsignal=differentiator(insignal);` (exempel 9.11,22)
- integration: `utsignal=integrator(insignal);` (sats 10.2)

I paketet kan man även fördröja signaler med kommandot `delay`.

2.8 Titta till exempel på

```
plot(t,delay(triangel,2));
```

Byt sedan 2 mot -3.

Gör sedan om fördröjningarna ovan med `triangel` bytt mot `kcos`.

Vid fördröjningen skiftas in nollor från det förflutna (eller framtiden, om fördröjningen inte är kausal). Detta är en ofullkomlighet som vi skulle kunna undvika om det fanns oändligt mycket minne i datorn. Då skulle hela funktionen kunna lagras och inte bara ett ändligt delintervall.

Prova nu att derivera några enkla funktioner.

2.9 Derivera triangelpulsen, rektangelpulsen, stegfunktionen, deltafunktionen och den avskurna cosinusen.

```
plotsyst('differentiator',triangel);
plotsyst('differentiator',krekt);
```

osv. Lägg märke till spikarna i en del av derivatorna. Hur kan man representera dem matematiskt? Svar: ______________________________

Egenskaper hos system

De fyra viktigaste speciella egenskaperna hos system på insignal-utsignalform är *linearitet*, *tidsinvarians*, *kausalitet* och *stabilitet*. Vi skall nu experimentellt undersöka vilka av våra system som har de tre första av dessa egenskaper. Stabiliteten är svårare att se med vårt system, eftersom den talar om vad som sker då $t \to \infty$, och det får dåligt plats på skärmen.

Linearitet

Ett system $\mathcal{S}$ kallas ju lineärt om

$$\mathcal{S}(c_1w_1 + c_2w_2) = c_1\mathcal{S}(w_1) + c_2\mathcal{S}(w_2)$$

för alla tal c_1, c_2 och alla insignaler w_1, w_2. Funktionen `lintest` testar detta genom att beräkna och rita upp

$$\mathcal{S}(c_1w_1 + c_2w_2) - (c_1\mathcal{S}(w_1) + c_2\mathcal{S}(w_2)).$$

2.10 Stäng inte av Matlab utan starta parallellt `Maple 16`. Hämta filen `uppgift210.mws` från kursens hemsida. Öppna filen i Maple. Kör filen kommandovis genom att placera markören på en Maplerad i taget och sedan trycka på return-knappen. Använd resultaten för att avgöra vilka av systemen i övning 9.3 som är lineära

2.11 Åter till Matlab! Undersök om `lpfilter` kan vara *lineärt* genom att stoppa in olika kombinationer av insignaler. Börja t ex med

```
lintest('lpfilter',triangel,kcos,-3,4)
```

Lägg märke till *skalan* i den undre figuren. Verkar systemet vara lineärt? ________

Anm: Man kan givetvis inte testa *alla* tänkbara insignaler, men det brukar märkas snabbt om ett system inte är lineärt.

Tidsinvarians

Ett system är *tidsinvariant* om fördröjningar av insignalen medför motsvarande fördröjningar av utsignalen men inga andra förändringar av denna. Detta är lätt att testa i vårt system.

2.12 Använd `plotsyst` på systemet `lpfilter` och insignalen `triangel`. Förskjut sedan insignalen åt höger, t ex $a = 2$, och åt vänster, t ex $a = -3$ och jämför utsignalerna:

```
plotsyst('lpfilter',triangel);
plotsyst('lpfilter',delay(triangel,2));
plotsyst('lpfilter',delay(triangel,-3));
```

Det finns även här en funktion som underlättar testen, `delaytest`.

2.13
```
delaytest('lpfilter',triangel,2);
delaytest('lpfilter',triangel,-3);
```

Lägg märke till *skalan* i den undre figuren.

I vissa fall kan testet störas av nollorna som skiftas in från höger eller vänster. Bortse därför från eventuella störningar i början eller slutet.

Kausalitet

Om för varje a gäller

$$\text{insignalen } w(t) = 0 \quad \text{då } t < a \Longrightarrow \text{utsignalen } y(t) = 0 \quad \text{då } t < a$$

så är systemet *kausalt* - om det är lineärt. (För olineära system måste man använda lärobokens definition. Se läroboken!)

2.14 Undersök om systemet `lpfilter` är kausalt genom att sända in några triangelpulser med olika fördröjningar. Titta noga på skalorna! Kausalt? ________________

Ett enklare kriterium för kausalitet, som kan användas då systemet är lineärt och tidsinvariant, är att impulssvaret är en kausal funktion.

Undersökning av system

2.15 Undersök om systemen `syst1-6` är lineära, tidsinvarianta och kausala. (Om ett systen inte är lineärt kan du hoppa över kausalitetsundersökningen.) Fyll i följande tabell:

system	lineärt	tidsinvariant	kausalt
syst1			
syst2			
syst3			
syst4			
syst5			
syst6			

Börja med lineariteten. Använd `lintest` som i 2.11. Fyll sedan i den andra kolumnen. Använd `delaytest` som i 2.13.

För de system, som är lineära och tidsinvarianta, kan kausaliteten testas med en enda insignal. Vilken? Se sats 9.2! Svar: ______________ De andra kräver lite mer omsorg.

System 3 bör alla känna igen från övningarna. Se efter vad det gör med en puls `krekt` och med steget `theta`. Titta sedan på övning 9.10.

Impulssvar

Impulssvaret $h = \mathcal{S}\delta$ till ett system fås som utsignalen då insignalen är deltafunktionen. Då impulssvaret är känt kan sedan utsignalen beräknas för vissa typer av system.

2.16 Låt `system` vara något av systemen i 2.15 och låt `h` vara motsvarande impulssvar. Låt `w` beteckna någon av de testsignaler vi arbetar med i denna labb. I denna uppgift ska du undersöka för vilka system i 2.15 som

`falt(h,w) = utsignalen i plotsyst('system',w).`

Gör som i följande exempel:

```
close all
plotsyst('syst1',kcos);
h = syst1(delta);
y = falt(h,kcos);
subplot(2,1,1,'replace')
plot(t,y);
xlabel('faltningssignalen')
```

och jämför. Den verkliga utsignalen visas i det undre fönstret medan faltningssignalen $h * w$ syns i det övre. För att minska ditt "knappande" kan du använda scriptet test216.m. Editera filen och ändra på två ställen `syst1` till annat system. Även testsignalen `kcos` kan du byta ut om du har tid och lust. Spara efter ändring!

Vilka egenskap har de system där

`falt(h,w) = utsignalen i plotsyst('system',w):` ____________________

Amplitudfunktioner

För de lineära och tidsinvarianta av våra system är överförings- och frekvensfunktion definierade. Frekvensfunktionens absolutbelopp $A(\omega) = |H(i\omega)|$ beskriver ju amplitudförstärkningen vid vinkelfrekvensen ω. Vi skall testa detta. Skriptet `amplitudf` beräknar amplitudfunktionen $A(\omega)$ om impulssvaret $h(t)$ är givet. ($H(i\omega)$ är ju Fouriertransformen av $h(t)$. Se sats 12.6 i kursboken.) Den bakomliggande metoden är FFT (snabb Fouriertransformation).

2.17 Beräkna amplitudfunktionen för systemen `lpfilter` och `bil`.
Börja med att sända in cosinus- eller sinussignaler med vinkelfrekvenser 1, 2, 3 och 6, t ex `plotsyst('bil',sin(3*t));`.

1. Uppskatta i den erhållna figuren hur stor amplitudförstärkningen är (insignalens amplitud bör vara 1 så utsignalens amplitud är lika med förstärkningen). Fyll i kolumn 3 tabellen

System	**Vinkelfrekvens**	**Ampl.förstärkning**	**Ampl.förstärkning**
`lpfilter`	1		
	2		
	3		
	6		
`bil`	1		
	2		
	3		
	6		

2. Jämför det erhållna värdet med motsvarande punkt på kurvan för amplitudfunktionen. Skriv

   `amplitudf(lpfilter(delta));` respektive `amplitudf(bil(delta));`

   Avläs amplitudförstärkningen för vinkelfrekvenserna i tabellen ovan och skriv in värdena i kolumn 4.

2.18 (I mån av tid.) Lös följande tentamensuppgift (nr 5, 2008-03-25).
Tre system $\mathcal{S}_1, \mathcal{S}_2$ och $\mathcal{S}_3$ definieras av

- $(\mathcal{S}_1 w)(t) = w'(t) + w(t+1)$
- $(\mathcal{S}_2 w)(t) = \displaystyle\int_{-\infty}^{\infty} e^{-(t-\tau)^2} w(\tau)\, d\tau$
- $(\mathcal{S}_3 w)(t) = w(t^2)$.

Alla systemen antas definierade för alla reellvärda och deriverbara funktioner $w(t)$, där $t \in \mathbb{R}$. För systemet $\mathcal{S}_2$ ska dessutom integralen vara konvergent.

Ange de system som är lineära, de som är tidsinvarianta och de som är kausala. Ge svaret genom att skriva av och fylla i tabellen

System	Lineärt	Tidsinvariant	Kausalt
$\mathcal{S}_1$			
$\mathcal{S}_2$			
$\mathcal{S}_3$			

med ja eller nej i de nio tomma rutorna. Endast svar behöver ges. För högst fyra rätt ges ingen poäng. Därefter ger varje rätt svar 0.2 poäng.

Tips: För att studera linearitet och tidsinvarians kan du bygga vidare på Maple-filen Uppgift210.mws använd i uppgift 2.10. Vid studium av $\mathcal{S}_2$ kan det underlätta med ett variabelbyte $\tau = t - x$ via Maple-kommandona `with(IntegrationTools)` och `Change`.

Kausaliteten kan du klara av genom att rita insignalen $w(t)$ och utsignalen $(\mathcal{S}w)(t)$ t ex då $w(t) = (t-2)\theta(t-2)$. Använd gärna `thickness = 3` i plotkommandot.

Fouriertransformation

Matlab

Matlab kan utföra numeriska diskreta Fouriertransformationer. Tyvärr är det en hel del bokföringsmässiga detaljer som måste klaras av för att man skall kunna beräkna kontinuerliga Fouriertransformer med Matlab, så tiden tillåter inte att vi går igenom det här. Kommandot för den direkta transformen är `fft` och för den inversa `ifft`.

Maple

Börja med kommandot `restart` för att radera gamla variabler från 2.10.

Maple har inbyggda rutiner för Fouriertransformation och invers Fouriertransformation. De heter `fourier` och `invfourier` och blir tillgängliga med `with(inttrans)`. Maple känner även till deltafunktionen $\delta(t)$, som fås med `Dirac(t)`, och stegfunktionen $\theta(t)$, som fås med `Heaviside(t)`, och kan ofta räkna rätt med dem. Om du föredrar samma namn som i kursboken går det att åstadkomma med `alias(delta=Dirac, theta=Heaviside);`.

2.19 Beräkna derivatan av $\theta(t)$ och $e^{-t}\theta(t)$ med hjälp av Maple (använd `diff`). Använd `simplify` på det senare resultatet så ser du att Maple kan regeln $f(t)\delta(t-a) = f(a)\delta(t-a)$.

2.20 Derivatan av $\delta(t)$, alltså $\delta'(t)$ betecknas i Maple med `Dirac(1,t)` (och vid högre derivator ersätts 1 med derivationsordningen). Låt Maple förenkla uttrycket $\sin(t)\delta'(t-2)$ och jämför med vad du själv fått det till.

2.21 Beräkna Fouriertransformen av $e^{-t}\theta(t)$ med Maple:

```
fourier(exp(-t)*Heaviside(t),t,omega);
```

Stämmer Maples resultat med det du kan avläsa i din tabell? Svar: __________

2.22 Försök beräkna Fouriertransformen av $e^{t}\theta(t)$ med Maple. Vad händer? Varför?

____

2.23 Även om egenskaperna hos en ospecifierad funktion `f` inte är kända av Maple förutsätts att `f` går att Fouriertransformera. Exempelvis blir `fourier(fourier(f(t),t,omega),omega,t);` vad man förväntar sig. Vad förväntar du dig enligt övning 13.17? ____________________
Säg att `falt:=int(f(t-tau)*g(tau),tau=-infinity.. infinity);`. Titta på sats 13.5. Vad tror du att `fourier(falt,t,omega);` blir?____________________

2.24 Lös övning 13.1b på följande sätt:

```
fourier(exp(t)*Heaviside(-t),t,omega);
```

och exempel 13.3 i läroboken,

```
fourier(exp(-abs(t)),t,omega);
```

2.25 Lös exempel 13.4 (transformen av e^{-t^2}) med Maple.

2.26 Lös övning 13.6 med Maple.

2.27 Lös övning 13.26 med Maple. Duger följande?

```
intekv:=y(t) +
int(exp(-abs(t-u))*y(u),u=-infinity..infinity)=t*exp(-abs(t));
fourier(intekv,t,omega);
subs(fourier(y(t),t,omega)=Y,%);
losning:=solve(%,Y);
invfourier(losning,omega,t);
```

Det blir lättare att jämföra Maples lösning av 13.26 med facit i övningshäftet om du studerar fallen $t > 0$ och $t < 0$ var för sig.

2.28 Lös övning 13.32 med Maple. Parsevals formel hittar du på första sidan i formelsamlingen. Kan Maple beräkna integralen direkt utan Parseval?

Laplacetransformation

Detta avsnitt är inte avsett att gås igenom under datorövningen. Gör det gärna på egen hand när du lärt dig Laplacetransformationen.

Maple kan räkna med *ensidiga* Laplacetransformer. Efter att ha gett kommandot `with(inttrans);` kan man använda `laplace` och `invlaplace`. Syntaxen är densamma som vid Fouriertransformationen.

2.29 Beräkna Laplacetransformen i övning 14.1b. Den fås med

```
laplace(exp(-2*t)*Heaviside(t),t,s);
```

Kontrollera resultatet med

```
invlaplace(%,s,t);
```

Problemen vid Laplacetransformation uppkommer i regel vid inverstransformationen.

2.30 Lös några deluppgifter i 14.15.

En av de metoder som Maple använder i kommandot `dsolve` för lösning av differentialekvationer är i själva verket Laplacetransformation.

För att visa Maples möjligheter så går vi igenom en lösning av integralekvationen i övning 14.54.

2.31 Finn en lösning till integralekvationen

$$\int_0^x f(x-y)f(y)\,dy = x^3e^x, \quad x > 0.$$

Det kan man göra så här:

```
intekv := int(f(x-y)*f(y),y=0..x)=x^3*exp(x);
laplace(intekv,x,s);
```

Vi byter ut Laplaceuttrycket mot något enklare, löser ut detta och väljer en lösning att inverstransformera.

```
subs(laplace(f(x),x,s)=U,%);
losningar := solve(%,U);
F := losningar[1];
invlaplace(F,s,x);
```

Med `F:=losningar[2]` hade vi fått den negativa lösningen.

Lösningsmetoden ovan innehåller några moment som kanske är obekanta. Använd Mapledokumentationen för att få reda på vad okända kommandon gör.

Därmed avslutar vi datorövningarna. Jag hoppas att ni kommer att hålla det ni lärt er om Matlab, Maple och matematik vid liv och att ni kan bygga på det när det behövs för er ingenjörsverksamhet.

Litteraturförteckning

Litteraturförteckning

Pärt-Enander, E. & Sjöberg, A. (2001), *Användarhandledning för Matlab 6*, Institutionen för informationsteknologi, Uppsala Universitet, Uppsala.

Heck, A. (2003), *Introduction to Maple*, Springer.

Spanne, S. (1997), *Lineära system*, KFS.

Werner, C.G. (2002), *Maplehandboken*, Matematikavdelningen (NF), Lund.