Lösningar till tentamen i Industriell reglerteknik TSRT07

Tentamensdatum: 2023-06-08

Martin Enqvist

1. (a) Genom att rita upp nyquistkurvan för kretsförstärkningen i det tidskontinuerliga och det tidsdiskreta fallet kan man beräkna amplitudmarginalerna. Det visar sig att stabilitetsgränsen är 3.41 när man använder en tidskontinuerlig PI-regulator och 3.06 i det tidsdiskreta fallet. Matlab-kod:

```
F=tf([3 1],[3 0]);
G=tf(1,[2 1],'InputDelay',1);
nyquist(F*G)

Fd=c2d(F,0.3,'tustin');
Gd=c2d(G,0.3);
nyquist(Fd*Gd)
```

(b) Enligt Ziegler-Nichols inställningsregel ska förstärkningen i en P-regulator vara halva den kritiska förstärkningen. I detta fall får man $K = 1.5$ när $z(t) = 0$ och $K = 3$ när $z(t) = 1$. Den parameterstyrda P-regulatorns förstärkning blir, med hjälp av linjärinterpolation,

$$K(z(t)) = 1.5(1 - z(t)) + 3z(t) = 1.5 + 1.5z(t).$$

(c) Nej, eftersom styrsignalen är begränsad kommer det initiala MPC-problemet inte att vara lösbart för alla initialtillstånd. Däremot kommer metoden att fungera för vissa begränsade initialtillstånd. (Det aktuella systemets poler kan beräknas med kommandot

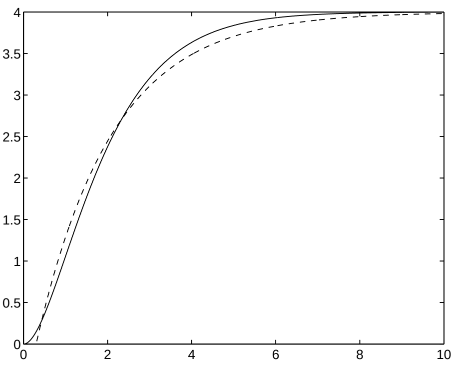
```
eig([-0.5 1.5;0.2 -0.5])
```

Polerna är 0.0477 och -1.0477 och eftersom en av polerna ligger utanför enhetscirkeln är systemet instabilt. Ingen regulator med en begränsad styrsignal kan därför stabilisera systemet för godtyckliga initialtillstånd.)

2. (a) Genom att rita upp stegsvaret för systemet kan man bestämma modellparametrarna L, K_p och T enligt följande principer.

   * Dödtiden L är den tid under vilken stegsvaret (ungefär) är lika med noll.
   * Den statiska förstärkningen K_p är det värde som stegsvaret närmar sig när tiden går mot oändligheten.
   * Tidskonstanten T väljs så att systemets och modellens stegsvar båda (ungefär) har värdet $0.63K_p$ vid tidpunkten $L + T$.

Systemets och modellens stegsvar visas i figur 1.



Figur 1: Stegsvaret för systemet (heldraget) och treparametermodellen (streckat) i uppgift 2(a).

Treparametermodellen blir

$$\hat{G}(s) = \frac{4.0}{1.8s + 1} e^{-0.30s},$$

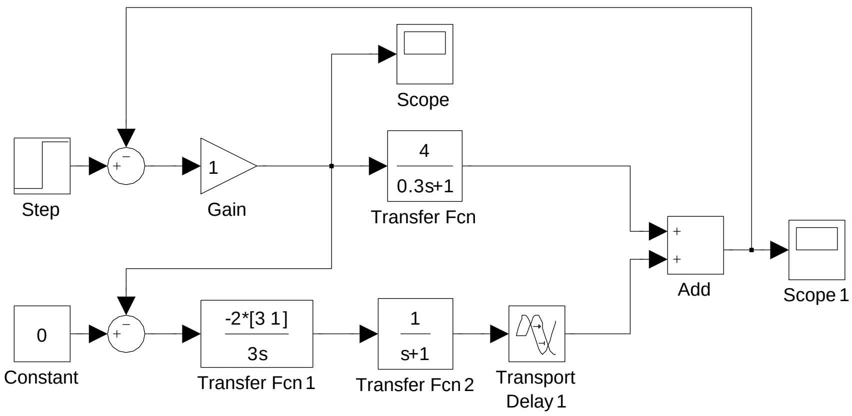
vilket ger regulatorparametrararna $K = 0.38$ och $T_i = 1.8$. Matlab-kod:

```
G=tf(4,[1 2 1]);
Kp=4;
L=0.3;
T=1.8;
Ghat=tf(Kp,[T 1],'InputDelay',L);
step(G,Ghat)
lambda=0.5;
K=T/(Kp*(lambda*T+L))
Ti=T
```

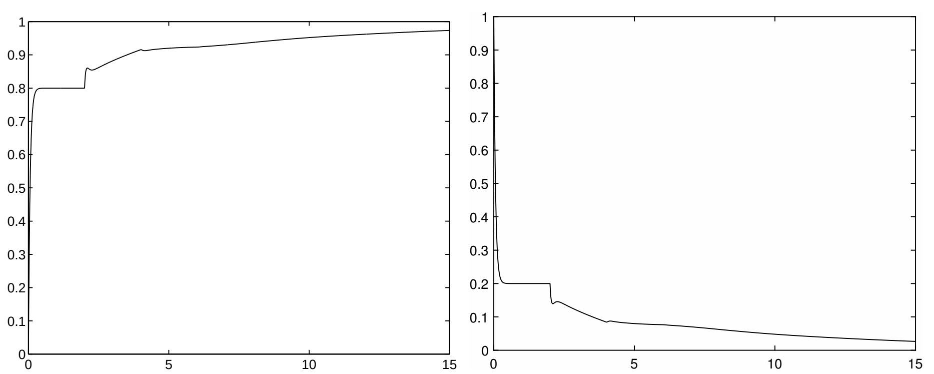
(b) Nej, det skulle inte gå om man använder en tidskontinuerlig P-regulator eftersom nyquistkurvan för systemet inte skär den negativa reella axeln. Detta följer av att systemet saknar nollställen och endast har två poler. Argumentet för $G(i\omega)$ kommer därför att vara skilt från $-180°$ för alla $\omega \geq 0$.

(c) Överslängens storlek kan minskas genom att man lägger till en D-del i regulatorn eller genom att skala referenssignalen i regulatorns P-del med en konstant $\alpha < 1$.

3. Vid ett enhetssteg i referenssignalen blir reglerfelet momentant lika med 1 och styrsignalen som genereras av P-regulatorn blir samtidigt $K \cdot 1$. Detta medför att det största värde på P-regulatorns förstärkning som man kan välja utan att styrsignalen mättas vid ett enhetssteg i referenssignalen är 1. Med denna P-regulator kan man sedan välja PI-regulatorns parametrar genom att simulera det slutna systemet för några olika inställningar. Till exempel verkar $K = -2$ och $T_i = 3$ fungera ganska bra. Referenssignalen till PI-regulatorn ska vara noll eftersom u_1 ska ligga i ett intervall som är symmetriskt runt noll. Ett simulinkschema för simulering av det slutna systemet visas i figur 2 och det slutna systemets stegsvar samt styrsignalen u_1 visas i figur 3.



Figur 2: Simulinkschema till uppgift 3.



(a) Stegsvaret för det slutna systemet.

(b) Styrsignalen u_1 vid ett steg i referenssignalen för y.

Figur 3: Plottar som visar att mitthållningsregulatorn i uppgift 3 fungerar som det är tänkt.

4. (a) Vid regulatordesign enligt IMC-metoden ska man välja

$$Q(s) = \frac{\tilde{G}(s)}{G(s)} = \frac{s+1}{8s+4}$$

för att få ett slutet system med de önskade referensföljningsegenskaperna. IMC-regulatorns överföringsfunktion blir

$$F_{IMC}(s) = \frac{Q(s)}{1-Q(s)G(s)} = 0.125\left(1+\frac{1}{s}\right).$$

Vid designen av den andra regulatorn ska man välja en framkoppling med referensmodellen

$$G_m(s) = \tilde{G}(s) = \frac{1}{2s+1}$$

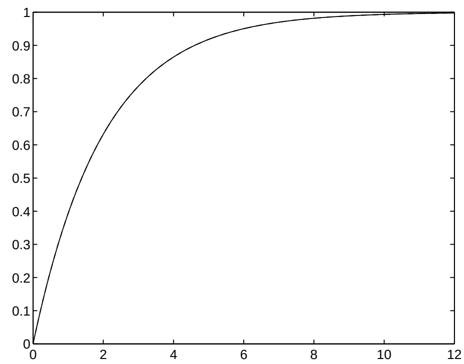
och framkopplingslänken

$$F_f(s) = \frac{G_m(s)}{G(s)} = \frac{s+1}{8s+4}$$

för att få de önskade referensföljningsegenskaperna. I detta fall kan man alltså använda sig av ideal framkoppling. PI-regulatorn med framkoppling ges av överföringsfunktionerna $G_m(s)$, $F_f(s)$ och $F_{PI}(s)$.

Stegsvaren som man erhåller med dessa båda regulatorer visas i figur 4. De båda regulatorerna ger exakt samma referensföljningsegenskaper. Matlab-kod:

```
G=tf(4,[1 1]);
Gtilde=tf(1,[2 1]);
Fpi=tf([5 5],[2 0]);
Q=minreal(Gtilde/G);
Fimc=minreal(Q/(1-Q*G));
Gcimc=minreal(feedback(Fimc*G,1));
Gm=Gtilde;
Ff=minreal(Gm/G);
Gryff=minreal((G*Fpi*Gm+G*Ff)/(1+G*Fpi));
step(Gcimc,Gryff)
```



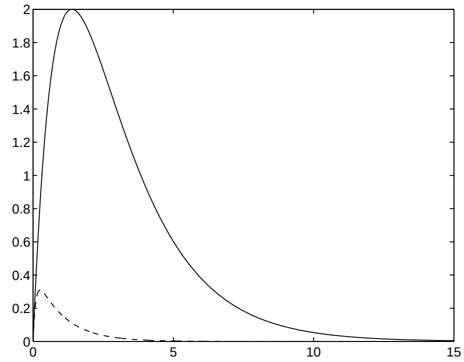
Figur 4: Stegsvaren för de båda slutna systemen i uppgift 4(a) sammanfaller helt.

(b) Överföringsfunktionerna från en insignalstörning till utsignalen i de båda slutna systemen ges av uttrycken

$$\frac{G(s)}{1+G(s)F_{IMC}(s)} \quad \text{respektive} \quad \frac{G(s)}{1+G(s)F_{PI}(s)}.$$

De utsignaler som man får vid stegstörningar på systemets ingång med de båda regulatorerna visas i figur 5. Regulatorn där den ursprungliga PI-regulatorn har kompletterats med en framkoppling visar sig ge klart bättre störningsundertryckning än IMC-regulatorn. Matlab-kod:

```
Su1=feedback(G,Fimc);
Su2=feedback(G,Fpi);
step(Su1,Su2)
```

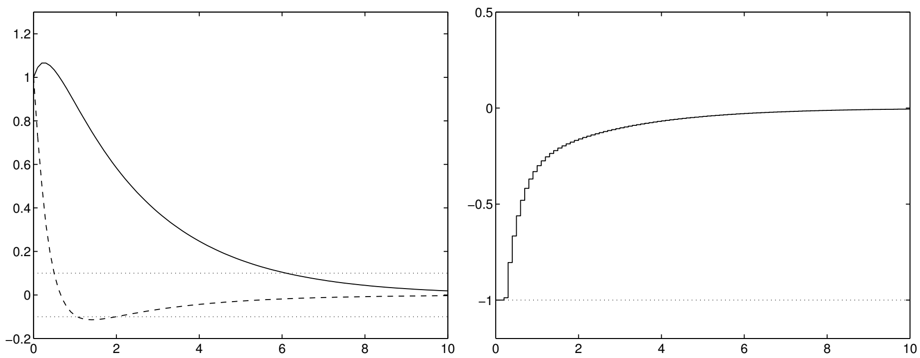


Figur 5: Utsignalen vid en stegstörning på systemets ingång då IMC-regulatorn (heldragen) och PI-regulatorn med framkoppling (streckad) används i uppgift 4(b).

5. (a) Man kan till exempel välja regulatorparametrarna $N = 20$,

$$Q_1 = \begin{pmatrix} 5.4 & 0 \\ 0 & 1 \end{pmatrix}$$

och $Q_2 = 1$. Plottar som visar att det slutna systemet uppfyller de ställda kraven visas i figur 6.



(a) Tillståndet x_1 (streckat) och x_2 (heldraget).

(b) Insignalen u.

Figur 6: Plottar som visar att designkraven i uppgift 5(a) är uppfyllda.

Matlab-kod:

```
A=[-2 0; 1 -0.2];
B=[1; 0.2];
C=eye(2);
D=zeros(2,1);
Gsys=ss(A,B,C,D);
M=eye(2);
x0=[1;1];
Ts=.1;
Gsysd=c2d(Gsys,Ts);
F=Gsysd.A;
G=Gsysd.B;
N=20;
Q1=diag([5.4 1]);
Q2=1;
ubounds=[-1 1];
```

(b) Bivillkoret $z_1 - z_2 \geq -1$ är ekvivalent med $-z_1 + z_2 \leq 1$ och kan ses som ett bivillkor på $\tilde{z} = \tilde{M}z$ om man väljer $\tilde{M}$-matrisen som

$$\tilde{M} = \begin{pmatrix} -1 & 1 \end{pmatrix}.$$

MPC-regulatorn kommer att ta hänsyn till bivillkoret $\tilde{z} \leq 1$ om man modifierar `solvempcproblem.m` till

```
function un=solvempcproblem(x,F,G,M,N,Q1,Q2,ubounds,Ts,t)
warning off;
[n,m]=size(G);
[H,S]=createpredictors(F,G,N);
Q1b=blockrepeat(Q1,N);
Q2b=blockrepeat(Q2,N);
Mb=blockrepeat(M,N);
Mt=[-1 1];
Mtb=blockrepeat(Mt,N);
Az=Mtb*S;
bz=ones(N,1)-Mtb*H*x;
Au=[eye(N*m);-eye(N*m)];
bu=[repmat(ubounds(:,2),N,1);repmat(-ubounds(:,1),N,1)];
A=[Au;Az];
b=[bu;bz];
options=optimset('Display','on');
U=quadprog(S'*Mb'*Q1b*Mb*S+Q2b,S'*Mb'*Q1b*Mb*H*x,A,b,[],[],[],[],[],options);
un=U(1:m);
```