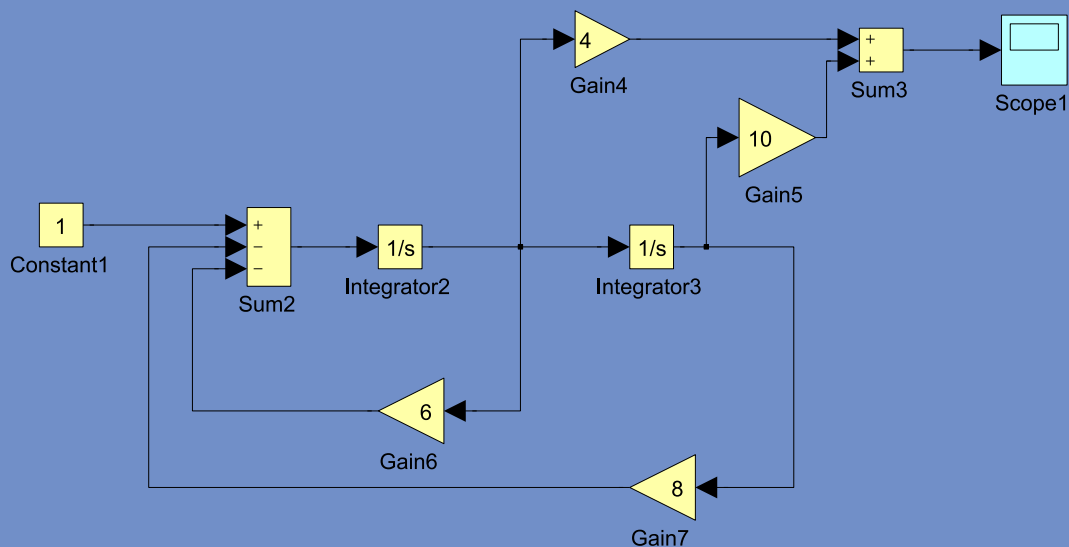




MODELIRANJE IN OBDELAVA SIGNALOV



Univerza v Ljubljani
Fakulteta za elektrotehniko

**MODELIRANJE IN
OBDELAVA SIGNALOV**

Borut Zupančič

Ljubljana 2011

Kazalo

1	Uvod	1
2	Sistemi in signali	3
2.1	Sistemi in sistemska teorija	3
2.2	Različni primeri sistemov	5
2.3	Procesi	7
2.4	Bistvo sistemske teorije v povezavi z modeliranjem	10
2.5	Signali	14
2.5.1	Osnovni pojmi	14
2.5.2	Vrste signalov	15
2.5.3	Moč in energija signala	16
2.5.4	Obdelava signalov	17
2.5.5	Temeljni signali	19
2.5.6	Uvod v spektralno analizo	27
2.5.7	Spektralna analiza s Fourier-jevo vrsto	29
2.5.8	Napake pri uporabi Fourier-jeve vrste	36
2.5.9	Numerično računa je spektrov - diskretna Fourier-jeva transformacija	39

3 Modeliranje procesov	53
3.1 Cilji in pomembnejše zakonitosti pri modeliranju	53
3.2 Vrste modelov	55
3.3 Načini modeliranja	61
3.3.1 Teoretično modeliranje	62
3.3.2 Eksperimentalno modeliranje	64
3.3.3 Kombinirano modeliranje	65
3.4 Modeliranje in simulacija kot enovit ciklični postopek	66
3.5 Primeri	69
4 Zapisi matematičnih modelov	83
4.1 Zapis z diferencialno enačbo	83
4.2 Laplace-ova transformacija	84
4.3 Reševanje diferencialne enačbe z Laplace-ovo transformacijo	87
4.4 Prenosne funkcije	90
4.5 Bločni diagrami	96
4.5.1 Splošne značilnosti bločnih diagramov	96
4.5.2 Postopek za risanje bločnih diagramov	100
4.5.3 Poenostavljanje bločnih diagramov	104
5 Računalniško podprta analiza in simulacija sistemov	109
5.1 Vnos linearnih dinamičnih sistemov	109

5.2	Osnovne zmožnosti dela s sistemi	111
5.3	Simulacija s pomočjo Matlabovih funkcij	112
6	Osnovne metode pri reševanju problemov s simulacijo	119
6.1	Simulacijska shema	119
6.2	Indirektna metoda	121
6.3	Simulacija prenosnih funkcij	126
6.4	Princip modularnosti	129
7	Orodja za simulacijo dinamičnih sistemov	131
7.1	Osnovne lastnosti simulacijskih sistemov	131
7.2	Simulacijsko okolje Matlab-Simulink	132
7.2.1	Osnovna uporaba Simulinka	132
7.2.2	Analiza modelov v Simulinku	143
7.2.3	Poglobljena uporaba Simulinka	144
7.3	Primeri v okolju Matlab - Simulink	147
8	Analiza sistemov v časovnem prostoru	155
8.1	Osnovne lastnosti časovnega odziva	155
8.2	Vpliv polov in ničel na časovni odziv	157
8.3	Razvrstitev sistemov	160
8.4	Proporcionalni sistemi	163

8.4.1	Sistem prvega reda	163
8.4.2	Sistem drugega reda	168
8.5	Integrirni sistemi	177
8.6	Diferencirni sistemi	178
8.7	Sistemi z mrtvim časom	180
8.8	Stabilnost sistemov	181
8.8.1	Pogoj za stabilnost linearnih, časovno nespremenljivih sistemov	181
8.8.2	Routhov stabilnostni kriterij	182
	Literatura	188

1.

Uvod

Čim boljša kakovost izdelkov je ena od najpomembnejših zahtev pri uspešni računalniško vodeni proizvodnji. Kakovost spremljamo s številnimi predpisanimi postopki, ki zahtevajo določeno eksperimentiranje in merjenje proizvodov, npr. pregrevanje, mehansko obremenjevanje itd. Eksperimentiranje pa je lahko problematično zaradi dolgotrajnosti, nevarnosti, zaradi velikih stroškov, zaradi ekološko nedopustnih posledic itd. Idealna rešitev je nadomestitev vsaj dela realnih eksperimentov z računalniškimi eksperimenti, t.j. z modeliranjem in simulacijo. S slednjim pa je eksperimentiranje bistveno bolj fleksibilno.

Modeliranje pomeni, da realni proces zapišemo s pomočjo matematičnih izrazov, simulacija pa pomeni, da enačbe pretvorimo v računalniški program.

Modeliranje in simulacija je ključnega pomena pri zagotovitvi velike dodane vrednosti v visoko tehnološki proizvodnji. S tem dosežemo boljše razumevanje mehanizmov delovanja, napovedujemo obnašanje, ocenjujemo nemerljive parametre, analiziramo občutljivost, načrtujemo vodenje sistemov, uporabljamo simulatorje za vadbo itd. Vse te komponente pa so bistvene pri doseganju visoke kakovosti. Osnova področja modeliranja in simulacij pa sta teorija sistemov in teorija signalov, zato bomo tema pojma posvetili nekaj pozornosti 2. poglavju.

2.

Sistemi in signali

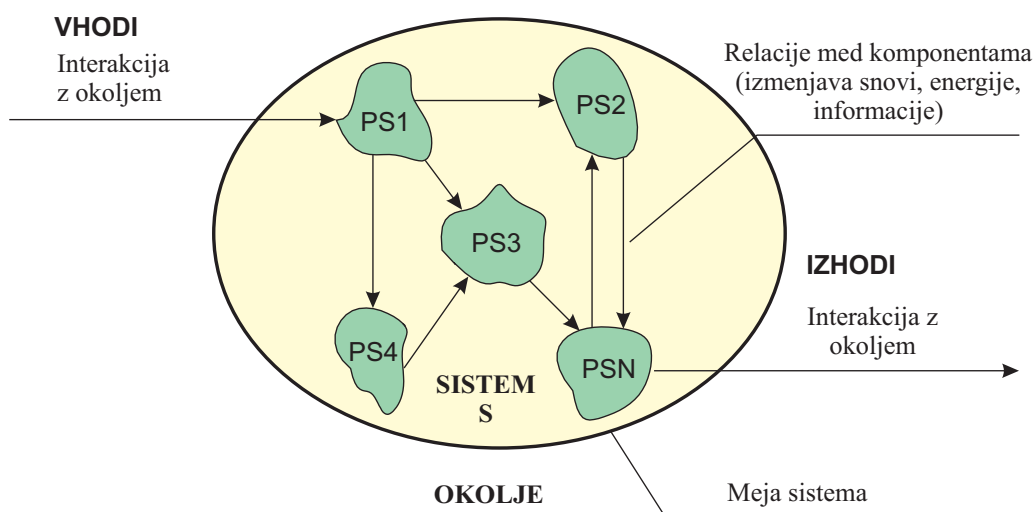
Uspešno reševanje problemov vodenja zahteva po eni strani poglobljeno razumevanje samih sistemov in procesov, ki jih vodimo, njihove strukture, dinamike, medsebojnih relacij, splošnih zakonitosti in podobnosti, ki se pojavljajo na različnih področjih, itd., po drugi strani pa obvladovanje splošnih konceptov in principov, na katerih vodenje sistemov temelji. To pa je predvsem **sistemska teorija**.

2.1 Sistemi in sistemska teorija

Sistemska teorija tvori najširši konceptualni okvir za reševanje številnih tehničnih problemov. Poglejmo si nekaj njenih splošnih značilnosti.

Beseda **sistem** izhaja iz grščine in v dobesednem prevodu pomeni: "povzročati, da stoji skupaj". V tem smislu pomeni sistem neko urejeno tvorbo, ki predstavlja nasprotje naključnosti ali kaosa. Ena od najbolj splošnih definicij pravi, da je **sistem množica elementov, ki imajo medsebojne relacije in relacije z okoljem** (Bertalanffy, 1972). Shemo tako razumljenega sistema vidimo na sliki 2.1.

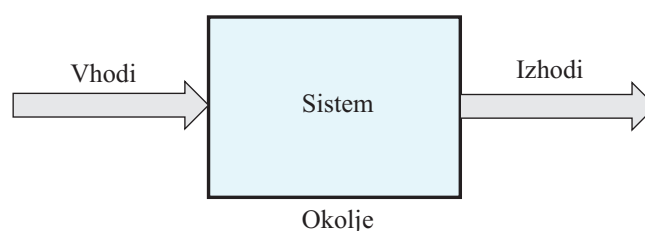
Vidimo, da ima sistem S neko notranjo strukturo, ki je definirana z različnimi komponentami. Vsaka komponenta ima svoje značilnosti in je običajno zopet sistem nižje kompleksnosti (podsistemi PS_1 do PS_N). Komponente so povezane z relacijami. Te relacije običajno pomenijo izmenjavo snovi, energije in/ali infor-



Slika 2.1: Shema sistema z notranjo strukturo

macije. Kaj je zunaj in kaj znotraj sistema, opredeljujejo meje sistema. Meja sistema torej loči sistem od okolja. Določene komponente sistema nimajo samo medsebojnih relacij, ampak imajo tudi relacije z okoljem. Te relacije predstavljajo interakcijo z okoljem. Relacijam iz okolja, ki imajo vpliv na posamezne komponente sistema, pravimo vhodi v sistem. Nasprotno pa relacijam, ki vplivajo na okolje, pravimo izhodi iz sistema.

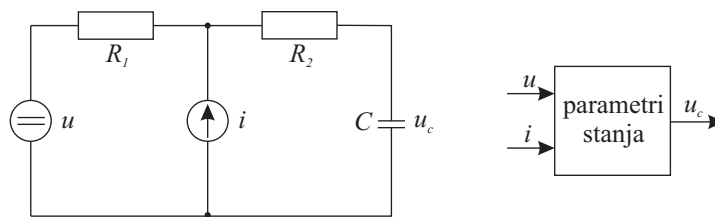
Mnogokrat nas notranja struktura sistema ne zanima, zanima pa nas medsebojna odvisnost med vhodi in izhodi sistema in njegova povezanost z okoljem. Zato često sistem predstavimo tudi v obliki, ki jo prikazuje slika 2.2.



Slika 2.2: Vhodno-izhodni prikaz sistema

Primer: Slika 2.3 prikazuje, kako električno shemo spremenimo v vhodno-izhodni blok. V primeru bloka nas vsaj struktura notranjosti ne zanima. Običajno nas zanimajo le parametri in stanja.

Najbolj pomembno je, da razumemo, v čem je bistvo sistema. Osnovni smisel



Slika 2.3: Transformacija električne sheme v vhodno-izhodni prikaz

sistema je, da predstavlja več kot zgolj vsoto posameznih komponent. Že Aristotel je ugotavljal, da je celota več kot vsota sestavnih delov. To, kar je več, ni rezultat ene posamezne komponente, ampak načina povezave med njimi, ki rezultira v sinergiji. Ta sinergija pomeni novo kvaliteto in kvantiteto. (Dvokolo npr. dobimo, ko na pravi način sestavimo dve kolesi z ogrodjem, dodamo sedež, krmilni mehanizem (balanco) in mehanizem za poganjanje in zaviranje. Dobili smo nekaj, kar ima v primerjavi z osnovnimi sestavnimi deli novo funkcijo in nov smisel.)

Sistemi so seveda zelo različni, zato si pri njihovi obravnavi pomagamo z različnimi klasifikacijami po različnih kriterijih. Namen tega dela ni, da bi se obširneje ukvarjali s to problematiko. Zgolj zaradi ilustracije širine koncepta sistema pogledjmo eno od možnih klasifikacij. Klasifikacija je prikazana na sliki 2.4.

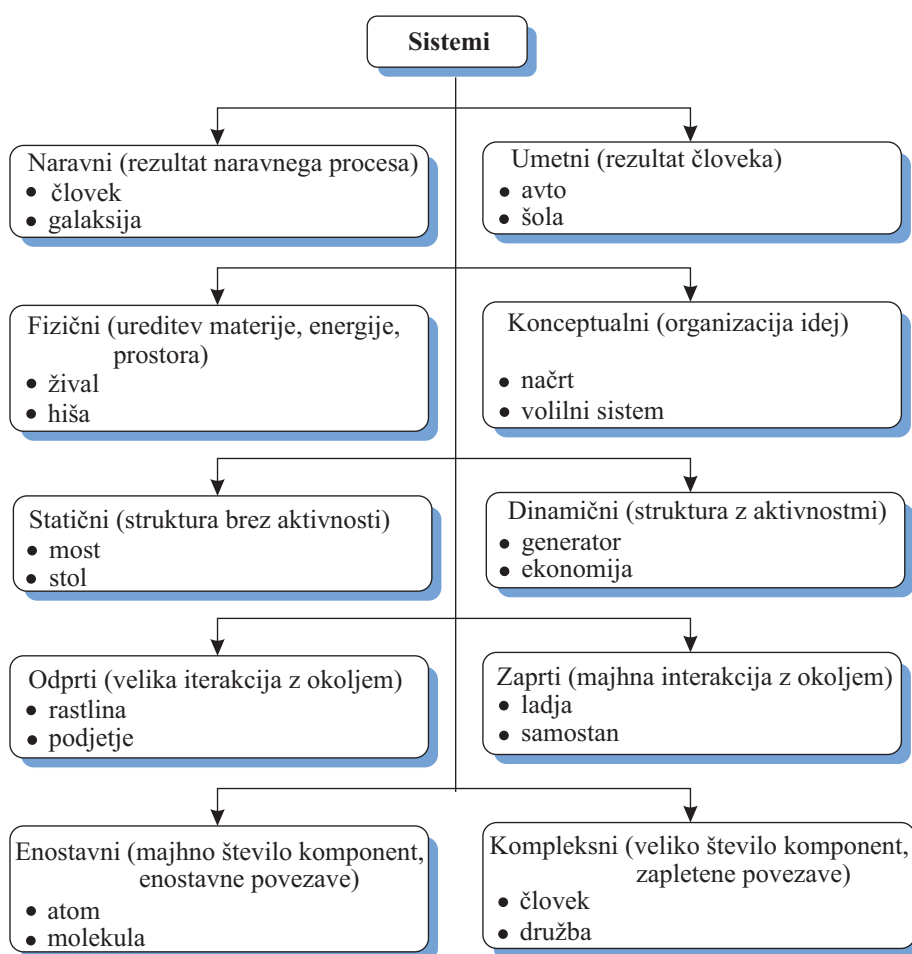
2.2 Različni primeri sistemov

Poglejmo si na osnovi doslej spoznanega nekaj primerov sistemov. Prvi primer sistema naj bo človek. Shematsko je kot vhodno-izhodni sistem prikazan na sliki 2.5.

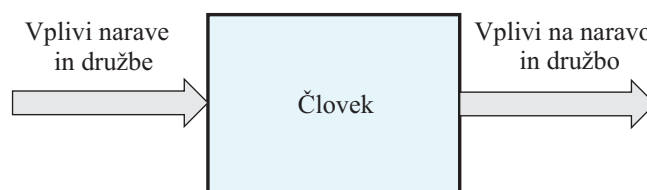
Vidimo, da vhode lahko opredelimo kot vplive narave in družbe na človeka, izhode pa kot človekov vpliv na naravo in družbo. Če želimo človeka kot sistem klasificirati, bi lahko rekli, da gre za naravni, fizični, dinamični, odprti in kompleksni sistem.

Drugi primer sistema je televizijski sprejemnik. Shematsko je prikazan na sliki 2.6.

Vhode v sistem predstavljajo električna energija in televizijski signali v obliki



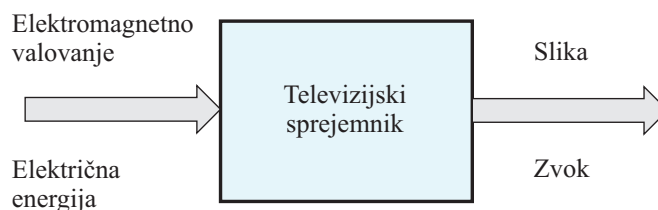
Slika 2.4: Klasifikacija sistemov



Slika 2.5: Človek kot sistem

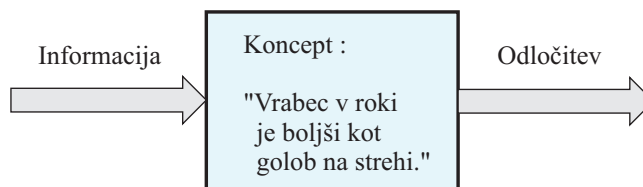
elektromagnetnega valovanja. Izhoda iz sistema sta slika in zvok. Televizijski sprejemnik bi lahko opredelili kot umetni, fizični, dinamični, odprti in srednje enostavni sistem.

Tretji primer sistema je konceptualni (miselni) sistem. Gre za znani pregovor:



Slika 2.6: Televizijski sprejemnik kot sistem

"Vravec v roki je boljši kot golob na strehi", ki v resnici predstavlja način obnašanja v določeni situaciji. Shema sistema je prikazana na sliki 2.7.



Slika 2.7: Koncept kot sistem

Vhod v sistem v tem konkretnem primeru predstavlja informacija, ki omogoča izbiro med različnimi možnostmi, izhod pa odločitev, ki izbira eno od možnosti v skladu z opredeljenim konceptom. Ta sistem bi lahko opredelili kot umetni, konceptualni, statični, odprti in enostavni sistem.

2.3 Procesi

Tematika tega dela je posvečena vodenju procesov, zato si še na kratko pogledimo, kakšna je razlika oz. relacija med procesom in sistemom. Beseda proces izhaja iz latinske besede "procedere" (napredovati) in pomeni nek postopek ali dogajanje (Lauber, 1989). Ta postopek oziroma dogajanje povzroči spremembo stanja v sistemu. Ko govorimo o procesu je torej ključna komponenta čas, saj samo na osnovi opazovanja nekega sistema v različnih časovnih trenutkih lahko opazimo spremembo stanja. Če smo prej definirali strukturo sistema kot relacijo med posameznimi komponentami in ugotovili, da te relacije pomenijo izmenjavo snovi, energije in informacije, potem lahko rečemo, da v sistemu poteka proces, če ta izmenjava rezultira v spremembah stanja posameznih komponent. Procesov je seveda veliko vrst (npr. naravni proces, politični proces, organizacijski proces, poslovni proces, psihološki proces, itd.), mi se bomo ukvarjali s tehničnimi

procesi. Po standardu DIN 66201 (Lauber, 1989) je tehnični proces skupek soodvisnih potekov v nekem sistemu, ki rezultirajo v transformaciji, transportu ali skladiščenju materije, energije ali informacije. Sistem je torej okvir, v katerem potekajo procesi ali pa sredstvo, ki omogoča potek procesa. Proces pa je vsebina sistema ali njegovo življenje. Shematsko je prikazan na sliki 2.8.



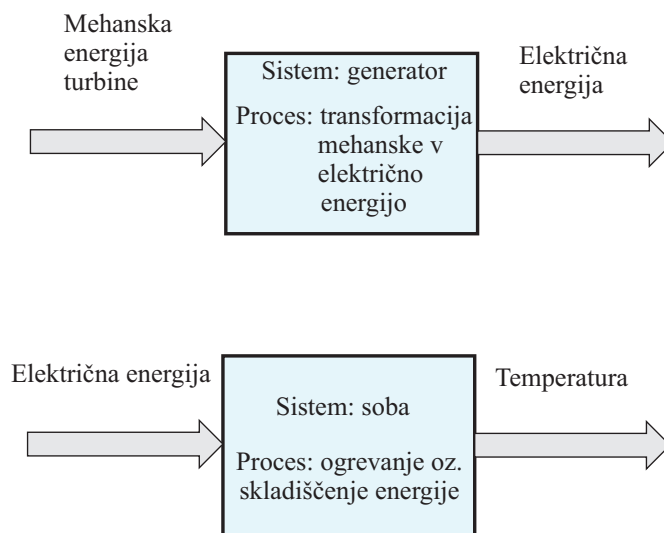
Slika 2.8: Shematski prikaz procesa

Sistem, v katerem potekajo procesi, je torej tisto, kar smo že nekoliko prej opredelili kot "dinamični sistem". Težko najdemo sistem, v katerem ne bi potekali procesi. Tudi sistemi, ki smo jih definirali kot statične (npr. most, stol), so podvrženi počasni degradaciji. V tem smislu so najmanj podvrženi spremembam nekateri konceptualni sistemi, abstraktne relacije med pojmi, itd. Obstajajo pa sistemi, katerih bistvo je, da omogočajo nek proces, oziroma, da iz začetnega stanja snovi, energije in informacije omogočajo prehod v neko končno stanje. Taki sistemi so npr. tovarna, ki omogoča izdelavo izdelkov; ladja, ki omogoča transport izdelkov in surovin; generator, ki omogoča pretvorbo mehanske v električno energijo, soba z električnim grelom, ki omogoča segrevanje, itd. Shematski prikaz slednjih dveh primerov lahko vidimo na sliki 2.9.

Tako s stališča modeliranja kot vodenja je pomembna delitev procesov na:

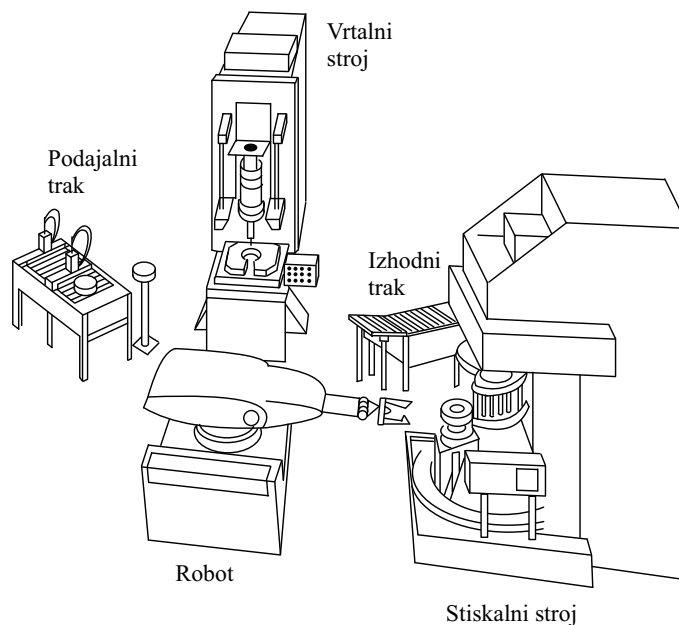
- kosovne procese,
- zvezne (kontinuirne) procese in
- šaržne procese.

Kosovne imenujemo procese, pri katerih se objekti, ki jih obdelujemo, pojavljajo v določenih časovnih razmakih (npr.: v izdelni industriji proizvodi iz metalov, lesa, plastike, stekla itd.). Tako modeliranje kot vodenje temeljita na teoriji diskretnih dogodkov (angl. discrete event systems). Primer takega procesa je delovna celica (slika 2.10), ki združuje več različnih strojev za različne obdelave posameznega



Slika 2.9: Sistema generator in soba s pripadajočima procesoma

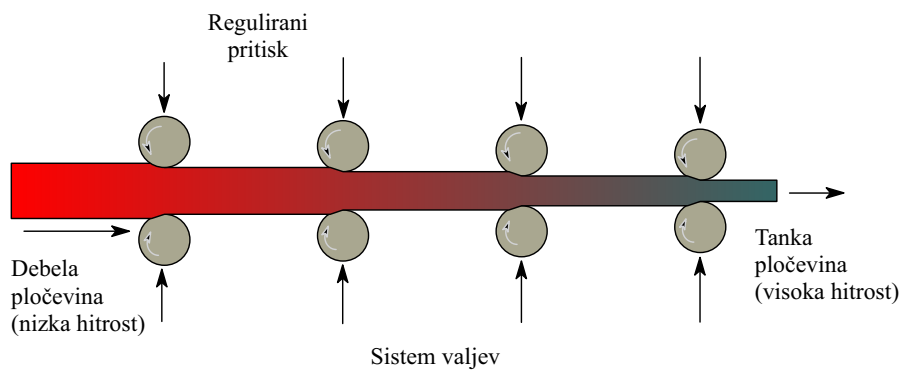
kosa. Kovinsko ploščo robotska roka prestavi na numerično krmiljen (CNC) vrtni stroj, ki izvrti različne vrtnice, ga prestavi na drug stroj (stiskalnico), ki izseka določene oblike in ga odstavi na odlagalno mesto.



Slika 2.10: Primer kosovnega procesa

Pri zveznih (kontinuirnih) procesih masa in/ali energija prihajata skozi enote

procesa v neprekinjenih tokovih (n.pr. v parnih kotlih, toplotnih izmenjevalnikih, pretočnih kemičnih reaktorjih, v valjarnah, itd.). Obravnava temelji na teoriji zveznih dinamičnih sistemov. Primer zveznega obdelovalnega procesa je proizvodnja jeklene pločevine (slika 2.11), ko iz velikih žarečih vložkov (ingotov) z zaporednim valjanjem tanjšamo debelino, dokler končno ne dobimo pločevine. Z zmanjševanjem debeline narašča hitrost valjanja (do 500 [m/min]), zato moramo proces ustrezno regulirati.

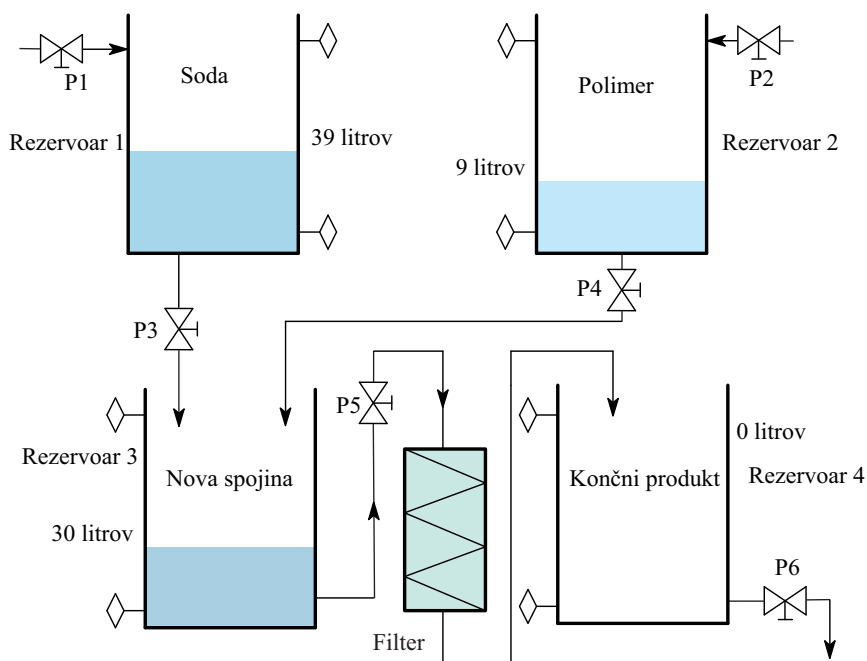


Slika 2.11: Primer zveznega obdelovalnega procesa

Pri šaržnih procesih pa pride do prekinitev omenjenih tokov. V glavnem razlikujemo tri faze: polnjenje procesnih enot, odvijanje (običajno zveznih) procesov in praznjenje procesnih enot (n.pr.: šaržni kemični reaktorji, rotacijske peči v industriji cementa, plavži itd.). Obravnava je v teh primerih bolj zamotana, saj združuje teoriji zveznih in diskretnih sistemov. Primer šaržnega procesa, ko iz dveh snovi sode in polimera naredimo novo spojino, jo filtriramo in pretočimo v skladiščni rezervoar, prikazuje slika 2.12.

2.4 Bistvo systemske teorije v povezavi z modeliranjem

Človek je v celotnem obdobju svoje zgodovine proučeval naravne sisteme ter snoval in gradil umetne sisteme. Rezultat tega razvoja je bilo spoznanje, da pri proučevanju strukture sistemov, njihove dinamike, medsebojnega vpliva, razvoja in vodenja naletimo na pojave, ki niso lastni samo sistemu določene vrste, ampak jih srečamo tudi pri drugih sistemih. Prav to spoznanje je bilo ena od osnovnih motivacij za nastanek systemske teorije. Pomemben impulz za nastanek



Slika 2.12: Primer šaržnega procesa

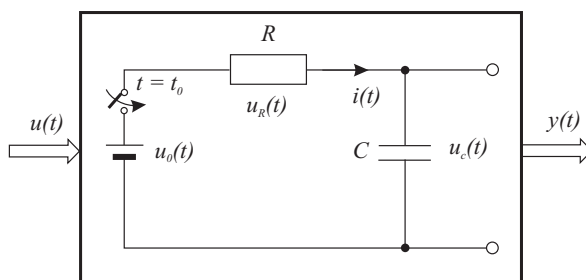
sistemske teorije je predstavljal razvoj abstraktnega (predvsem matematičnega) zapisa posameznih pojavov oziroma sistemov. Pokazalo se je namreč, da imajo povsem različni sistemi ali bolje rečeno modeli različnih sistemov enak matematični zapis. Pravimo, da velja analogija med ustreznimi spremenljivkami sistema. **Zato je bistvo sistemske teorije, da omogoča obravnavo dinamičnih sistemov iz različnih področij na poenoten in sistematičen način.**

Razvoj modeliranja zato sloni prav na sistemski teoriji.

Poglejmo si primer takšne analogije.

Iz osnov elektrotehnike dobro poznamo sistem - RC vezje, ki ga prikazuje slika 2.13.

Če želimo določiti potek napetosti na kondenzatorju, oz. kako se polni kondenzator, ko v trenutku $t = 0$ vključimo vir konstantne napetosti u_0 , moramo priti do diferencialne enačbe, ki povezuje vhodno spremenljivko $u(t) = u_0(t)$ in izhodno spremenljivko $y(t) = u_C(t)$. V tem primeru sta $u_0(t)$ in $u_C(t)$ vhodni in izhodni spremenljivki, medtem ko smo uvedli spremenljivki $u(t)$ in $y(t)$ kot spremenljivki, ki jih uporabljamo za vhod in izhod v posplošeni obliki. Z uporabo Kirchofovih



Slika 2.13: RC vezje

zakonov pridemo v nekaj korakih do enačbe 2.2.

$$\begin{aligned}
 u_0(t) &= u_R(t) + u_C(t) \\
 u(t) &= u_R(t) + y(t) \\
 u_R(t) &= i(t)R = RC \frac{du_C(t)}{dt} = RC \frac{dy(t)}{dt} \\
 u(t) &= RC \frac{dy(t)}{dt} + y(t) \\
 \frac{dy(t)}{dt} + \frac{1}{RC}y(t) &= \frac{1}{RC}u(t) \\
 y(0) &= u_C(0) = y_0
 \end{aligned} \tag{2.1}$$

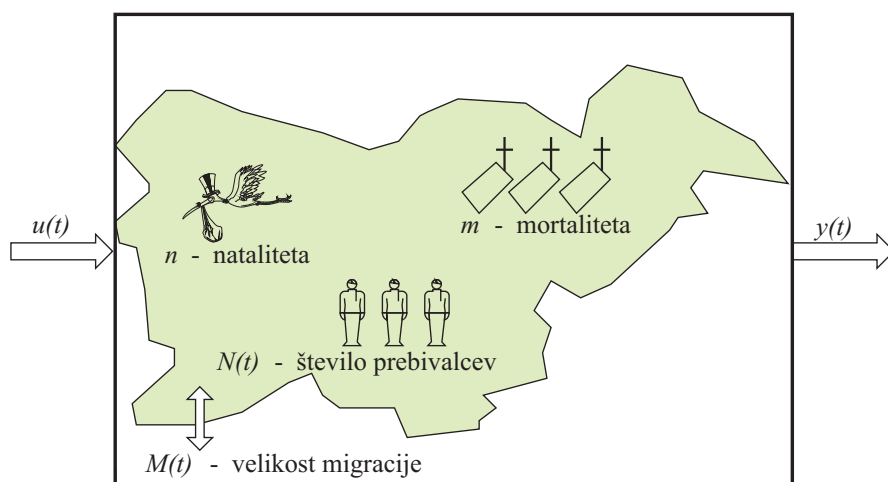
Dobili smo enostavno diferencialno enačbo 1. reda, ki je v posplošeni obliki (vpeljava konstant a in b)

$$\frac{dy(t)}{dt} + ay(t) = bu(t) \quad y(0) = y_0 \tag{2.2}$$

Diferencialna enačba 1. reda je rešljiva, če je podan en začetni pogoj ($y(0) = y_0$) in če je podan vhodni signal $u(t)$.

In še drugi primer. Vzemimo, da nas zanima gibanje števila prebivalcev $N(t)$ v neki državi. Sprememba v številu prebivalcev $\frac{dN(t)}{dt}$ bo seveda odvisna od natalitete (n), mortalitete (m) in od števila priseljenih na časovno enoto, t.j. od migracije $M(t)$ (glej sliko 2.14).

Intuitivno lahko razumemo, da je sprememba številčnosti $\frac{dN(t)}{dt}$ odvisna od razlike



Slika 2.14: Prebivalstvo države

med novo rojenimi in umrlimi $((n - m)N(t))$ in od migracije $M(t)$.

$$\begin{aligned}
 \frac{dN(t)}{dt} &= (n - m)N(t) + M(t) \\
 N(t) &= y(t) \\
 M(t) &= u(t) \\
 N(0) &= y_0
 \end{aligned}
 \tag{2.3}$$

Če označimo $a = m - n$, $b = 1$ pridemo ponovno do enačbe 2.2.

Potrdili smo tezo, da imajo na videz zelo različni sistemi enak matematični zapis oz. enak matematični model, zato pravimo, da velja analogija med opisanimi sistemoma (napetost na kondenzatorju in številčnost prebivalcev sta analogni spremenljivki).

S primeroma električnega vezja in prebivalstva smo začeli z obravnavo modeliranja, ki je pomembno področje avtomatskega vodenja sistemov.

2.5 Signali

2.5.1 Osnovni pojmi

Teorija signalov je ena najstarejših ved systemske teorije (Mihelič, 2010, Mlakar, 2002, Cha, Molinder, 2006). Beseda signal izvira iz latinske besede *signum*, kar pomeni znak.

Definicija signala: Signali so fizikalne tvorbe oz. nosilci sporočil oz. informacij, ki jih sistemi uporabljajo za medsebojno sporazumevanje.

Šum je le posebna vrsta signala, ki v sistemih predstavlja motilni vpliv. Zato namesto tega izraza zlasti v sistemih vodenja uporabljamo izraz motilni signal ali motnja.

V zvezi s signali pogosto govorimo o generatorjih signalov (npr. govor) ali oddajnikih in o sprejemnikih signalov (npr. vid). V teoriji sistemov pa govorimo o vhodnih (vplivnih) signalih in o izhodnih signalih (opazovanih, merjenih signalih). To pa je le del signalov, ki jim lahko določimo t.i. vzročnost (angl. *causality*), kar pomeni, da lahko signalom določimo, ali so vzrok ali posledica (npr. enosmerni električni vir povzroči napetost na kondenzatorju). Večina signalov v sistemu pa ima t.i. nevzročni karakter, kar pomeni, da ne moremo govoriti o odnosu vzrok-posledica (npr. za razmere na električnem uporu nikoli ne moremo reči, ali tok povzroči napetost ali napetost povzroči tok).

Za razvoj teoretične obdelave signalov so pomembne iznajdbe sistemov za komunikacijo (telegraf, telefon, radio, televizija, ...) iz 19. in začetka 20. stoletja, ter digitalnih računalnikov v drugi polovici 20. stoletja. Torej so na razvoj pretežno vplivali električni signali. Nekaj mejnikov v zvezi z iznajdbami iz audio tehnologij:

- 1877 gramofon
- 1920 AM v radiu
- 1956 stereo
- 1960 stereo FM
- 1982 CD (Sony in Philips)

Signale kot časovne funkcije prikazujemo na osciloskopu, na risalniku, njihovo frekvenčno vsebino oz. frekvenčni spekter pa na spektralnem analizatorju.

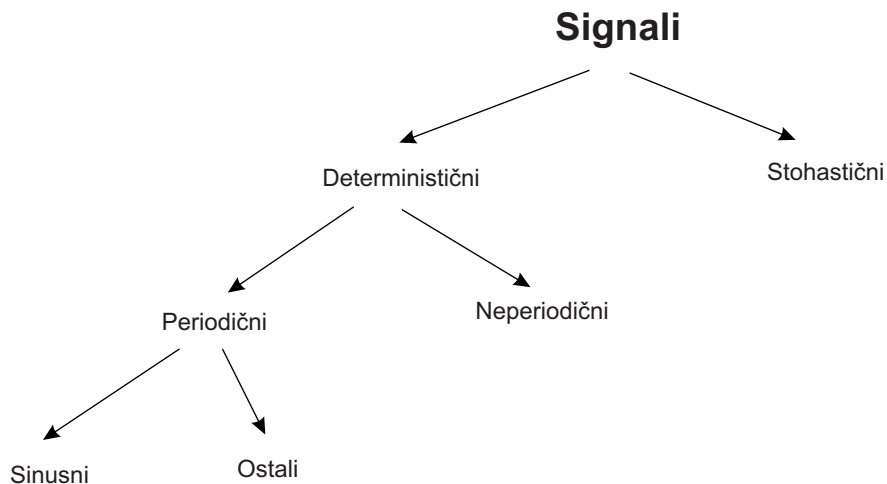
2.5.2 Vrste signalov

Signale se deli na različne načine. Ena delitev je delitev na fizikalno realne signale in na njihove teoretične modele.

Fizikalno realni signali so signali, ki nas obdajajo v prostoru in času in jih lahko zaznamo z ustreznimi tipali (senzorji). Taki signali imajo omejeno velikost-amplitudo, omejeno energijo in omejen spekter.

Teoretični modeli signalov so predstavljeni kot realne ali kompleksne funkcije običajno ene neodvisne spremenljivke (ponavadi časa). V obravnavi sistemov in modelov imamo često opravka z teoretičnimi modeli signalov.

Signale delimo tudi na deterministične (določljivostne) in stohastične (naključne) signale (glej sliko 2.15).



Slika 2.15: Delitev signalov

Deterministični signali so signali, pri katerih je odvisnost amplitude od časa enolično določena. Torej signalu lahko določimo model z ustrežno matematično funkcijo. Signal $f(t)$ je periodičen natanko takrat, ko je za vsako vrednost časa t izpolnjen pogoj

$$f(t + T) = f(t) \quad (2.4)$$

kjer je T različen od nič ($T \neq 0$). Konstanta T je perioda signala $f(t)$.

Signal je neperiodičen, če ni periodičen.

Stohastični signali so signali, katerih časovno obnašanje je nepredvidljivo. Zato signalov ne moremo opisovati z matematičnimi funkcijami, temveč si pomagamo z nekaterimi statističnimi značilkami (verjetnostna porazdelitev, srednja vrednost, standardna deviacija, ...).

2.5.3 Moč in energija signala

Trenutna moč signala

Če je v električnem vezju t konkreten časovni trenutek, $i(t)$ tok na uporu R v tistem trenutku, $v(t)$ pa napetost, potem trenutno moč določimo kot

$$P(t) = i(t)v(t) = i^2(t)R = \frac{1}{R}v^2(t) \quad (2.5)$$

Trenutna moč je torej proporcionalna kvadratu toka ali napetosti. Na osnovi opisanih relacij pa posplošimo definicijo trenutne moči, ki velja za realne in kompleksne signale

$$P_f(t) = |f(t)|^2 \quad (2.6)$$

Energija signala

Skladno z definicijo trenutne moči definiramo energijo signala $f(t)$ na končnem časovnem intervalu (t_1, t_2) kot

$$E_f(t_1, t_2) = \int_{t_1}^{t_2} |f(t)|^2 dt \quad (2.7)$$

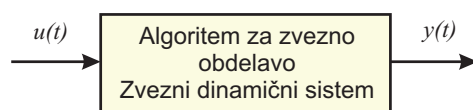
Povprečna moč signala

V skladu z definicijama trenutne moči in energije definiramo povprečno moč signala $f(t)$ na časovnem intervalu (t_1, t_2)

$$P_f(t_1, t_2) = \frac{E_f(t_1, t_2)}{t_2 - t_1} = \frac{1}{t_2 - t_1} \int_{t_1}^{t_2} |f(t)|^2 dt \quad (2.8)$$

2.5.4 Obdelava signalov

Klasična obdelava signalov se je v preteklosti vršila predvsem s pomočjo analognih elektronskih vezij. Zvezni signal vodimo v zvezni dinamični sistem, ki ga lahko opišemo z diferencialnimi enačbami, prenosnimi funkcijami ali kakšnimi drugimi zveznimi zapisi. Zvezno obdelavo signalov prikazuje slika 2.16.



Slika 2.16: Zvezna obdelava signalov

Dandanes pa se za obdelavo signalov pa tudi za sprotno vodenje procesov vse bolj uporabljajo digitalni računalniki (slika 2.17, Cha, Molinder, 2006). Za razumevanje digitalne obdelave signalov je potrebna delitev signalov po tem, kako je določeno definicijsko območje in zaloga vrednosti signala.

Zvezni signal

je funkcija, ki je definirana na zveznem časovnem intervalu in katere amplituda lahko zavzame zvezno področje vrednosti. Torej tako neodvisna kot odvisna spremenljivka lahko zavzameta kakršnekoli vrednosti iz definicijskega območja oz. zaloge vrednosti (slika 2.17a).

Vzorčenje

V primeru, ko želimo signale predstaviti v digitalnem računalniku, smo zaradi tehnoloških omejitev prisiljeni v računalnik prenesti le t.i. vzorce v končnem številu običajno ekvidistantnih časovnih trenutkov. Časovni razliki med dvema zaporednima vzorcema rečemo čas vzorčenja T . Postopek se imenuje vzorčenje

(angl. sampling) in se opravi znotraj analogno/digitalnih pretvornikov na računalniku (A/D). Signal, ki ima določene vrednosti odvisne spremenljivke le v trenutkih vzorčenja, se imenuje **diskretni signal** (na sliki 2.17a predstavljen z usmerjenimi črtami). Pri vzorčenju je znan Shannonov teorem, ki pravi, da je signale potrebno vzorčiti z vsaj 2x večjo frekvenco vzorčenja kot je najvišja frekvenca signala, ki ga vzorčimo (f_m)

$$f > 2f_m \quad (2.9)$$

Le v takem primeru je možno vzorčeni signal brez izgub rekonstruirati nazaj v zvezni signal. Shannonov teorem določa teoretično frekvenco vzorčenja. V praksi vzamemo po možnosti 5-10x višjo frekvenco vzorčenja

$$f = (5 - 10)f_m \quad (2.10)$$

Kvantizacija

Prav tako pa smo zaradi tehnoloških omejitev prisiljeni za amplitude izbirati le med končnim številom različnih vrednosti. Govorimo o končnem številu kvantizacijskih nivojev (npr. 256, če signale kodiramo z 8 biti, 8, če signale kodiramo s 3 biti). Vrednosti zveznega signala v trenutku vzorčenja priredi računalnik vrednost najbližjega kvantizacijskega nivoja oz. njegove kode. Ustrezen postopek se imenuje kvantizacija in pogrešek, ki pri tem nastane, je kvantizacijski pogrešek. Tudi kvantizacija se opravi znotraj A/D pretvornika. Tako pridemo do **digitalnega signala**, kjer sta čas in amplituda kvantizirani (slika 2.17b).

Digitalna obdelava signalov

Digitalni signali se znotraj računalniških programov ustrezno obdelujejo (procesirajo). Tako npr. iz digitalnega signala odstranimo nezaželeno frekvenčno področje, ali pa npr. signalu, ki predstavlja temperaturo v nekem prostoru, preko regulacijskega algoritma določimo signal, ki deluje na mešalni sistem ogrevanja (prehod iz slike 2.17b v 2.17c).

Rekonstrukcija digitalnega signala v zvezni signal

Zlasti pri vodenju oz. regulacijah želimo digitalni signal znotraj računalnika spremeniti v fizikalno realni (odsekoma) zvezni signal na izhodu računalnika. Tipične enote, ki to opravljajo, so digitalno-analogni pretvorniki (D/A). V nekem

trenutku vzorčenja se mora digitalni vzorec vpisati v register D/A pretvornika in od takrat pa do naslednjega trenutka vzorčenja se vzorcu na izhodu računalnika priredi vrednost (tok ali napetost), ki ustreza digitalni vrednosti. Zato je tipičen signal na izhodu D/A pretvornika zvezni signal z odsekoma konstantnimi vrednostmi. Taki funkciji se v literaturi pravi tudi izhodni zatič, matematično pa tej funkciji pravimo zadrževalnik ničtega reda (ZOH - zero order hold) (slika 2.17d).

Postopek digitalne obdelave signalov ob predpostavljenih 3 bitnih pretvornikih z območjem 0-7V prikazuje slika 2.17.

Slika 2.18 prikazuje digitalni računalniški sistem za obdelavo signalov. Prav enaka shema velja pri računalniškem vodenju sistemov, ko računalnik dobi podatke o fizikalnih veličinah in izračuna signale za t.i. aktuatorje (npr. iz merjene temperature v prostoru določi signal za mešalni ventil).

Nesprotna in sprotna obdelava signalov

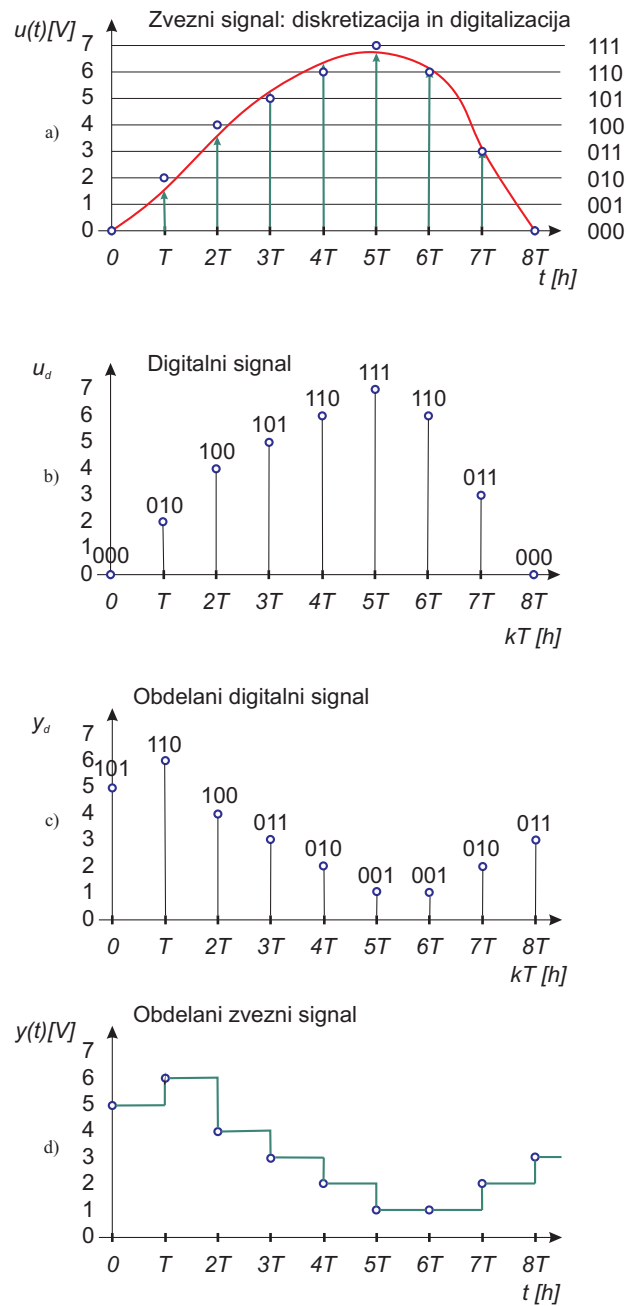
Nesprotna obdelava pomeni, da posnamemo določeno število vzorcev signala in nato obdelamo te vzorce. Tipično je npr. filtriranje, ki iz signala izloči nezaželeno frekvenco oz. frekvenčni pas. Za vodenje pa je bolj značilna sprotna obdelava, kar pomeni, da vsak vzorec sproti obdelamo in rekonstruiramo v zvezni signal. Ob tem običajno upoštevamo še nekaj preteklih vrednosti, ki jih je potrebno hraniti v pomnilniku. Postopek, ki jo prikazujeta sliki 2.17 in 2.18 predpostavlja, da za obdelavo ne potrebujemo časa, kar seveda v praksi ne drži. Pri sprotni obdelavi je nujno, da se vzorec obdela do naslednjega trenutka vzorčenja, zaželeno pa še bistveno prej, saj že majhne zakasnitve pri vodenju sistemov lahko zelo vplivajo na kvaliteto vodenja.

2.5.5 Temeljni signali

V nadaljevanju se bomo predvsem omejili na zvezne signale. Signali pogosto nastopajo v električnih vezjih pa tudi v mehanskih, hidravličnih in drugih sistemih.

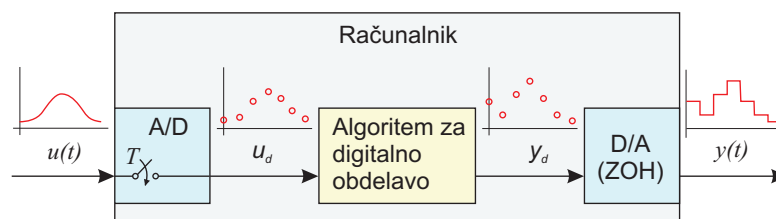
Enosmerni signali

Tako imenujemo signale, ki jih matematično opredelimo kot konstante (glej sliko 2.19)

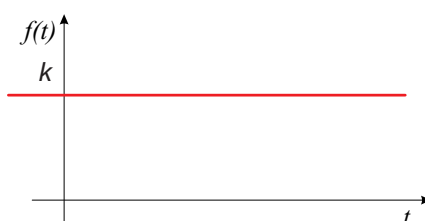


Slika 2.17: Digitalna obdelava signalov

$$f(t) = konst \quad (2.11)$$



Slika 2.18: Računalniški sistem za obdelavo signalov ali za vodenje sistemov



Slika 2.19: Konstanta

S temi signali predvsem opisujemo razmere v ustaljenem (stacionarnem) stanju, ko prehodni pojavi izzvenijo.

To je tudi edini signal, ki smo ga definirali za pozitivne in negativne čase. Pri vseh ostalih signalih smatramo, da imajo za negativne čase vrednost nič.

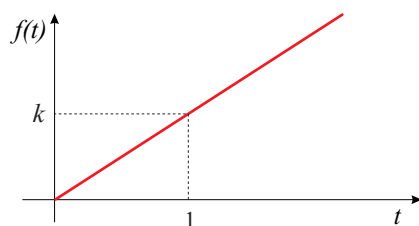
Linearno naraščajoči signal (rampa)

V električnih vezjih dobimo linearno naraščajoči signal, če priključimo idealni tokovni generator na kondenzator. Napetost takrat linearno narašča (glej sliko 2.20). Opisuje ga enačba

$$f(t) = \begin{cases} kt & ; t \geq 0 \\ 0 & ; t < 0 \end{cases} \quad (2.12)$$

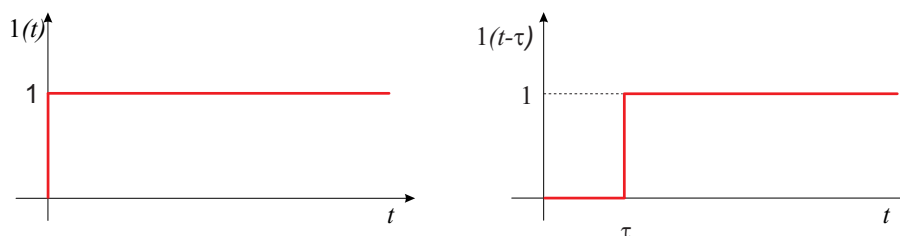
Enotin stopničasti signal

Enotina stopnica se od konstante loči po tem, da je za negativne čase vrednost enaka nič (glej sliko 2.21).



Slika 2.20: Linearno naraščajoči signal

$$f(t) = 1(t) = \begin{cases} 1 & ; t \geq 0 \\ 0 & ; t < 0 \end{cases} \quad (2.13)$$



Slika 2.21: Stopnica in zakasnjena stopnica

Slika 2.21 prikazuje tudi za čas τ zakasnjeno stopnico.

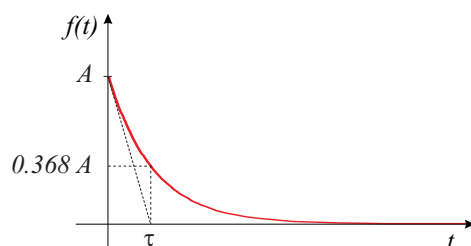
Enotina stopnica je torej nezvezna funkcija, ki v nekem trenutku preskoči iz 0 v 1. Je osnovni signal za modeliranje preklpov v signalih. Včasih namreč obravnavamo signale z modeli, ki veljajo na časovnem intervalu $-\infty$ do $+\infty$. Ker pa večino pojavov začnemo opazovati v trenutku nič in takrat izvedemo tudi razne vklope, lahko take signale modeliramo kot produkt signala z enotino stopnico.

Eksponentni signali

Pri proučevanju električnih vezij se kondenzator preko upora prazni po eksponentnem zakonu (slika 2.22)

$$f(t) = \begin{cases} Ae^{-at} = Ae^{-\frac{t}{\tau}} & ; t \geq 0 \\ 0 & ; t < 0 \end{cases} \quad (2.14)$$

A je začetna vrednost napetosti na kondenzatorju. Končna vrednost napetosti je

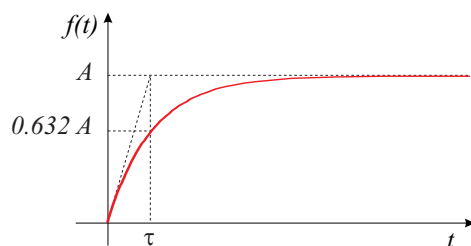


Slika 2.22: Eksponentna funkcija- upadanje

nič. $\tau = \frac{1}{a}$ je časovna konstanta. V času ene časovne konstante se kondenzator izprazni na 36.8% začetne vrednosti.

Če pa na RC člen v trenutku $t=0$ priklopimo enosmerno napetost A , pa se napetost na kondenzatorju polni po zakonitosti (slika 2.23)

$$f(t) = \begin{cases} A(1 - e^{-at}) = A(1 - e^{-\frac{t}{\tau}}) & ; t \geq 0 \\ 0 & ; t < 0 \end{cases} \quad (2.15)$$



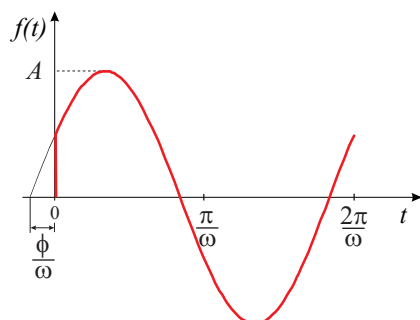
Slika 2.23: Eksponentna funkcija- naraščanje

Sinusni signali

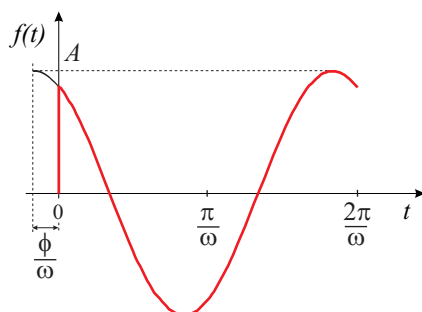
V električnih vezjih so vzbujalni signali ter signali na elementih v ustaljenem stanju pogosto v obliki sinusnih ali cosinusnih funkcij (sliki 2.24 in 2.25)

$$f(t) = \begin{cases} A \sin(\omega t + \phi) & ; t \geq 0 \\ 0 & ; t < 0 \end{cases} \quad (2.16)$$

$$f(t) = \begin{cases} A \cos(\omega t + \phi) & ; t \geq 0 \\ 0 & ; t < 0 \end{cases} \quad (2.17)$$



Slika 2.24: Sinusna funkcija



Slika 2.25: Kosinusna funkcija

A je amplituda sinusnega oz. kosinusnega signala, ω je krožna frekvenca [rd/s^{-1}], $\omega = 2\pi f = 2\pi/T$, f je frekvenca [Hz], T pa perioda [s]. ϕ je fazni zasuk [rd].

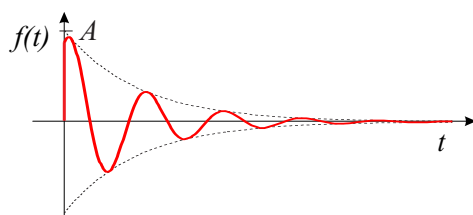
Signal dušenega nihanja

Če imamo v električnem vezju upor, tuljavo in kondenzator (ali v mehanskih sistemih maso, vzmet, dušilnik), dobimo v prehodnih pojavih pogosto signal dušenega nihanja. Pri praznjenju kondenzatorja ima napetost obliko (slika 2.26)

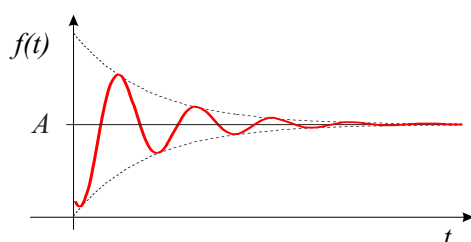
$$f(t) = \begin{cases} Ae^{-at}\cos(\omega t + \phi) & ; t \geq 0 \\ 0 & ; t < 0 \end{cases} \quad (2.18)$$

Pri polnjenju kondenzatorja pa ima napetost obliko (slika 2.27)

$$f(t) = \begin{cases} A(1 - e^{-at}\cos(\omega t + \phi)) & ; t \geq 0 \\ 0 & ; t < 0 \end{cases} \quad (2.19)$$



Slika 2.26: Dušena kosinusna funkcija - upadanje



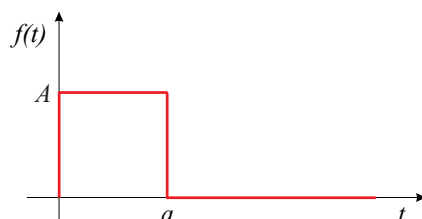
Slika 2.27: Dušena kosinusna funkcija - naraščanje

Konstanta a , ki je obratna vrednost časovne konstante, je tudi povezana z dušenjem. Če je nič, ostanejo čisti trigonometrični izrazi. Čim večja je vrednost za a , manj izrazito je nihajoče vedenje.

Pravokotni impulz

To je prav tako pogosta funkcija zlasti uporabna pri testiranju vezij (slika 2.28)

$$f(t) = \begin{cases} A & ; 0 \leq t < a \\ 0 & ; a \leq t < \infty \end{cases} \quad (2.20)$$



Slika 2.28: Pravokotni impulz

Enotin pravokotni impulz

Pravokotni impulz je enotin, če je njegova ploščina enaka 1. Torej mora veljati

$$\int_0^{\infty} f(t) dt = 1 \quad (2.21)$$

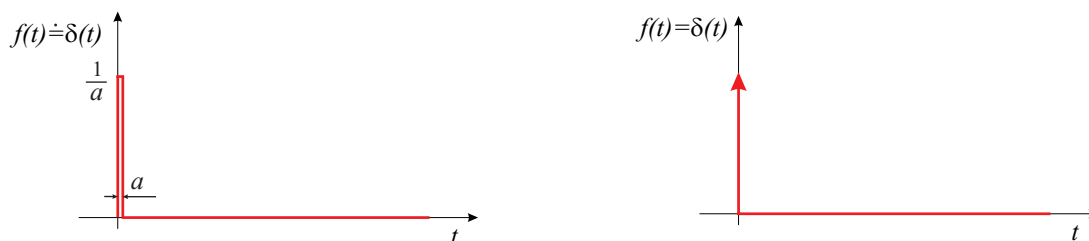
Veljati mora torej odvisnost

$$a = \frac{1}{A} \quad (2.22)$$

Dirac-ov impulz

Dirac-ov impulz (delta impulz) nastane iz enotnega pravokotnega impulza, če limitiramo a proti nič (slika 2.29). Takrat gre $A = \frac{1}{a}$ čez vse meje.

$$f(t) = \delta(t) = \begin{cases} \infty & ; t = 0 \\ 0 & ; t \neq 0 \end{cases} \quad (2.23)$$



Slika 2.29: Dirac-ov impulz

Bistveno je torej, da je ploščina Diracovega impulza enaka 1.

$$\int_0^{\infty} \delta(t) dt = 1 \quad (2.24)$$

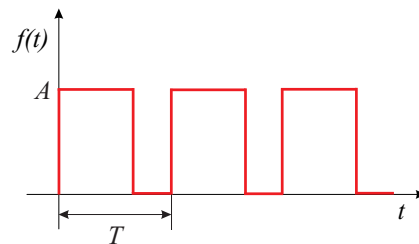
Dirac-ov impulz je povsem teoretska forma, vendar ima zaradi nekaterih ugodnih lastnosti pomembno mesto pri analizi in načrtovanju dinamičnih sistemov. Z nekaj matematike pa lahko pokažemo, da dobimo Dirac-ov impulz tudi z odvajanjem stopničaste funkcije.

$$\delta(t) = \frac{d 1(t)}{dt} \quad (2.25)$$

Vlak impulzov

Iz pravokotnega impulza lahko zgradimo tudi periodično funkcijo, ki jo imenujemo vlak impulzov (slika 2.30). Veljati mora seveda osnovna zahteva

$$f(t + T) = f(t) \quad (2.26)$$



Slika 2.30: Vlak impulzov

2.5.6 Uvod v spektralno analizo

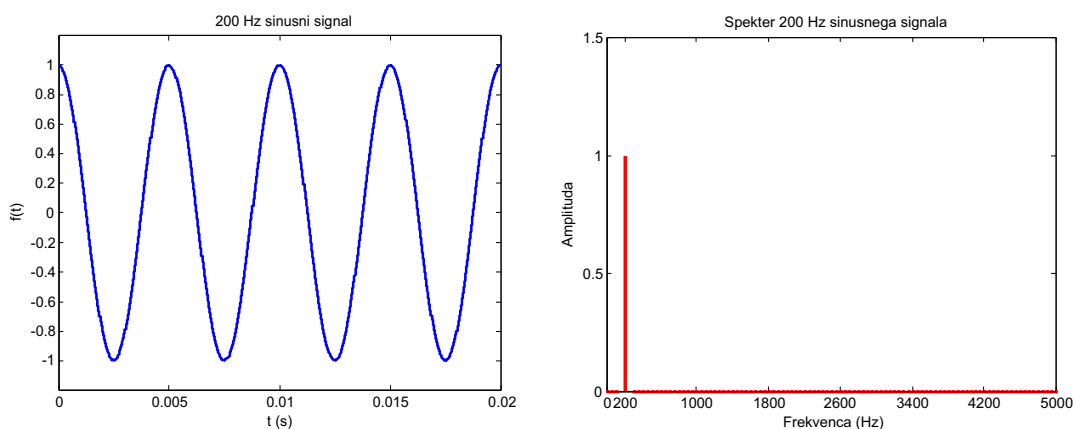
Do sedaj smo signale obravnavali v časovnem prostoru, kar pomeni, da smo jih predstavljali kot časovne funkcije. Spektralna analiza pa pove, iz katerih frekvenc je sestavljen nek signal, torej iz kakšnih trigonometričnih signalov bi nek signal lahko sestavili (Mlakar, 2002, Cha, Molinder, 2006). Spekter je zlasti enostaven pri periodičnih časovnih signalih, saj je diskreten (črtast). Če je frekvenca periodičnega signala ω_0 , potem spekter v splošnem vsebuje enosmerno komponento in frekvence $\omega_0, 2\omega_0, 3\omega_0, \dots$, torej osnovno harmonsko in višje harmonske. Slika 2.31 prikazuje 200Hz kosinusni signal

$$f(t) = \cos[2\pi(200t)] \quad (2.27)$$

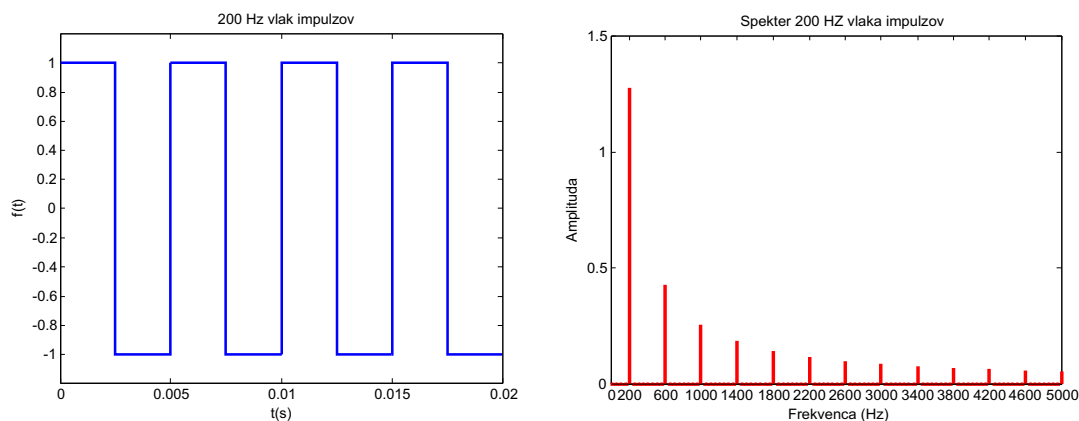
in njegov spekter. Razvidna je prisotnost ene spektralne komponente frekvence 200Hz in amplitude 1.

Slika 2.32 prikazuje periodični vlak impulzov frekvence 200Hz in njegov spekter.

Vlak impulzov se spreminja med $+1$ in -1 s periodo $T = 0.005\text{s}$. Spekter vsebuje komponente pri $200\text{Hz}, 600\text{Hz}, 1000\text{Hz}, 1400\text{Hz}, \dots$ Najmočnejša je torej



Slika 2.31: 200 Hz signal in njegov spekter



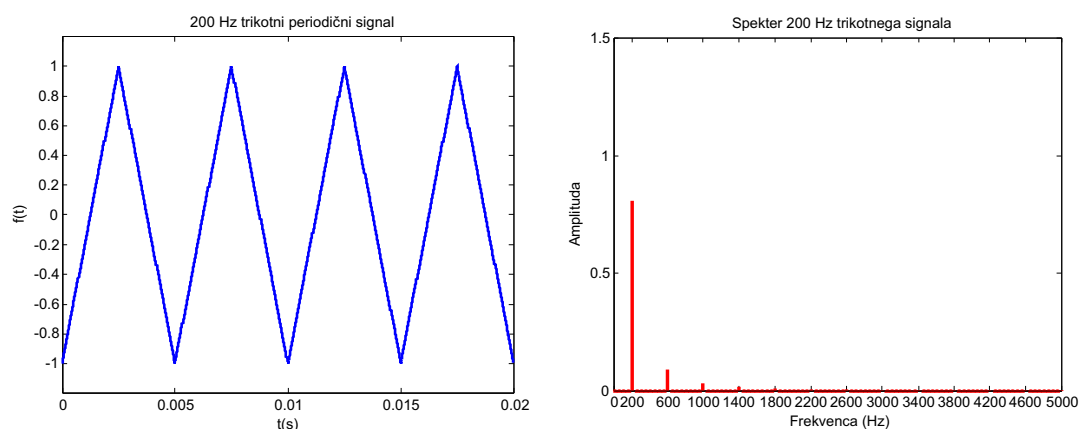
Slika 2.32: 200 Hz vlak impulzov in spekter

frekvenca 200 Hz , nato 3 harmonska 600 Hz , pa peta harmonska 1000 Hz itd. Spekter torej vsebuje le lihe harmonske komponente.

Slika 2.33 pa prikazuje trikotni periodični signal frekvence 200 Hz in njegov spekter.

V tem primeru tudi nastopajo lihe harmonske komponente, je pa očitno, da višje harmonske komponente hitro upadajo.

Slika 2.34 pa prikazuje dvo-kanalni audio signal, ki je vzorčen s 44.1 kHz , kar zagotavlja ustrezno CD kvaliteto in njegov frekvenčni spekter. Oko težko razbere, kaj vsebuje časovni signal. Če pa ga pretvorimo v akustični signal, tako da ga pošljemo na zvočnik, pa prepoznamo Beethovnovo 5. simfonijo.



Slika 2.33: 200 Hz trikotni periodični signal in spekter

Signale lahko torej predstavljamo in analiziramo na različne načine: v časovnem prostoru, v frekvenčnem prostoru, kot akustične signale, ... Včasih je primernejša neka predstavitev, včasih pa kaka druga.

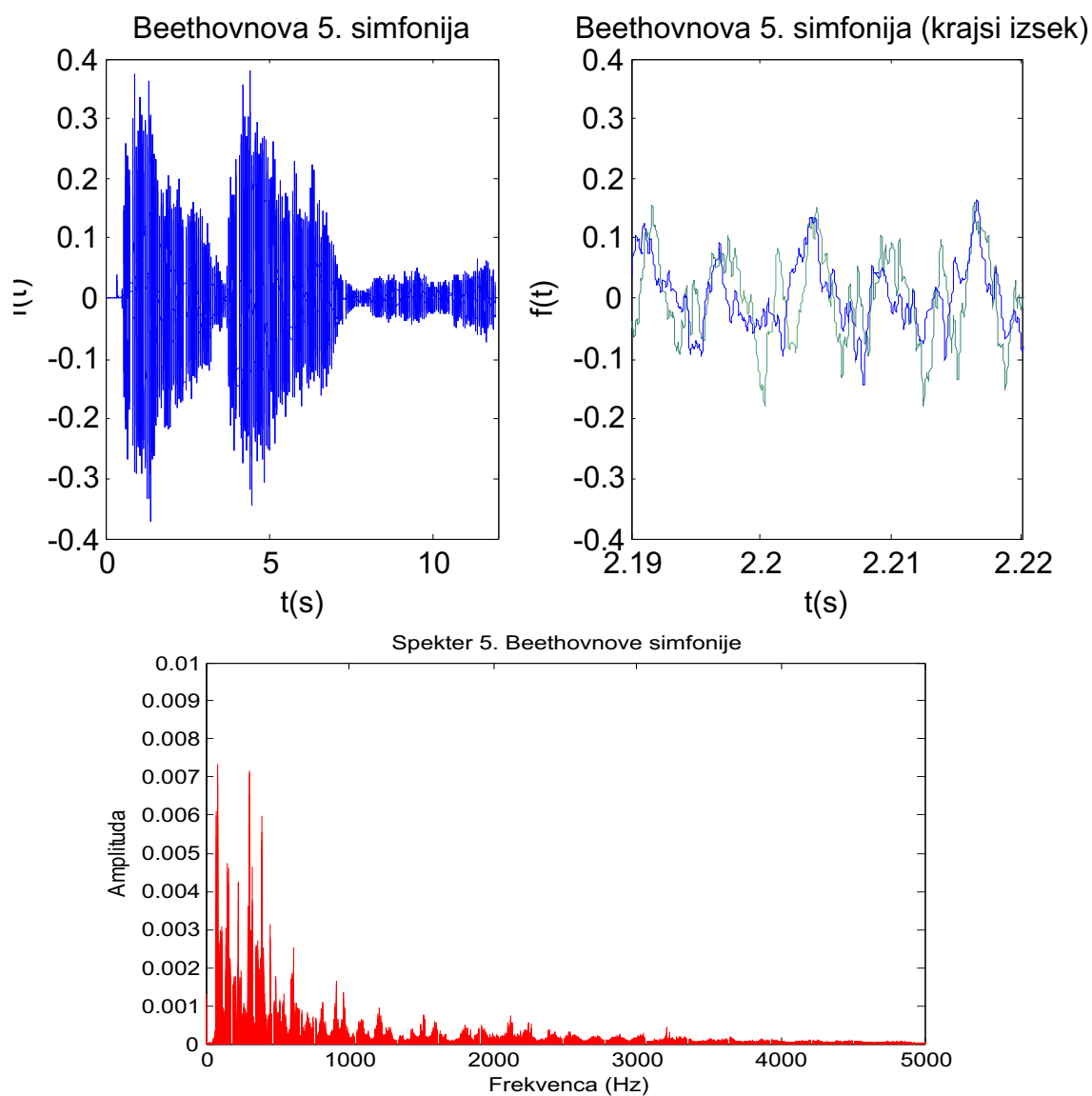
2.5.7 Spektralna analiza s Fourier-jevo vrsto

Pri spektralni analizi je zelo veliko pristopov, ki se ločijo predvsem glede na to

- ali je funkcija periodična ali neperiodična,
- ali je funkcija zvezna ali diskretna.

Osnovni pristop je spektralna analiza zveznih periodičnih časovnih funkcij, ki ga imenujemo Fourier-jeva vrsta (Mihelič, 2010, Mlakar, 2002). Namesto da govorimo o spektralni analizi, kar pomeni, da nas zanima, katere frekvence nastopajo v signalu, lahko govorimo tudi o postopku, kako aproksimirati periodično funkcijo s pomočjo vsote sinusnih (in kosinusnih) signalov. Fourier-jeva vrsta pomeni, da izrazimo periodično funkcijo z vsoto sinusnih in kosinusnih signalov na način, da se originalna funkcija in vsota čim bolj ujemata glede na kriterij najmanjših kvadratov. Fourier-jeva analiza velja torej za periodične funkcije, ko velja

$$f(t) = f(t + T_0) \quad \text{za vse } t \quad (2.28)$$



Slika 2.34: Dvokanalni audio signal - 12 s opazovanja in 30 ms izsek, spodaj frekvenčni spekter

T_0 imenujemo periodo signala.

Funkcijo $f(t)$ torej lahko sestavimo iz vsote trigonometričnih signalov. V posplošeni obliki uporabimo osnovni gradnik v eksponentnem zapisu (Fourierjeva eksponentna vrsta).

$$\phi_k(t) = e^{jk\omega_0 t} = e^{jk\frac{2\pi}{T_0}t} = \cos\left(k\frac{2\pi}{T_0}t\right) + j \sin\left(k\frac{2\pi}{T_0}t\right)$$

kjer je $-\infty < k < +\infty$. Funkcijo $f(t)$ sestavimo kot

$$f(t) = \sum_{k=-\infty}^{+\infty} c_k \phi_k(t) = \sum_{k=-\infty}^{+\infty} c_k e^{jk\omega_0 t} \quad (2.29)$$

$\omega_0 = \frac{2\pi}{T_0} = 2\pi f_0$. Izpeljava, ki vključuje metodo najmanjših kvadratov, nas pripelje do enačbe za izračun Fourier-jevih koeficientov c_k

$$c_k = \frac{1}{T_0} \int_{T_0} f(t) e^{-jk\omega_0 t} dt = \frac{1}{T_0} \int_{T_0} f(t) e^{-j2\pi k f_0 t} dt \quad (2.30)$$

T_0 pri integralu pomeni, da lahko integracijo naredimo kjerkoli na intervalu T_0 (od $-\infty$ do $+\infty$).

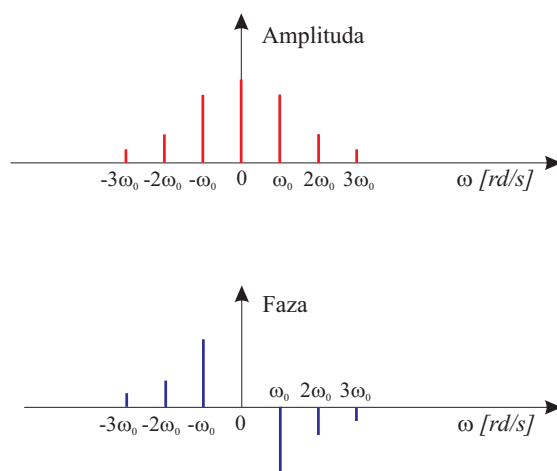
Glede na enačbo 2.29 koeficienti c_k pomenijo uteži, preko katerih so posamezne frekvence zastopane v signalu. Govorimo o frekvenčnem spektru signala. Ker je c_k v splošnem kompleksen, je tudi spekter v splošnem kompleksen: govorimo o amplitudnem in faznem spektru

$$\begin{aligned} c_k &= |c_k| e^{j\Theta} \\ |c_k| &= \sqrt{\operatorname{Re}^2(c_k) + \operatorname{Im}^2(c_k)} \\ \Theta &= \arctan\left[\frac{\operatorname{Im}(c_k)}{\operatorname{Re}(c_k)}\right] \end{aligned} \quad (2.31)$$

Pri periodičnih signalih je spekter vedno diskreten (črtast). Komponente se pojavijo pri frekvencah 0 (enosmerna komponenta), $\pm\omega_0$, $\pm 2\omega_0$, ... in pripadajo indeksom $k = 0, \pm 1, \pm 2, \dots$, velikosti pa so določene z vrednostmi $|c_k|$. Običajno z naraščajočim k ali ω spekter hitro upada. Slika 2.35 prikazuje primer spektra. Amplitudni del je vedno soda, fazni del pa liha funkcija.

Vpeljali smo kompleksno (eksponentno) Fourier-jevo vrsto, ki pravzaprav združuje sinusne in kosinusne signale. Lahko pa uporabimo tudi osnovno Fourier-jevo vrsto, ki sestavi signal iz sinusnih in kosinusnih funkcij

$$f(t) = a_0 + 2 \sum_{k=1}^{\infty} [a_k \cos(k\omega_0 t) + b_k \sin(k\omega_0 t)] \quad (2.32)$$



Slika 2.35: Primer kompleksnega spektra

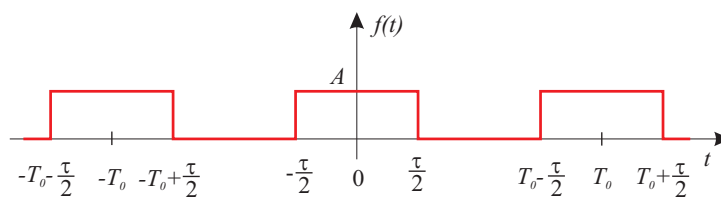
kjer so koeficienti a_k in b_k

$$a_k = \frac{1}{T_0} \int_{T_0} f(t) \cos(k\omega_0 t) dt \quad (2.33)$$

$$b_k = \frac{1}{T_0} \int_{T_0} f(t) \sin(k\omega_0 t) dt \quad (2.34)$$

Primer 1: Fourier-jeva eksponentna vrsta vlaka impulzov

Slika 2.36 prikazuje periodični signal vlak impulzov.



Slika 2.36: Vlak impulzov

$$c_0 = \frac{1}{T_0} \int_{T_0} f(t) dt = \frac{1}{T_0} \int_{-\frac{\tau}{2}}^{+\frac{\tau}{2}} A dt = \frac{At}{T_0} \Big|_{-\frac{\tau}{2}}^{+\frac{\tau}{2}} = A \frac{\tau}{T_0}$$

$$\begin{aligned}
c_k &= \frac{1}{T_0} \int_{T_0} f(t) e^{-jk\omega_0 t} dt = \frac{1}{T_0} \int_{-\frac{\tau}{2}}^{+\frac{\tau}{2}} A e^{-jk\omega_0 t} dt = \\
&= \frac{A}{T_0 (-jk\omega_0)} e^{-jk\omega_0 t} \Big|_{-\frac{\tau}{2}}^{+\frac{\tau}{2}} = \\
&= \frac{A}{-jkT_0\omega_0} \left(e^{-jk\omega_0 \frac{\tau}{2}} - e^{jk\omega_0 \frac{\tau}{2}} \right) = \\
&= \frac{2A}{kT_0\omega_0} \frac{e^{jk\omega_0 \frac{\tau}{2}} - e^{-jk\omega_0 \frac{\tau}{2}}}{2j} = \\
&= \frac{2A}{kT_0\omega_0} \sin k\omega_0 \frac{\tau}{2} = \\
&= \frac{2A}{kT_0 \frac{2\pi}{T_0}} \sin k \frac{2\pi}{T_0} \frac{\tau}{2} = \frac{A}{k\pi} \sin \left(k\pi \frac{\tau}{T_0} \right) = \\
&= \frac{k\pi \frac{\tau}{T_0} A \sin \left(k\pi \frac{\tau}{T_0} \right)}{k\pi} = \frac{A\tau}{T_0} \frac{\sin \left(k\pi \frac{\tau}{T_0} \right)}{k\pi \frac{\tau}{T_0}}
\end{aligned}$$

Zapis poenostavimo, če definiramo $d = \frac{\tau}{T_0}$, kjer je d razmerje med časom, ko je vrednost signala A in časom periode (delavnost signala - duty cycle).

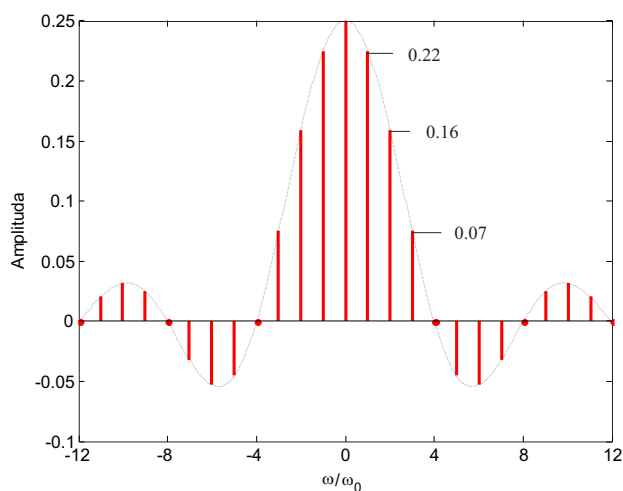
$$\begin{aligned}
c_0 &= Ad \\
c_k &= Ad \frac{\sin k\pi d}{k\pi d} = Ad \operatorname{sinc}(kd)
\end{aligned} \tag{2.35}$$

Slika 2.37 prikazuje spekter vlaka impulzov pri $A = 1$ in $d = \frac{1}{4}$.

Signal torej vsebuje enosmerno komponento ter frekvence ω_0 (osnovna frekvenca), $2\omega_0, 3\omega_0, \dots$ (višje harmonske). Frekvence $4\omega_0, 8\omega_0, \dots$ ne nastopajo. Negativne frekvence nimajo nekega fizikalnega pomena, pač pa se moramo zavedati, da vsebujejo polovico 'moči' (npr. pri frekvenci ω_0 ima komponenta pri frekvenci ω_0 polovico moči in pri frekvenci $-\omega_0$ polovico moči).

Opomba:

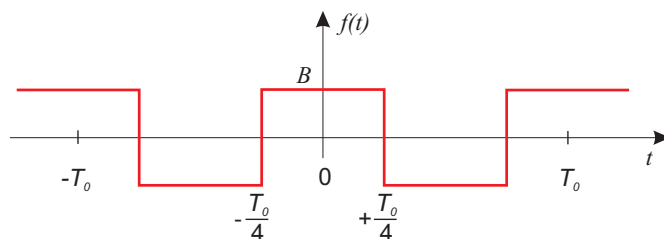
Funkcija $\operatorname{sinc}(x)$ ima dve različni definiciji: v matematičnih priročnikih je ponavadi $y = \operatorname{sinc}(x) = \frac{\sin x}{x}$, v okolju Matlab pa $y = \operatorname{sinc}(x) = \frac{\sin \pi x}{\pi x}$.



Slika 2.37: Spekter vlaka impulzov

Primer 2: Fourier-jeva eksponentna vrsta vlaka impulzov brez enosmerne komponente

Signal prikazuje slika 2.38.

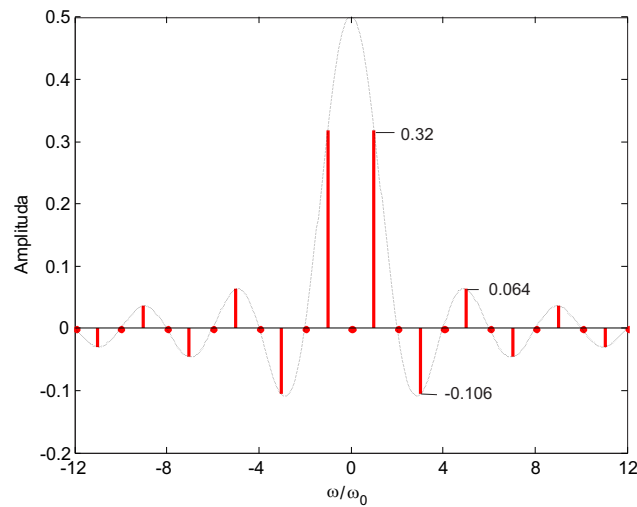


Slika 2.38: Vlaka impulzov brez enosmerne komponente

Signal očitno nima enosmerne komponente. Glede na prejšnji primer je amplituda $B = \frac{A}{2}$, $d = \frac{1}{2}$. Ne da bi izpeljali rezultat, lahko napišemo rešitev

$$\begin{aligned} c_0 &= 0 \\ c_k &= B \frac{\sin\left(k\frac{\pi}{2}\right)}{k\frac{\pi}{2}} = B \operatorname{sinc}\left(\frac{k}{2}\right) \end{aligned} \quad (2.36)$$

Spekter prikazuje slika 2.39 za $B = \frac{1}{2}$ in $d = \frac{1}{2}$.



Slika 2.39: Spekter vlaka impulzov brez enosmerne komponente

Opazimo, da ni enosmerne komponente, prav tako ni frekvenc $2\omega_0$, $4\omega_0$, $6\omega_0$, ...

Primer 3: Sinusoidni signal s fazno premaknitvijo

Imamo signal

$$f(t) = A \cos(\omega_0 t + \theta) \quad (2.37)$$

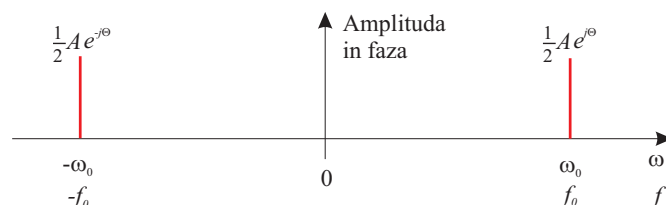
Lahko bi izračunali Fourier-jeve koeficiente po enačbi 2.30, toda v tem primeru hitreje pridemo do rezultata z upoštevanjem Euler-jeve formule $\cos x = \frac{e^{jx} + e^{-jx}}{2}$. Torej je

$$\begin{aligned} f(t) &= \frac{A}{2} [e^{j(\omega_0 t + \theta)} + e^{-j(\omega_0 t + \theta)}] = \\ &= \frac{1}{2} A e^{-j\theta} e^{-j\omega_0 t} + \frac{1}{2} A e^{j\theta} e^{+j\omega_0 t} \end{aligned} \quad (2.38)$$

Iz enačbe 2.38 določimo

$$\begin{aligned} c_0 &= 0 \\ c_{-1} &= \frac{A e^{-j\theta}}{2} \\ c_1 &= \frac{A e^{j\theta}}{2} \end{aligned} \quad (2.39)$$

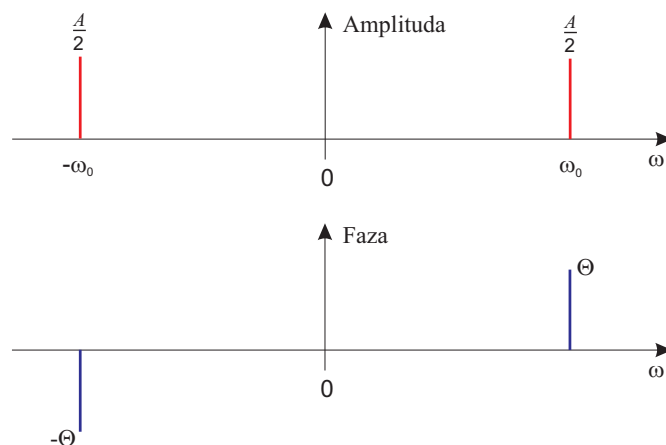
Vsi ostali koeficienti so nič. Spekter ima torej le dve spektralni komponenti, ki sta kompleksni in skupaj sestavljata signal. Prikazuje ga slika 2.40.



Slika 2.40: Kompleksni spekter fazno zamaknjene kosinusnega signala

Spekter je kompleksen zaradi faznega zamika kosinusnega signala. V primeru, da bi bil $\Theta = 0$, bi imeli dve realni spektralni komponenti.

Predstavitev spektra v tej obliki ni edina možna. Ta oblika namreč v enem diagramu nosi informacijo o amplitudi in fazi posamezne komponente. Ta zapis je zlasti v primeru večjega števila spektralnih komponent nepregleden. Zato pogosto uporabimo dva diagrama za posredovanje enake informacije: posebej amplitudni diagram in posebej fazni diagram. Za zgornji primer sta oba diagrama prikazana v sliki 2.41.



Slika 2.41: Amplitudni in fazni spekter signala $f(t) = A \cos(\omega_0 t + \theta)$

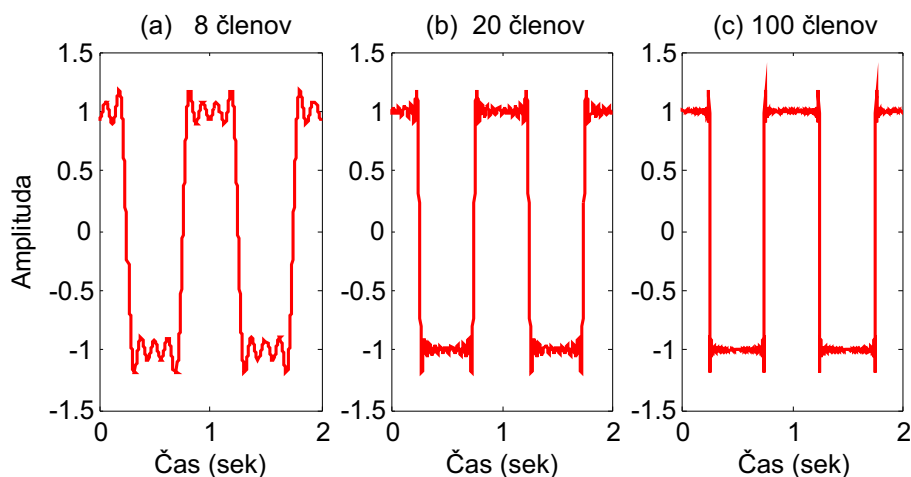
2.5.8 Napake pri uporabi Fourier-jeve vrste

Teoretično je potrebno pri izražavi signala $f(t)$ uporabiti neskončno členov Fourier-jeve vrste. V praksi seveda lahko uporabimo le končno členov (Cha, Molinder, 2006).

Če uporabimo $2K + 1$ členov se izražava glasi

$$f(t) \doteq \sum_{k=-K}^K c_k \phi_k(t) = \sum_{k=-K}^K c_k e^{jk\omega_0 t} \quad (2.40)$$

Uspešnost aproksimacije lahko preverimo s programom v okolju Matlab. Analiziramo signal vlak impulzov brez enosmerne komponente (slika 2.38): $B = 1$, $T_0 = 1$ in $d = \frac{1}{2}$. Slika 2.42 prikazuje aproksimacije z uporabo $K = 8$, 20 in 100 členov.



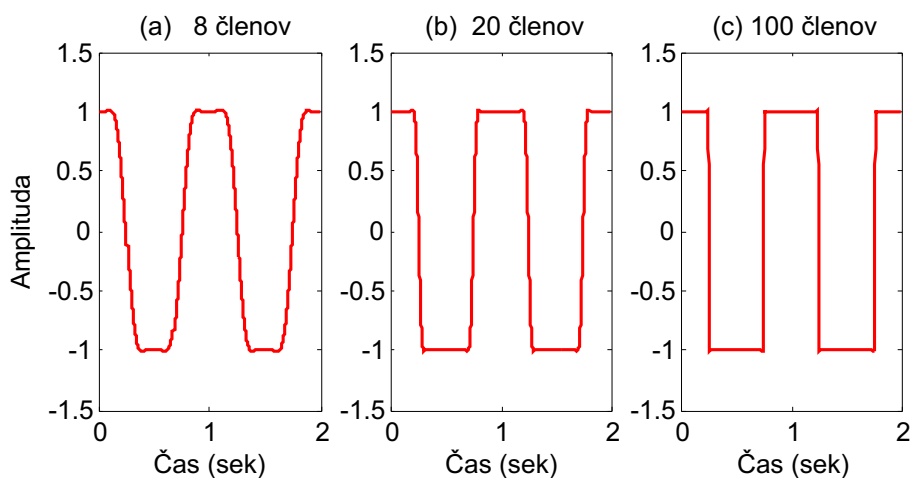
Slika 2.42: Izražava vlaka impulzov s končnim številom členov

Z večanjem števila členov se rezultat sicer izboljšuje, vendar napaka pri prehodih vlaka impulzov nikoli ne izgine. Pojav se imenuje Gibbs-ov pojav in je posledica optimizacije z metodo najmanjših kvadratov. Zato obstajajo metode, ki zgladijo poteke v nezveznostih, čeprav ne dajejo več optimalnih rezultatov v smislu najmanjših kvadratov. Slika 2.43 prikazuje rezultate po eni od metod glajenja.

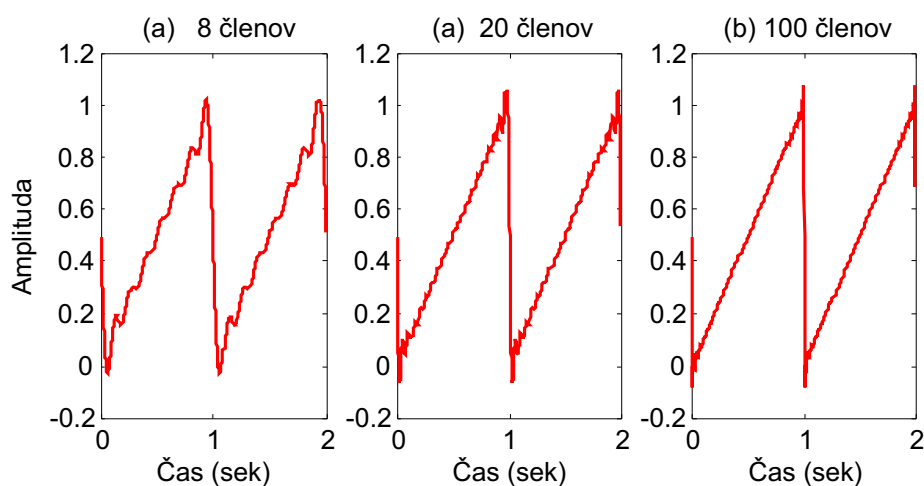
Slika 2.44 pa prikazuje aproksimacijo 'žagastega' signala s $K = 8$, 20 in 100 členi. Tudi tu opazimo Gibbs-ov pojav v točkah nezveznosti.

Pri izražavi je potrebno najti kompromis med natančnostjo in porabo računalniškega časa. Za praktično uporabnost zadošča običajno majhno število členov.

Zlasti kadar uporabljamo Fourier-jevo analizo za spektralno analizo signala, se običajno zadovoljimo z manjšim številom spektralnih komponent.



Slika 2.43: Uporaba metode glajenja



Slika 2.44: Izražava žagastega signala s Fourier-jevo vrsto s končnim številom členov

Zaključki

Kompleksni spekter periodičnega zveznega signala je sestavljen iz spektralnih harmonskih komponent pri pozitivnih in negativnih frekvencah. Zato je diskretni oz. črtasti. Koefficient z indeksom 0 predstavlja enosmerno komponento signala. Amplitudni spekter predstavlja amplitude (absolutne vrednosti) spektralnih harmonskih komponent. Fazni spekter je množica faz, ki pripadajo posameznim spektralnim harmonskim komponentam. Kompleksna Fourier-jeva vrsta se torej loči od osnovne Fourier-jeve vrste z razširitvijo tudi na negativne frekvence.

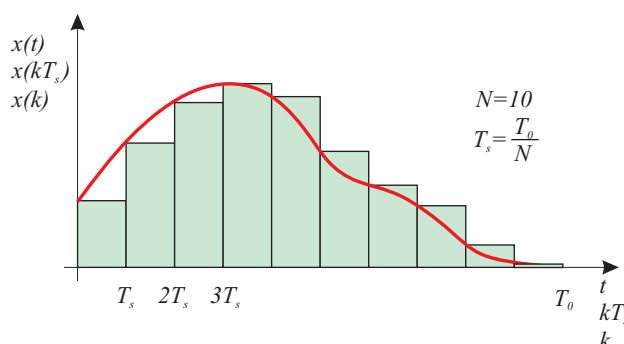
Velja: Amplitudni spekter je soda, fazni pa liha funkcija frekvence. Negativne frekvence so zgolj matematični formalizem, ki ga prinese eksponentna vrsta. Posamezno harmonsko komponento, ki jo v elektrotehniki interpretiramo kot izmenični signal, sestavljata po enačbi 2.38 kompleksni eksponentni funkciji pri negativni in pri pozitivni frekvenci (pri $-\omega_0$ in pri $+\omega_0$, v splošnem pri $-k\omega_0$ in $+k\omega_0$). Obe komponenti v vsoti generirata realni signal z amplitudo A in fazo Θ ($f(t) = A \cos(\omega_0 t + \theta)$).

Graf amplitudnega spektra kompleksne Fourier-jeve vrste je enak grafu polovičnih vrednosti amplitudnega spektra, ki ga dobimo s klasično Fourier-jevo transformacijo in kaže spekter le za fizikalno realne pozitivne frekvence. V primerih na začetku (slike 2.31, 2.32, 2.33) smo v programskem okolju MATLAB sicer uporabili orodja, ki zgradijo spekter za negativne in pozitivne frekvence, narisali pa smo le del za pozitivne frekvence tako, da smo te komponente pomnožili z 2 (to ne velja za enosmerno komponento).

2.5.9 Numerično računanje spektrov - diskretna Fourier-jeva transformacija

Fourier-jeva eksponentna vrsta je le osnovni pristop pri obdelavi signalov v smislu določevanja spektrov. Ima pa nekaj pomanjkljivosti. Deluje le za zvezne periodične signale. Z zvezno integracijo (enačba 2.30) dobimo diskretni (črtasti) spekter v obliki neperiodične funkcije. Frekvenčno območje je običajno omejeno. Dandanes pa spektralno analizo skoraj brez izjeme opravlja računalnik (Cha, Molinder, 2006). Zaradi omejenih tehnoloških zmožnosti ne more delati z zveznimi signali, ampak le z njihovimi vzorci, torej z diskretnimi signali. Ker tudi predpostavljamo, da imajo računalniki običajno vsaj 16 bitne A/D pretvornike, lahko tudi zanemarimo napako zaradi kvantizacije oziroma digitalizacije. Enačbo 2.30 je potrebno numerično integrirati, potem ko zvezni signal nadomestimo z njegovimi vzorci. Postopek prikazuje slika 2.45.

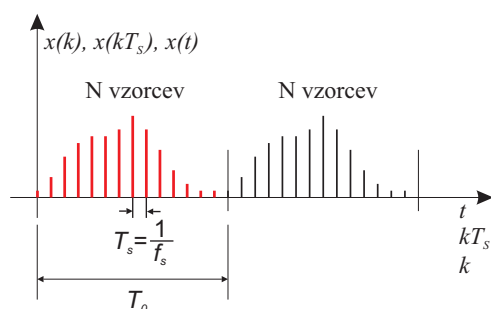
Pomemben podatek je čas opazovanja signala T_0 . Ni nujno, da je $x(t)$ periodičen. Vendar sam postopek vendarle predpostavlja, da je opazovani interval T_0 pravzaprav ena perioda opazovanja. Čeprav formalizem predpostavlja periodičnost, naj nas to ne moti, saj se vedno lahko osredotočimo le na osnovno periodo. Dogodi pa se še ena pomembna sprememba. Ker prevedemo zvezni signal $x(t)$ v diskretni signal $x(kT_s)$ ali $x(k)$, tudi frekvenčni spekter postane periodičen, seveda pa je še vedno diskreten. Torej je pri digitalni obdelavi pravilo, da sta tako časovna funkcija kot njen spekter diskretna in periodična. Periodičnost se vedno trans-



Slika 2.45: Postopek numerične integracije

formira v diskretnost, diskretnost pa v periodičnost. Na našem nivoju obdelave bomo praviloma ignorirali nadaljnje periode tako časovnega kot frekvenčnega signala. Časovno zvezni signal bomo skušali čim hitreje vzorčiti, ker bomo s tem bliže našemu razumevanju s Fourier-jevo eksponentno vrsto.

Izpeljava nas pripelje do diskretne Fourier-jeve transformacije (DFT), ki N vzorcev časovne funkcije $x(t)$, ki jo zapišemo z $x(k)$ (glej sliko 2.46)

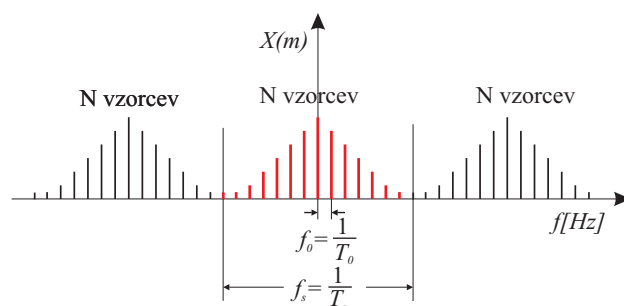
Slika 2.46: Časovna funkcija $x(k)$

pretvori v N vzorcev frekvenčne funkcije $X(m)$ po enačbi

$$X(m) = \frac{1}{N} \sum_{k=0}^{N-1} x(k) e^{-j \frac{2\pi}{N} km} \quad m = 0, 1, 2, \dots, N-1 \quad (2.41)$$

Slika 2.47 prikazuje značilen spekter.

Veljajo vse lastnosti, kot pri spektru, ki smo ga dobili s kompleksno Fourier-jevo transformacijo. Spekter je v splošnem kompleksen in ga bolj nazorno



Slika 2.47: Spekter $X(m)$, ki ga dobimo z diskretno Fourier-jevo transformacijo

prikažemo v dveh diagramih - v diagramu za absolutno vrednost in v diagramu za fazo. Frekvenčna razlika med dvema sosednima spektralnima komponentama je obratna vrednost časa opazovanja oziroma predpostavljene periode zveznega signala T_0 . Frekvenčna širina ene periode frekvenčnega spektra pa je obratna vrednost periode vzorčenja. Če zagotovimo, da je frekvenca vzorčenja nekajkrat višja od najvišjih frekvenc v signalu, potem frekvenčna širina ene periode lepo prikaže vse frekvence.

Obstaja pa tudi obratna transformacija, ki jo imenujemo inverzna diskretna Fourier-jeva transformacija (IDFT). Le-ta iz ene periode frekvenčne funkcije $X(m)$ vrne časovno diskretno funkcijo $x(k)$ po enačbi

$$x(k) = \sum_{m=0}^{N-1} X(m) e^{j \frac{2\pi}{N} km} \quad k = 0, 1, 2, \dots, N-1 \quad (2.42)$$

Enačbi 2.41 in 2.42 sta si zelo podobni in se razlikujeta le po členu $\frac{1}{N}$ in predznaku v eksponentu. Slika 2.46 prikazuje periodično časovno diskretno funkcijo. Vendar se običajno ukvarjamo le z osnovno periodo. Pomembna je odvisnost

$$N = \frac{1}{T_s f_0} = f_s T_0 \quad (2.43)$$

Število vzorcev je torej obratna vrednost produkta resolucij časovne in frekvenčne funkcije oz. sorazmerna produktu frekvence vzorčenja in časa opazovanja.

DFT in IDFT sta torej enačbi 2.41 in 2.42 in z določenimi izboljšavami predvsem v smislu hitrejšega izvajanja programiranimi tudi v okolju Matlab. Postopka

se imenujeta hitra Fourier-jeva transformacija (funkcija `fft`) in inverzna hitra Fourier-jeva transformacija (funkcija `ifft`). Ker zahteva uporaba funkcij nekaj več izkušenj, bomo uporabljali dve nadgrajeni funkciji, ki kličeta omenjeni Matlab-ovi funkciji (Cha, Molinder, 2006).

DFT ali bolj splošno spektralno analizo časovno diskretne (vzorčene) časovne funkcije bomo izvajali s funkcijo `fdomain`, ki jo kličemo:

$$[X, f] = \text{fdomain}(x, fs)$$

Funkcija iz N vzorcev časovne funkcije izračuna N vzorcev frekvenčne funkcije. Ker gre v Matlabu indeks od ena naprej, moramo upoštevati premik za ena glede na enačbo 2.41.

Vhodna parametra:

`x` ... vektor N elementov, ki opisuje vzorčeni časovni signal

`fs` ... frekvenca vzorčenja

Izhodna parametra:

`X` ... vektor N elementov, ki opisuje frekvenčno funkcijo

`f` ... vektor N elementov, ki vsebuje frekvence, katerim pripadajo elementi vektorja `X`

Uporaba je uporabniško prijazna. Spekter lahko v Matlabu narišemo z ukazom `stem(f, X)`

V kolikor pa želimo kakorkoli obdelati spekter (npr. izločiti določene frekvence), moramo natančneje razumeti povezavo med indeksom vektorja `X`, ki gre od 1 do N in frekvenco. Tu prihaja celo do razlike pri sodem in lihem številu N .

Če je N sodo število, velja tabela 2.1, če pa je N liho število, pa velja tabela 2.2.

Tabela 2.1: Relacija med indeksom in frekvenco, če je N sodo število

Indeks	1	2	...	$\frac{N}{2} - 1$	$\frac{N}{2}$	$\frac{N}{2} + 1$	$\frac{N}{2} + 2$...	N
Frekvenca ω_0*	$-\frac{N}{2}$	$-\frac{N}{2} + 1$...	-2	-1	0 enosm.	1	...	$\frac{N}{2} - 1$

Tabela 2.2: Relacija med indeksom in frekvenco, če je N liho število

Indeks	1	2	...	$\frac{N-1}{2}$	$\frac{N+1}{2}$	$\frac{N+3}{2}$...	N
Frekvenca ω_0^*	$-\frac{N-1}{2}$	$-\frac{N-1}{2} + 1$...	-1	0 enosm.	1	...	$\frac{N-1}{2}$

Primer: Če želimo pri sodem številu vzorcev izločiti iz spektra frekvenco ω_0 , moramo postaviti na nič komponenti vektorja \mathbf{X} z indeksoma $\frac{N}{2}$ in $\frac{N}{2} + 2$ ($N=1000$, $X(500)=0$, $X(502)=0$, $X(501)$ je enosmerna komponenta).

IDFT pa naredimo z ukazom `tdomain`:

```
[x,t]=tdomain(X,fs)
```

Funkcija iz N komponent frekvenčne funkcije izračuna N vzorcev časovne funkcije. Ker gre v Matlabu indeks od ena naprej, moramo upoštevati premik za ena glede na enačbo 2.42.

Vhodna parametra:

\mathbf{X} ... vektor N elementov, ki opisuje frekvenčno funkcijo

\mathbf{fs} ... frekvenca vzorčenja

Izhodna parametra:

\mathbf{x} ... vektor N elementov, ki opisuje vzorčeni časovni signal

\mathbf{t} ... vektor N elementov, ki vsebuje časovne trenutke, katerim pripadajo elementi vektorja \mathbf{x}

Časovno funkcijo lahko v Matlabu narišemo z ukazom `plot(t,x)`

Primer 4: Spektralna analiza vlaka impulzov s pomočjo Matlab

V primeru 1 smo izračunali Fourier-jevo eksponentno vrsto sode funkcije vlaka pravokotnih impulzov ($A = 1$, $d = 0.25$). Rezultate opisuje enačba 2.35 in slika 2.37.

Do enakih rezultatov pridemo z uporabo hitre Fourier-jeve transformacije v Matlabu (uporaba funkcij `fdomain` in `tdomain`).

Program v Matlabu za spektralno analizo je naslednji:

```

% Primer: Spektralna analiza vlaka impulzov
clear all;
T0=2*pi      %čas opazovanja, perioda vlaka impulzov, torej omega0=1
tau=T0/4;    % čas, ko je signal na 1
d=tau/T0;    %razmerje med trajanjem 1 in 0
N=8000;      % število vzorcev
Ts=T0/N;     % čas vzorčenja
t=(0:N-1)*Ts; % trenutki vzorčenja

% definiramo vlak impulzov
% 1/8 je impulz na 1, nato 6/8 na nič in spet 1/8 na 1
x=ones(1,N);
% x(1001:7000)=0;
x(1+N*tau/(T0*2):(N-(N*tau/(T0*2))))=0;
%alternativa
%x=0.5*(square(2*pi/T0*t+pi/4,25)+1);
% narišemo signal
figure(1);
plot(t,x,'r','LineWidth',3);
axis([0,7,-0.2,1.2]);
title('A - Vlak impulzov, T0=2pi,tau=T0/4');
xlabel('Cas (sek)');
ylabel('Amplituda');

% Izračunamo spekter
fs=1/Ts; % frekvenca vzorčenja
[X,f]=fdomain(x,fs);

% Narišemo spekter
figure(2);
% ovojnica
k=[-12:0.1:12];
ss=d*sinc(k*d);
plot(k,ss);
hold on

stem(2*pi*f,X,'b','LineWidth',2,'Marker','none');
axis([-12,12,-0.1,0.30]);
title('B - Kompleksni spekter');
xlabel('Frekvenca (rd/s)');
ylabel('Absolutna vrednost');

```

```

hold off;

figure(3);
stem(2*pi*f,abs(X),'k','LineWidth',2,'Marker','none');
axis([-12,12,-0.1,0.30]);
title('C - Amplitudni spekter');
xlabel('Frekvenca (rd/s)');
ylabel('Absolutna vrednost');

figure(4);
phi=atan2(imag(X),real(X));
stem(2*pi*f,phi,'k','LineWidth',2,'Marker','none');
axis([-12,12,-5,5]);
title('D - Fazni spekter');
xlabel('Frekvenca (rd/s)');
ylabel('Faza');

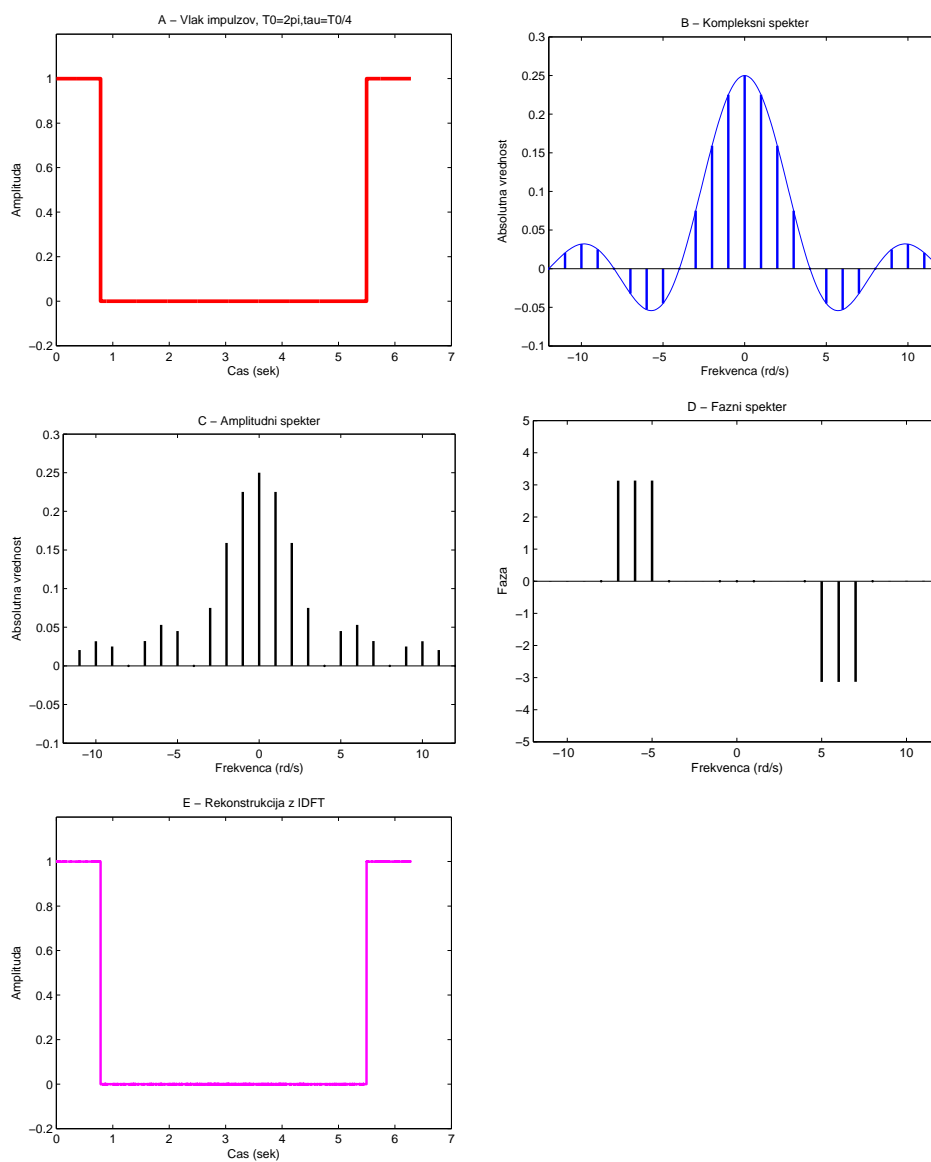
%Dokaz, da z IDFT dobimo nazaj originalni časovni signal
[xxx,ttt]=tdomain(X,fs)
figure(5)
plot(ttt,xxx,'m','Linewidth',2);
axis([0,7,-0.2,1.2]);
title('E - Rekonstrukcija z IDFT');
xlabel('Cas (sek)');
ylabel('Amplituda');

```

Izberemo čas opazovanja $T_0 = 2\pi$, $N = 8000$ vzorcev, kar pomeni, da je perioda vzorčenja $T_s = \frac{2\pi}{8000}$. V programu definiramo časovni signal in nato s pomočjo funkcije `fdomain` izračunamo spekter. V tem primeru je spekter realen (ima sicer pozitivne in negativne komponente vendar ne kompleksnih), zato ga lahko predstavimo z enim diagramom. V ta diagram narišemo tudi ovojnico s pomočjo funkcije `sinc`. Nato v posebna diagrama narišemo absolutno vrednost spektra in fazo. Fazne vrednosti so 0 , π in $-\pi$. Na koncu s pomočjo funkcije `tdomain` frekvenčni spekter nazaj pretvorimo v časovni signal in se prepričamo, da dobimo enak signal.

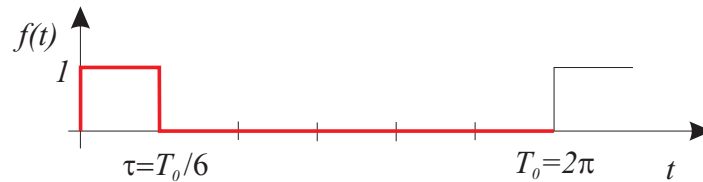
Rezultate prikazuje slika 2.48.

Slika 2.48: Rezultati Matlabovega programa: A - vlak impulzov, B - kompleksni spekter, C - amplitudni spekter, D - fazni spekter, E - rekonstruirani časovni signal



Primer 5: Izračun kompleksnega spektra vlaka impulzov, potrditve z Matlabom

Izračunajmo in z Matlabom preverimo spektralno analizo signala, ki ga prikazuje slika 2.49. Čas opazovanja je $T_0 = 2\pi$, dolžina časa, ko je vrednost 1, je $\tau = \frac{T_0}{6}$, $d = \frac{\tau}{T_0} = \frac{1}{6}$.



Slika 2.49: Impulz

S pomočjo enačbe 2.30 izračunamo

$$\begin{aligned}
 c_k &= \frac{1}{T_0} \int_0^{\tau} f(t) e^{-jk\omega_0 t} dt = \frac{1}{T_0} \int_0^{\tau} f(t) e^{-jk\frac{2\pi}{T_0} t} dt = \frac{j}{2k\pi} \left[e^{-jk2\pi\frac{\tau}{T_0} t} - 1 \right] = \\
 &= \frac{\tau}{T_0} \frac{\sin\left(k\pi\frac{\tau}{T_0}\right)}{k\pi\frac{\tau}{T_0}} e^{-jk\pi\frac{\tau}{T_0}} = \frac{\tau}{T_0} \operatorname{sinc}\left(k\frac{\tau}{T_0}\right) e^{-jk\pi\frac{\tau}{T_0}} = \\
 &= d \operatorname{sinc}(kd) e^{-jk\pi d}
 \end{aligned} \tag{2.44}$$

$$f(t) = \sum_{k=-\infty}^{+\infty} c_k e^{jk\omega_0 t} = d \sum \operatorname{sinc}(kd) e^{-jk\pi d} e^{jk2\pi\frac{t}{T_0}}$$

Pri $d = \frac{1}{6}$ je amplitudni spekter

$$|c_k| = \frac{1}{6} \left| \operatorname{sinc}\left(\frac{k}{6}\right) \right| \tag{2.45}$$

fazni spekter pa

$$\phi_k = \begin{cases} -k\frac{\pi}{6} & , \operatorname{sinc}\frac{k}{6} > 0 \\ -k\frac{\pi}{6} + \pi & , \operatorname{sinc}\frac{k}{6} < 0 \end{cases} \tag{2.46}$$

Program v Matlabu za spektralno analizo je naslednji:

```
% Primer iz knjige Mlakar: Linearna vezja in signali, zglede 160, str. 234
clear all;
T0=2*pi      %perioda vlaka impulzov, čas opazovanja
tau=T0/6;    %čas, ko je vrednost 1
f0=1/T0;     %frekvenca vlaka impulzov, tudi frekvenčni razloček med
              %dvema sosednjima frekvencama v spektru
N=600;       % število vzorcev
Ts=T0/N;     % čas vzorčenja
t=(0:N-1)*Ts; % trenutki vzorčenja

%definiramo vlak impulzov (korekcija, ker square daje vrednosti med -1 in 1
x=0.5*square(2*pi*f0*t,100*tau/T0)+0.5;

% narišemo signal
figure(1);
plot(t,x,'r','LineWidth',3);
axis([0,7,0,1.2])
title('A - Vlak impulzov, T0=2pi,tau=T0/6');
xlabel('Cas (sek)');
ylabel('Amplituda');

% Izračunamo kompleksni spekter
fs=1/Ts; % frekvenca vzorčenja
[X,f]=fdomain(x,fs);

% narišemo spekter

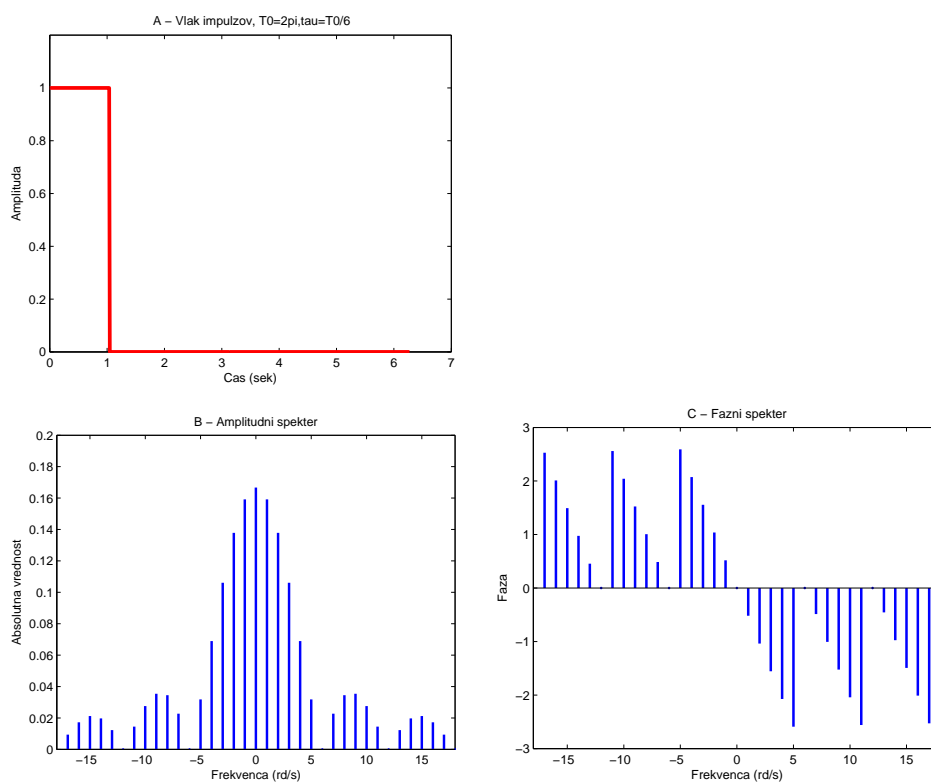
figure(2);
stem(2*pi*f,abs(X),'b','LineWidth',2,'Marker','none');
axis([-18,18,0,0.2]);
title('B - Amplitudni spekter');
xlabel('Frekvenca (rd/s)');
ylabel('Absolutna vrednost');

figure(3);
stem(2*pi*f,phase(X),'LineWidth',2,'Marker','none');
axis([-18,18,-3,3]);
title('C - Fazni spekter');
xlabel('Frekvenca (rd/s)');
ylabel('Faza');
```


Izberemo čas opazovanja $T_0 = 2\pi$, $N = 600$ vzorcev, kar pomeni, da je perioda vzorčenja $T_s = \frac{2\pi}{600}$. V programu definiramo časovni signal in nato s pomočjo funkcije `fdomain` izračunamo spekter. V tem primeru je spekter kompleksen, zato ga lahko nazorno predstavimo le v dveh diagramih.

Rezultate prikazuje slika 2.50.

Slika 2.50: Rezultati Matlabovega programa: A - časovni signal, B - absolutna vrednost spektra, C - faza spektra



Primer 6: Digitalno filtriranje

20Hz sinusni signal amplitude 3 je moten s 50 Hz signalom amplitude 0.5. V Matlabu določite spekter z DFT. Želimo filtrirati (izločiti) signal 50Hz in iz filtriranega signala rekonstruirati časovni signal.

Program v Matlabu za digitalno filtriranje:

```
clear all;
T0=0.2      %čas opazovanja
f0=1/T0;    %frekvenčni razloček med
            %dvema sosednjima frekvencama v spektru
N=1000;     % število vzorcev
Ts=T0/N;    % čas vzorčenja
t=(0:N-1)*Ts; % trenutki vzorčenja

%definiramo časovni signal
x1=0.5*sin(2*pi*50*t);% motnja
x2=3*sin(2*pi*20*t); %koristni signal
x=x1+x2;
% narišemo vse signale
figure(1);
plot(t,x, t,x1,t,x2,'LineWidth',1.5);
title('A - koristni signal (20Hz), motja (50Hz), vsota - moteni signal');
xlabel('Cas (sek)');
ylabel('Casovni signal');
axis([0,0.2,-4,4]);

% Izračunamo kompleksni spekter z DFT
fs=1/Ts; % frekvenca vzorčenja
[X,f]=fdomain(x,fs);

% narišemo spekter

figure(2);
stem(f,abs(X),'k','LineWidth',2,'Marker','none');
axis([-100,100,0,2]);
title('B - Amplitudni spekter motenega signala');
xlabel('Frekvenca (Hz)');
ylabel('Absolutna vrednost');
```

```

figure(3);
phi=atan2(imag(X),real(X));
stem(f,phi,'LineWidth',2,'Marker','none');
axis([-100,100,-4,4]);
title('C - Fazni spekter motenega signala');
xlabel('Frekvenca (Hz)');
ylabel('Faza');

%Filtriranje 50hz (501+-10)
X(491)=0;
X(511)=0;

figure(4);
stem(f,abs(X),'k','LineWidth',2,'Marker','none');
axis([-100,100,0,2]);
title('D - Amplitudni spekter filtriranega signala');
xlabel('Frekvenca (Hz)');
ylabel('Absolutna vrednost');

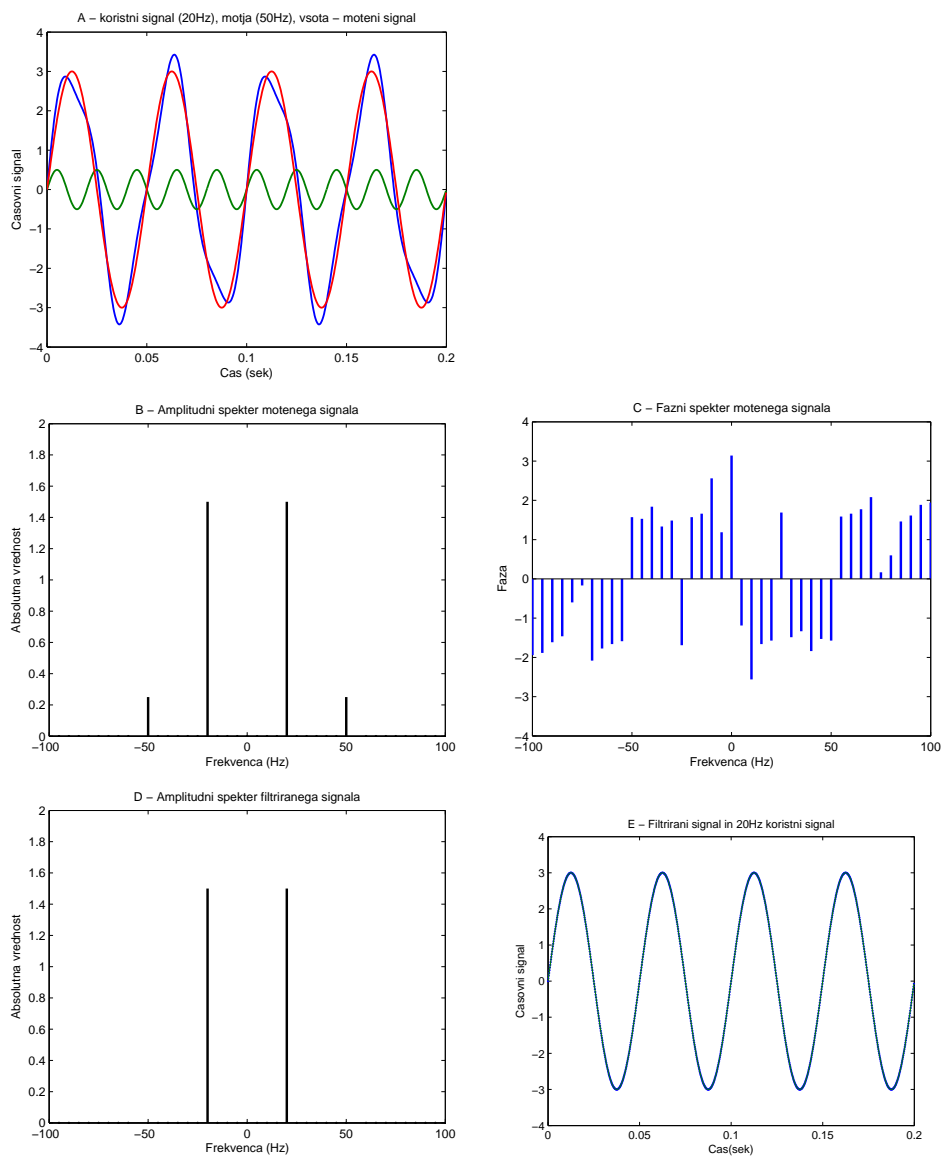
%Pretvorimo nazaj v časovni signal
[xxx,ttt]=tdomain(X,fs)
figure(5)
plot(t,x2,'.',ttt,xxx);
axis([0,0.2,-4,4]);
title('E - Filtrirani signal in 20Hz koristni signal');
ylabel('Casovni signal');
xlabel('Cas(sek)');

```

Izberemo čas opazovanja $T_0 = 0.2$, $N = 1000$ vzorcev, kar pomeni, da je perioda vzorčenja $T_s = 0.2ms$. V programu definiramo časovni signal, ki je sestavljen iz koristnega signala ($20Hz$) in motnje ($50Hz$). Z DFT določimo spekter in iz spektra izločimo $50Hz$ signal (postavimo na nič 491 in 511 komponento, ker predstavljata frekvenci $-50Hz$ in $50Hz$, enosmerna komponenta je 501 komponenta). Filtriran signal prevedemo nazaj v časovni signal z IDFT in se prepričamo, da je enak koristnemu signalu v motenem signalu.

Rezultate prikazuje slika 2.51.

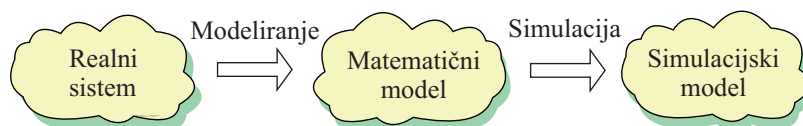
Slika 2.51: Digitalno filtriranje: A - koristni signal, motnja, moteni signal, B - amplitudni spekter motenega signala, C - fazni spekter motenega signala, D - amplitudni spekter filtriranega signala, E - filtrirani signal in 20Hz koristni signal



3.

Modeliranje procesov

Modeliranje in simulacija sta dva neločljiva postopka, ki vsebujeta kompleksne aktivnosti v zvezi s konstrukcijo modelov in eksperimentiranje z modeli v smislu pridobivanja podatkov o obnašanju modeliranega procesa (Matko in ostali, 1992, Karba, 1999). Pri tem je modeliranje vezano predvsem na relacije med realnim procesom in njegovimi modeli, simulacija pa se ukvarja s povezavo med matematičnim in simulacijskim modelom (računalniškim programom)- glej sliko 3.1. S slednjim je mogoče zelo fleksibilno eksperimentirati. Pridobljene časovne odzive vsaj v začetni fazi zlasti uporabljamo za vrednotenje modela.



Slika 3.1: Modeliranje in simulacije

3.1 Cilji in pomembnejše zakonitosti pri modeliranju

Namen študija sistemov s pomočjo modeliranja in simulacije je doseganje različnih ciljev, ne da bi bilo potrebno eksperimentirati na realnem objektu ali pa v primeru, ko realni objekt (še) ne obstaja.

Najpogostejši cilji modeliranja in simulacije so:

- izboljšati poznavanje in razumevanje nekaterih mehanizmov delovanja obravnavanega sistema,
- napovedovati obnašanje v različnih situacijah, v prihodnosti (predikcija),
- oceniti parametre procesa, ki niso direktno merljivi,
- analizirati občutljivost parametrov realnega sistema,
- omogočiti eksperimente, ki bi bili v realnem svetu dragi, tvegani ali problematični, kar je pomembno tudi pri simulatorjih za učenje operaterjev.
- vadba operaterjev, pilotov, kirurgov, ... s pomočjo simulatorjev,
- uporaba modela za področje vodenja sistemov:
 - za optimiranje sistemov,
 - za odkrivanje napak,
 - za razvoj metod vodenja (npr. metode Ziegler-Nichols za ugaševanje PID regulatorjev),
 - za razvoj in vrednotenje rešitev vodenja.

Čeprav obstaja mnogo tehnik modeliranja in je na voljo precej različnih računalniških okolij za podporo pri modeliranju in predvsem simulaciji (npr. Dymola-Modelica, Matlab-Simulink - Dymola, 2008, Modelica, 2007, Simulink, 2009), pa se moramo zavedati, da niti model, niti računalnik ne moreta popolnoma nadomestiti človeških odločitev, presoje, intuicije in izkušenj, ki še vedno igrajo odločilno vlogo pri določanju vrednosti in uporabnosti modelov v praktičnih aplikacijah. Pri tem je modeliranje bolj problemsko orientirano, medtem ko je simulacija relativno neodvisna od obravnavanega primera.

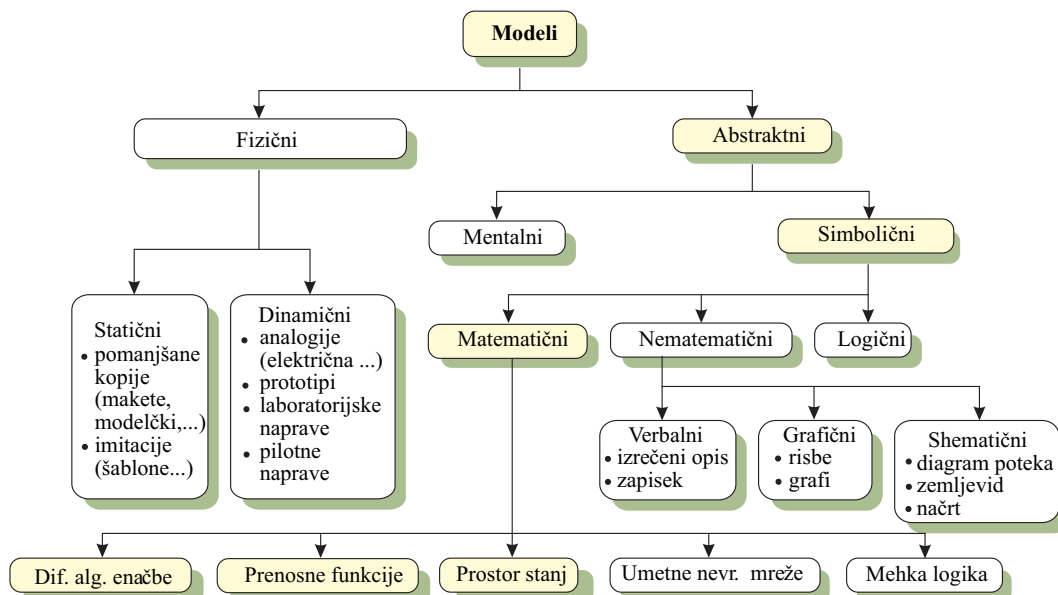
Za uspešno uporabo modeliranja moramo imeti v mislih naslednje najpomembnejše zakonitosti:

- model obravnava le **bistvene aspekte** realnega sistema,
- model poudarja tiste učinke gradnikov sistema, ki so pomembni s stališča **namena modeliranja**,

- model mora biti kar najbolj **enostaven**, saj je razvoj univerzalnega modela nemogoč, razvoj prekompleksnega modela pa je nepraktičen, neekonomičen in običajno povezan z dolgim časom potrebnim za simulacijo in z numeričnimi problemi med simulacijo,
- če hočemo dobiti enostaven model, moramo skrbno analizirati, **kaj upoštevati** in **kaj zanemariti** (zanemariti moramo dele, ki imajo majhen vpliv na delovanje, ne pa delov, ki jih dobro ne razumemo),
- pri modeliranju je treba skrbno upoštevati **omejitve** (npr. fizične omejitve signalov in komponent),
- model mora predstaviti naše znanje o sistemu **v primerni obliki** (papir, računalniški program, ...).

3.2 Vrste modelov

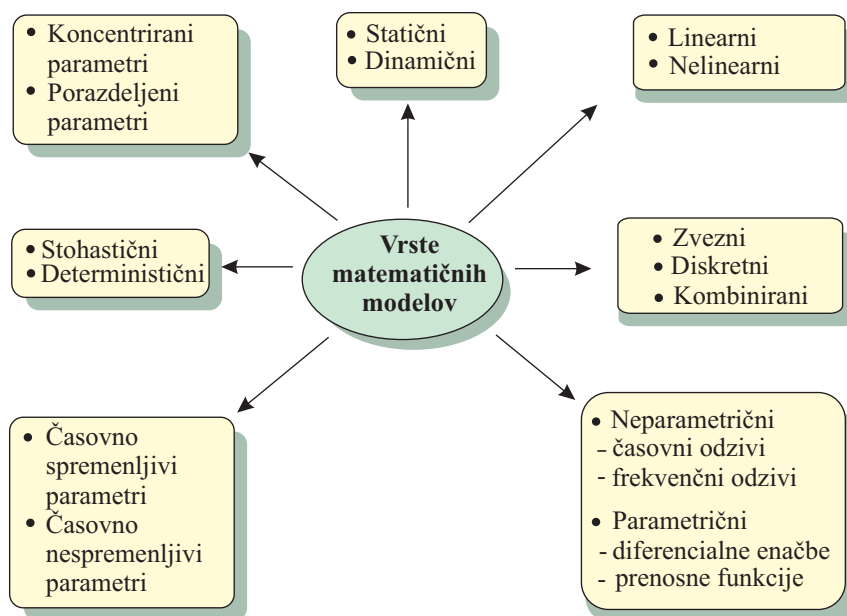
Poznamo seveda veliko različnih vrst modelov. Eno od možnih delitev modelov na različne vrste prikazuje slika 3.2.



Slika 3.2: Vrste modelov

Iz slike vidimo, da gre za različne predstavitve nekega realnega procesa. V našem delu se bomo posvetili le matematičnim modelom, ki bodo opisani z diferencialno algebrskimi enačbami in prenosnimi funkcijami. Informacija, ki jo dajejo, je splošno razumljiva, enoumno razložljiva in precizna. Manipulacija pa je učinkovita in poceni.

Matematične modele delimo na različne načine, kar prikazuje slika 3.3.



Slika 3.3: Razdelitev matematičnih modelov

Mi se bomo v nadaljevanju osredotočili predvsem na dinamične in deterministične modele v parametrični obliki.

Linearni in nelinearni modeli

Procese, pri katerih lahko relacije med vhodnimi in izhodnimi spremenljivkami ter spremenljivkami stanj izrazimo z linearnimi matematičnimi izrazi, imenujemo linearne. Primer:

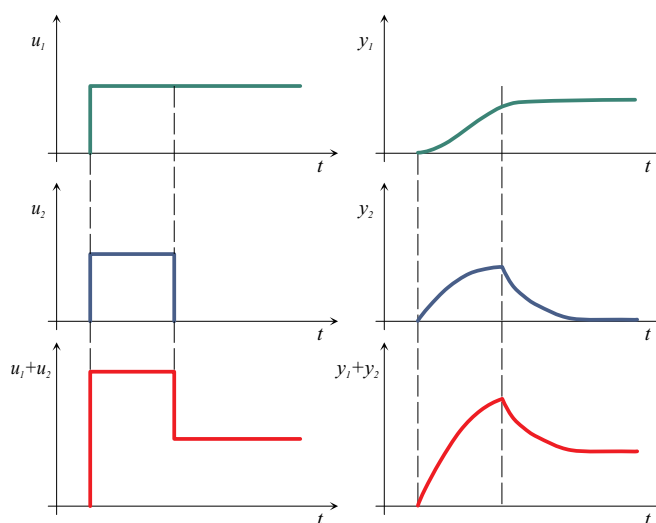
$$\begin{aligned}y(t) &= 2u(t) \\ \dot{y}(t) + 3y(t) &= 2u(t)\end{aligned}$$

Če pa so odvisnosti med procesnimi veličinami nelinearne, govorimo o nelinearnih procesih. Primer:

$$y(t) = au^2(t)$$

$$\begin{aligned} \dot{y}(t) + \frac{y(t)}{u(t)} &= 0 \\ \dot{y}(t) + \sqrt{y(t)} &= u(t) \\ \dot{y}(t) &= u(t) \sin y(t) \end{aligned}$$

Za linearne sisteme velja princip **superpozicije**, ki ga opišemo z naslednjimi besedami: če vhod u_1 povzroči izhod y_1 in vhod u_2 izhod y_2 , potem bo odziv na vhod $u_1 + u_2$ enak $y_1 + y_2$. Grafično princip superpozicije prikažemo s primerom na sliki 3.4.



Slika 3.4: Princip superpozicije

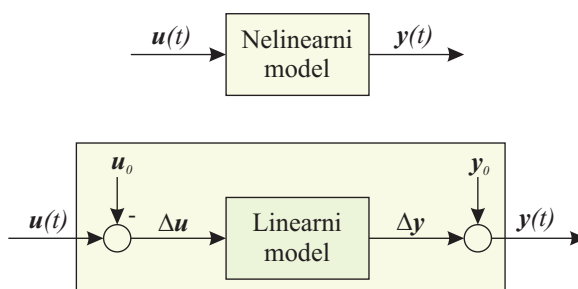
Že pri najenostavnejši nelinearni relaciji $y = u^2$ pa vidimo, da zakon superpozicije ne velja, saj $(u_1 + u_2)^2$ ni enako $u_1^2 + u_2^2$.

Sistem, ki je sestavljen izključno iz linearnih sklopov oz. podsistemov, je tudi linearen. Zato lahko linearnost sistema preverimo tako, da se prepričamo o linearnosti njegovih sestavnih sklopov.

Če ne modeliramo ravno enostavnih 'šolskih' primerov, nas matematično modeliranje skoraj vedno pripelje do nelinearnih modelov. Narava je žal pretežno nelinearna. Nelinearni modeli pa povzročajo probleme pri analizi, vrednotenju in še posebej pri načrtovanju vodenja, ker je večina metod razvitih za linearne modele. Zato, če je le možno, nelinearni model lineariziramo. To naredimo okoli delovne točke. Ta je določena z ustaljenimi vrednostmi procesnih spremenljivk. Predpostavljamo, da se model za majhne spremembe okoli delovne točke vede

približno linearno. To je še zlasti uporabno ravno pri načrtovanju vodenja, ko regulacijski sistem skrbi, da procesne spremenljivke ostajajo v področju želenih vrednosti.

Slika 3.5 prikazuje, na kakšen način uporabljamo nelinearni in linearni model.



Slika 3.5: Nelinearni in linearni model

Predvsem se moramo zavedati, da so procesne spremenljivke sestavljene iz delovnih vrednosti in majhnih sprememb. Tako je vhodni signal sestavljen iz

$$\mathbf{u}(t) = \mathbf{u}_0 + \Delta\mathbf{u}(t)$$

izhodni signal pa

$$\mathbf{y}(t) = \mathbf{y}_0 + \Delta\mathbf{y}(t)$$

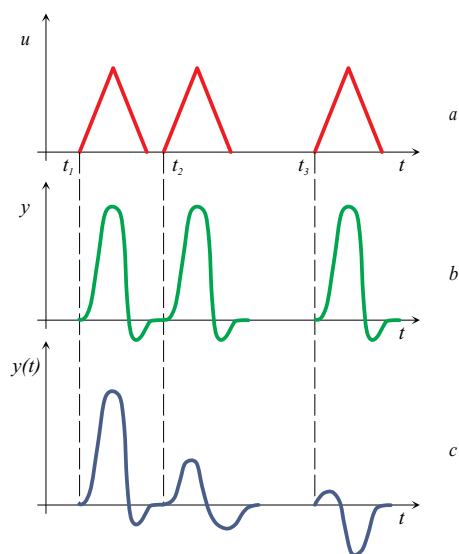
\mathbf{u}_0 in \mathbf{y}_0 sta delovni (ustaljeni) vrednosti vhodnega in izhodnega signala, $\Delta\mathbf{u}(t)$ in $\Delta\mathbf{y}(t)$ pa sta majhni spremembi. Pri uporabi linearnega modela moramo zato od operativnega vhodnega signala $\mathbf{u}(t)$ odšteti delovno vrednost \mathbf{u}_0 , da dobimo signal $\Delta\mathbf{u}$, ki je vhodni signal lineariziranega modela. Izhodu lineariziranega modela $\Delta\mathbf{y}$ pa moramo nato prišteti delovno vrednost izhodnega signala \mathbf{y}_0 in tako dobimo operativni izhodni signal $\mathbf{y}(t)$.

Časovno nespremenljivi in časovno spremenljivi modeli

Osnovna lastnost časovno nespremenljivih procesov oz. njihovih modelov je v tem, da je oblika izhoda neodvisna od trenutka nastopa vhodne veličine ali motnje.

Razliko med časovno spremenljivimi in nespremenljivimi procesi ponazorimo s sliko 3.6.

Omenimo še primer izrazito časovno spremenljivega procesa, kot je let rakete, ki z izgorevanjem goriva zelo hitro izgublja težo. V tem primeru seveda časovne spremenljivosti parametra mase $m(t)$ ne smemo zanemariti.



Slika 3.6: Prikaz: a) vhodnega signala b) izhoda časovno nespremenljivega procesa c) izhoda časovno spremenljivega procesa

Modeli s koncentriranimi in porazdeljenimi parametri

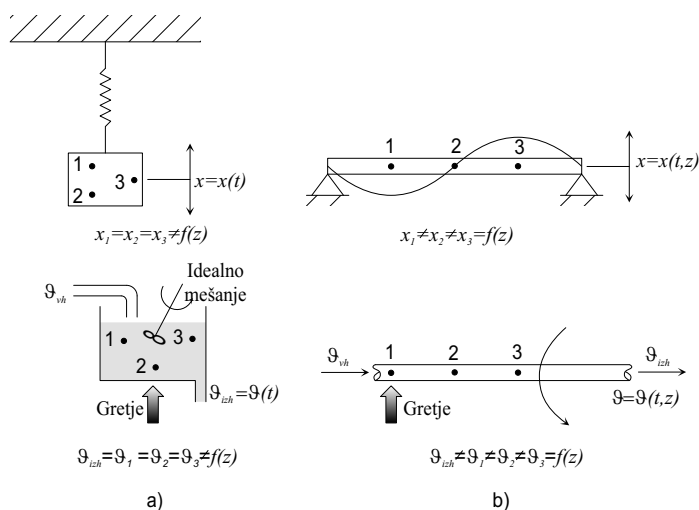
Procese, pri katerih je sprememba procesnih veličin neodvisna od prostorskih koordinat (torej je odvisna največkrat le od časa) imenujemo procese s koncentriranimi parametri. Kjer pa procesne spremenljivke v istem trenutku niso enake po celotnem prostoru, govorimo o procesih s porazdeljenimi parametri. Nekaj primerov kaže slika 3.7.

Največ procesov s porazdeljenimi parametri je takih, da je potrebno upoštevati le eno, najbolj poudarjeno prostorsko koordinato.

Zvezni, diskretni in kombinirani modeli

Bolj fizikalno opisuje razliko delitev na modele zveznih, kosovnih in šaržnih procesov. Pri kosovnih procesih se spremembe stanj dogajajo le v diskretnih trenutkih. Pri zveznih sistemih imamo kontinuirne in neprekinjene masne ali energijske tokove. Pri šaržnih procesih pa se ti tokovi med proizvodnjo prekinjajo, tako da gre za kombinacijo zveznih procesov in diskretnih dogodkov.

Z bolj matematično formulacijo opredelimo za zvezne tiste modele, v katerih so vse spremenljivke zvezne funkcije časa (zavzamejo poljubno vrednost iz definicijskega območja in zaloge vrednosti). Uporabljamo diferencialne enačbe in



Slika 3.7: Primera nihanja in gretja: a) procesa s koncentriranimi parametri (enake vrednosti pomikov oz. temperatur v vseh treh točkah) b) procesa s porazdeljenimi parametri (različne vrednosti pomikov oz. temperatur v vseh treh točkah)

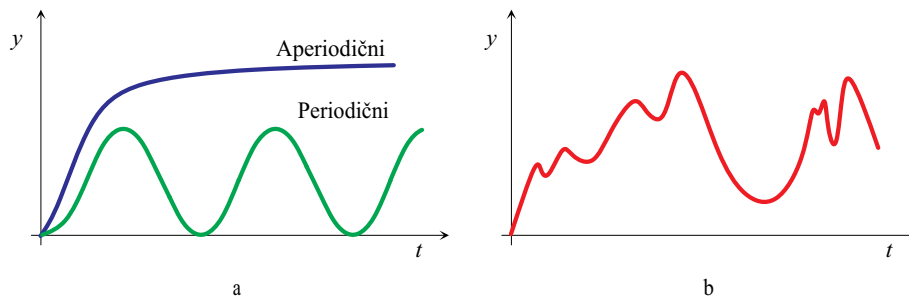
Laplace-ovo transformacijo. Pri diskretnih modelih lahko signali zavzamejo le določene vrednosti iz definicijskega območja (diskretni signali) in/ali iz zaloge vrednosti (digitalni signali). Obravnava temelji na teoriji vzorčenja, diferenčnih enačbah in z-transformaciji. Opisani diskretni modeli so konceptualno podobni zveznim modelom. Imamo pa tudi diskretne modele, ki se konceptualno precej razlikujejo. To so modeli, ki opisujejo sisteme diskretnih dogodkov (angl. DEVS - discrete event systems). S temi modeli proučujemo promet na letališčih, v križiščih, bencinskih črpalkah, v veleblagovnicah ipd. Tu se uporablja teorija strežnih sistemov (queueing systems), Petrijevih mrež, končnih avtomatov.

Deterministični (določljivostni) in stohastični (naključni) modeli

Deterministične imenujemo tiste modele, v katerih nastopajo signali, ki so opisljivi z matematičnimi funkcijami. Stohastični modeli pa so opisljivi s statističnimi parametri: srednja vrednost, standardna deviacija, porazdelitveni zakon. Primer odziva determinističnega in stohastičnega modela prikazuje slika 3.8.

Parametrični in neparametrični modeli

Prvi so podani s pomočjo diferencialnih enačb, prenosnih funkcij, zapisa v prostoru stanj. Vse te strukture namreč zahtevajo tudi določitev parametrov.



Slika 3.8: Primer časovnega odziva: a) determinističnega modela b) stohastičnega modela

Neparametrični modeli pa so v obliki časovnih odzivov, frekvenčnih odzivov, korelacijskih funkcij ipd.

Statični in dinamični modeli

Za statične modele je značilno, da ne vsebujejo stanj, ki bi se med delovanjem s časom spreminjala. Povezava med tokom in napetostjo na električnem uporu

$$i = \frac{1}{R}u$$

je povsem algebrajska. S statičnimi modeli včasih prikazujemo tudi razmere v ustaljenem stanju.

Pri dinamičnih modelih so v igri diferencialne enačbe oz. sistemi s časovno spremenljivimi notranjimi stanji. Primer:

$$\ddot{y}(t) + 2\dot{y}(t) + 7y(t) = 2u(t)$$

3.3 Načini modeliranja

Uporabljamo naslednje načine modeliranja:

- teoretično,
- eksperimentalno in
- kombinirano modeliranje.

3.3.1 Teoretično modeliranje

Bistvo tega pristopa je v dobrem poznavanju fizikalnih (ravnotežnih) zakonov. Temeljni so zakoni o

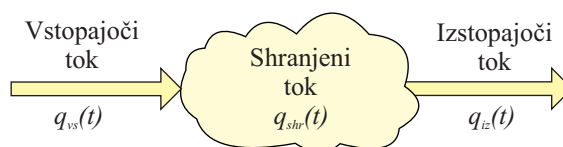
- ohranitvi mase,
- energije in
- gibalne količine.

Zelo praktični pa so še razni zakoni iz ustreznega področja (Kirchofovi zakoni, Newtonovi zakoni, ...). Tak pristop je zlasti uporaben v večini tehničnih znanosti, pri tako imenovanih mehkih znanostih (družboslovje, sociologija in včasih biološke in medicinske vede) pa lahko predstavljajo resen problem. Teoretično modeliranje nas običajno vodi do precej kompleksnih sistemov, kar metodi lahko jemlje značaj uporabnosti. Zato se moramo največkrat odločiti za skrbno pretehtane in dobro argumentirane zanemaritve, poenostavitve in razne predpostavke, ki naj bi omogočale predstavitev sistema z navadnimi, linearnimi diferencialnimi enačbami čim nižjega reda (n.pr. prehod iz sistema nelinearnih parcialnih diferencialnih enačb na sistem lineariziranih diferencialnih enačb, ki ga končno še ustrezno reduciramo). Tako moramo poiskati nek kompromis med natančnostjo in kompleksnostjo modela, pri čemer odločilno vlogo igra namen modeliranja. Teoretično modeliranje zahteva zelo poglobljeno in detajlno poznavanje modeliranega procesa in zato je sodelovanje problemskega strokovnjaka v postopku modeliranja nujno (problem ustvarjanja interdisciplinarne skupine). Važna lastnost teoretičnih modelov je, da jih lahko uporabimo tudi za podobne probleme ob ustrezni spremembi parametrov. Prav tako je pomembno, da lahko tovrstne modele uporabljamo že v fazi planiranja in načrtovanja nekega objekta in pa povsod tam, kjer meritve na procesu iz kakršnegakoli vzroka niso možne.

Modeliranja električnih sistemov se lotimo na nam že znan način z Kirchofovimi zakoni. Modeliranja mehanskih sistemov se lotimo z Newtonovimi zakoni. V kolikor niso na voljo zakoni določenega področja, lahko uporabimo zakona o ohranitvi mase ali energije in formulacijo z besedilnim zapisom (glej sliko 3.9)

shranjeni tok = vstopajoči tok - izstopajoči tok

$$q_{shr} = q_{vs} - q_{iz} \quad (3.1)$$

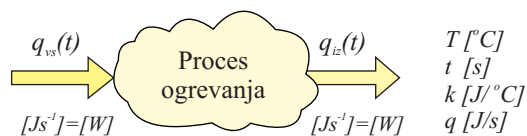


Slika 3.9: Teoretično modeliranje

Tok je lahko masni, volumski, energijski ali celo informacijski.

Primer: Modeliranje ogrevanja prostora

Slika 3.10 prikazuje proces ogrevanja prostora.



Slika 3.10: Ogrevanje prostora

Spremenljivka $q_{vs}(t)$ je v tem primeru vstopajoči energijski tok električnega grelca (enota $[Js^{-1}] = [W]$). $q_{iz}(t)$ je izstopajoči izgubni energijski tok v enakih enotah. Če obstaja razlika med obema tokoma, pomeni, da obstaja shranjeni energijski tok, ki povzroči spremembo temperature $\frac{dT(t)}{dt}$.

$$k \frac{dT(t)}{dt} = q_{vs}(t) - q_{iz}(t) \quad (3.2)$$

k je konstanta, ki jo včasih določimo s fizikalnim znanjem, včasih pa eksperimentalno.

Primer: Modeliranje hidravličnega sistema

Slika 3.11 prikazuje hidravlični sistem polnjenja posode.

Posoda ima masni dotok q_{vs} ($[kgs^{-1}]$). Na dnu je odprtina, ki povzroči iztok $q_{iz}(t)$. V tem primeru razlika med vstopnim in izstopnim masnim tokom povzroči spremembo volumna oz. nivoja tekočine $\frac{dh(t)}{dt}$.

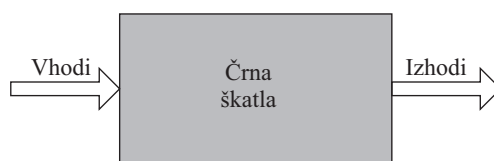
Slika 3.11: Hidravlični sistem

$$k \frac{dh(t)}{dt} = q_{vs}(t) - q_{iz}(t) \quad (3.3)$$

k je konstanta, ki jo določimo s fizikalnim znanjem ali z eksperimentalnim pristopom.

3.3.2 Eksperimentalno modeliranje

Pristop, ki ga imenujemo tudi identifikacija, temelji na izbiri vhodov in izhodov obravnavanega sistema in na meritvah teh signalov. Želimo določiti matematično strukturo in ustrezne parametre tako, da bi bili njeni časovni odzivi čim bližje merjenim ob enakih vhodnih signalih. Zanimajo nas le vhodno-izhodne informacije, ne pa tudi mehanizmi, ki jih povzročajo. Sistem torej obravnavamo kot 'črna škatla' (slika 3.12).



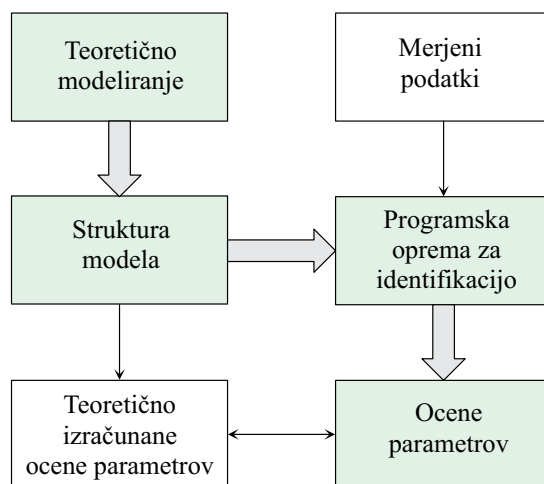
Slika 3.12: Eksperimentalno modeliranje

Uporabljamo številne računalniške programe (npr. Identification Toolbox v Matlabu). Metode so relativno enostavne in jasne ter dajejo kot rezultat linearne modele nizkega reda. Uporabljamo jih lahko tudi na področjih, kjer je teoretično modeliranje problematično. Zavedati pa se moramo, da iz njih ne dobimo nobene informacije o fizikalnem ozadju problema. Dobljeni modeli so zato uporabni le kot celota.

Pri eksperimentalnem modeliranju se pojavljajo problemi z ločevanjem koristnega signala od merjenega, ki je moten s šumi, pa tudi z izbiro vzbujevalnih signalov, ki bi zagotovili dovolj informacije o modeliranem procesu. Običajno se uporabljajo 'živahni' signali, ki vzbujejo sistem na dovolj širokem frekvenčnem področju (npr. kombinacija stopničastih signalov, pseudo naključni binarni signal, ...). Žal pa taki signali ponavadi niso podobni vhodnim signalom pri normalnem delovanju procesa in je zato največkrat težko dobiti dovoljenje za ustrezne eksperimente. Končno je potrebno poudariti, da identifikacije ne moremo uporabiti v primerih, ko ni na razpolago ustreznih meritev.

3.3.3 Kombinirano modeliranje

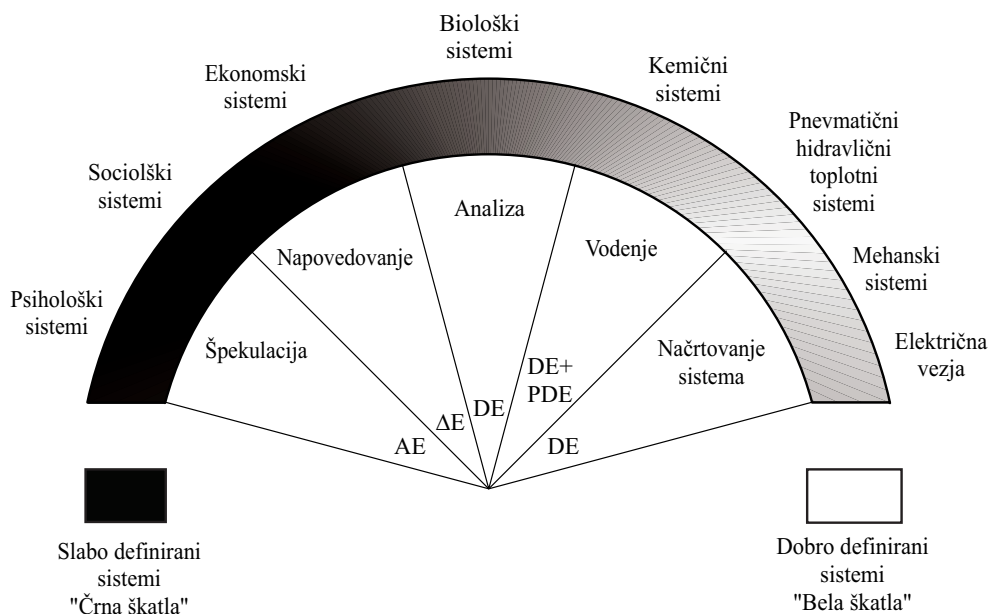
V praksi največkrat uporabljamo kombinirano modeliranje, ki skuša izkoristiti dobre lastnosti omenjenih pristopov (slika 3.13). Pri tem strukturo modela v glavnem definiramo s pomočjo teoretičnega pristopa, medtem ko nekatere parametre lahko ocenimo s pomočjo eksperimentalnega modeliranja. Poglavitna prednost tega pristopa je v tem, da na ta način ohranimo osnovne funkcionalne povezave v modelu, kar omogoča lažje razumevanje delovanja sistema in bolj fleksibilno uporabo modela. Kombinirano modeliranje na sliki 3.13 označujejo odebeljene puščice.



Slika 3.13: Prikaz kombiniranega modeliranja

Slika 3.14 prikazuje spekter področij uporabe, namenov in metodologij pri matematičnem modeliranju in simulaciji. Opazimo, kako se področja spreminjajo od slabo definiranih, kjer je model le boljša ali manjša špekulacija, pa do dobro

definiranih, kjer je 'moč modela' običajno še posebej velika. Pri slabo definiranih sistemih se predvsem poslužimo eksperimentalnega modeliranja, pri dobro definiranih sistemih pa teoretičnega modeliranja.

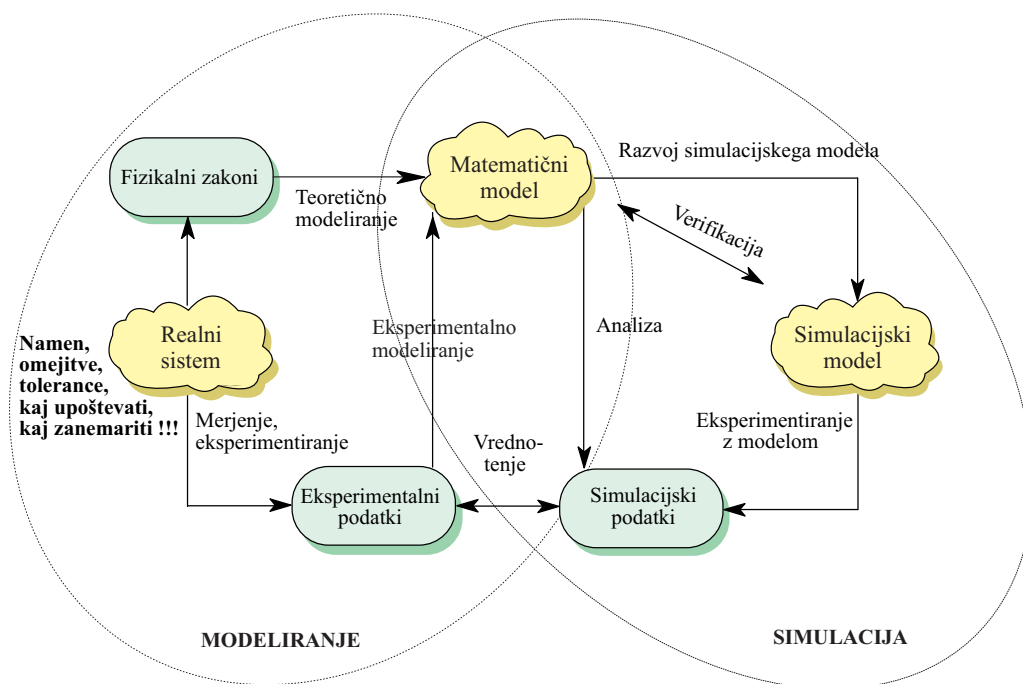


Slika 3.14: Spekter področij, uporabe, namenov in metodologij pri matematičnem modeliranju in simulaciji (AE, ΔE , DE in PDE - algebrajske, diferencialne, diferencialne enačbe)

3.4 Modeliranje in simulacija kot enovit ciklični postopek

Postopek modeliranja je iterativnega (cikličnega) značaja, kar pomeni, da ne pridemo v enem zaporedju korakov do zelenega modela, ampak se moramo po raznih korakih vračati nazaj in izvesti določene spremembe. Bistveno vlogo ima tudi simulacija, zato pravimo, da sta modeliranje in simulacija nerazdružljivi metodi povezani v enovit ciklični postopek. Prikazuje ga slika 3.15.

Vsak realni objekt, za katerega želimo razviti matematični model (npr. za potrebe vodenja), začnemo proučevati s pomočjo fizikalnih zakonov oz. zakonov o ohranitvi mase, energije ali gibalne količine. Pravimo, da s teoretičnim modeliranjem pridemo do matematičnega modela. V primeru slabo definiranih fizikalnih za-



Slika 3.15: Iterativni postopek modeliranja in simulacije

konov se zatečemo k eksperimentalnemu modeliranju. Na realnem sistemu izvajamo meritve in razne bolj kompleksne eksperimente in na ta način pridobimo eksperimentalne podatke za eksperimentalno modeliranje. V praksi pa največkrat kombiniramo oba pristopa. Pri gradnji matematičnega modela moramo zlasti:

- jasno opredeliti namen modeliranja,
- definirati omejitve, znotraj katerih deluje model,
- upoštevati tolerance (npr. ne moremo delati z večjo natančnostjo, kot je natančnost meritev),
- izbrati lastnosti (attribute), ki jih bomo upoštevali in zanemariti nepomembne (to so tiste, ki malo vplivajo na delovanje, ne pa tiste, ki jih ne poznamo!!!) ter idealizirati določene realne zakonitosti.

Ko zgradimo matematični model, moramo zbrati čim več informacij o njegovem vedenju. Če morda obstajajo analitične rešitve, jih je vredno uporabiti, saj se lahko tako izognemo nekaterim nevšečnostim, ki se pojavijo pri simulaciji (npr.

precejšnja poraba računalniškega časa, numerični zapleti, ...). V splošnem pa uporabljamo pristop, da iz matematičnega modela zgradimo t.i. simulacijski model, ki služi nato za preizkušanje realnega sistema. Pomembno je torej, da imata realni sistem in simulacijski model enak matematični model. Izgradnja simulacijskega modela je prvi korak pri simulaciji. Drugi korak pa je eksperimentiranje s tem modelom. S pomočjo analitične obravnave matematičnega modela in eksperimentiranja s simulacijskim modelom dobimo simulacijske podatke. S primerjavo simulacijskih in eksperimentalnih podatkov (glej sliko 3.15) za ustrezne realne in simulacijske eksperimente izvedemo eno najpomembnejših faz v postopku modeliranja - vrednotenje ali validacijo, t.j. analizo ujemanja obnašanja realnega sistema in matematičnega modela. Pri tem moramo poudariti, da vrednotenje modela ni isto kot verifikacija. Verifikacija je analiza ujemanja simulacijskega in matematičnega modela. Pravzaprav ob predpostavki, da je matematični model pravilen, preverjamo, če je bil simulacijski program pravilno izveden. Verifikacija je torej preverjanje dela opisanega postopka, medtem ko je validacija celovito preverjanje.

Analizo sistema, konstrukcijo modela, eksperimentiranje z modelom, vrednotenje in verifikacijo je običajno potrebno večkrat ponoviti, dokler ne pridemo do zahtevanih rezultatov. Zato Neelamkavil (Neelamkavil, 1987) simulacijo v širšem smislu definira kot počasno, iterativno in eksperimentalno orientirano tehniko.

Znano je, da za vsak mehanski ali hidravlični sistem lahko poiščemo električni ekvivalent. To pomeni, da so sistemi izrazljivi z enakimi diferencialnimi enačbami, torej imajo enak matematični model. V praksi je lahko eden izmed njih realni sistem, drugi pa simulacijski model, ali obratno. Seveda za simulacijske modele izberemo vedno take, s katerimi je možno enostavno eksperimentirati (logično je, da npr. mehanski sistem simuliramo z električnim analognim sistemom in ne obratno).

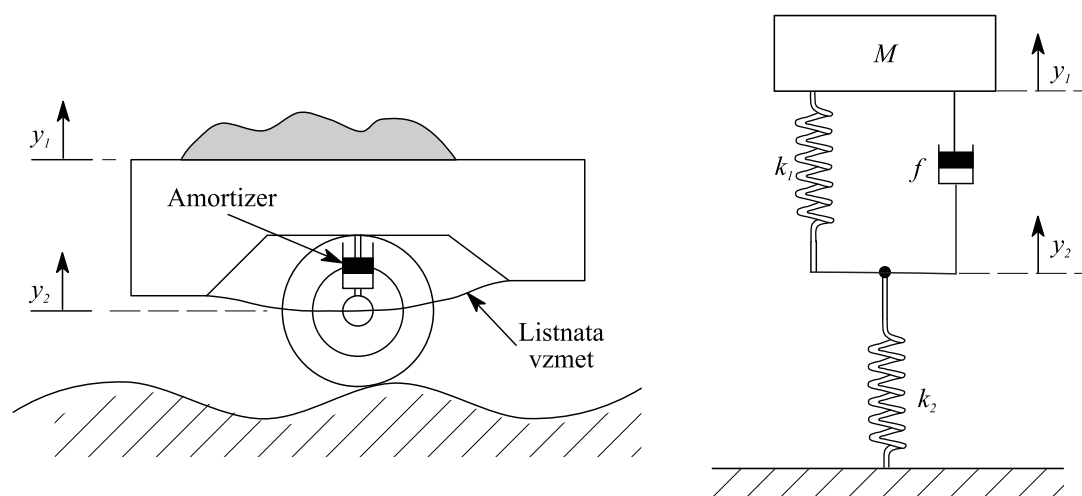
Da bo povezava modeliranja in simulacije dovolj poudarjena, bomo podali tri primere, v katerih bo poudarek najprej na modeliranju, v naslednjih poglavjih pa bomo te modele uporabljali za prikaz simulacijskih metod in orodij.

3.5 Primeri

Primer 3.1 Modeliranje avtomobilskega vzmetenja

Modeliranje poenostavljenega avtomobilskega vzmetenja predstavlja primer teoretičnega modeliranja na osnovi ravnotežja gibalnih količin. Običajno je model sistem dveh linearnih diferencialnih enačb drugega reda, ki opisujeta premike karoserije avtomobila in premike kolesa. Predpostavljamo torej model s koncentriranimi parametri. Pri tem upoštevamo vzmeti in dušilnike avtomobila ter elastičnost pnevmatike. Oblika podlage predstavlja vzbujanje sistema. Model lahko služi kot dober pripomoček za spoznavanje modeliranega procesa, za študij vpliva posameznih elementov na obnašanje sistema, pa tudi za grobo dimenzioniranje elementov.

V našem primeru pa bomo uvedli še eno predpostavko. Ker smo zainteresirani le za obliko gibanja karoserije, bomo zanemarili še maso kolesa in obese, ki je majhna v primeri s četrtino mase avtomobila. Tako lahko obravnavani sistem prikažemo kot kombinacijo translatorskih mehanskih elementov na sliki 3.16.



Slika 3.16: Poenostavljen prikaz vzmetenja avtomobila

Na sliki 3.16 predstavlja M četrtino mase avtomobila, k_1 je konstanta togosti vzmeti avtomobila, f je konstanta dušenja dušilnika avtomobila, k_2 je konstanta togosti pnevmatike, y_1 je premik karoserije, y_2 pa premik kolesa iz mirovne lege.

Osnovno relacijo za začetek teoretičnega modeliranja predstavlja *drugi Newtonov zakon*, saj predstavlja eno od oblik zakona o ravnotežju gibalnih količin

$$\Sigma F = Ma = M\ddot{y} \quad (3.4)$$

kjer je M masa [kg], a je pospešek [ms^{-2}] in F je sila [N]. Glede na to enačbo moramo najti ustrezne relacije še za ostale sile v sistemu (njihove povezave s premikom, ki nas zanimajo). Za silo vzmeti je na razpolago Hook-ov zakon, ki podaja silo, ki je potrebna, da raztegemo (ali pa stisnemo) vzmet za razdaljo y glede na njeno osnovno dolžino

$$F_s = ky \quad (3.5)$$

pri čemer je k konstanta vzmeti [Nm^{-1}].

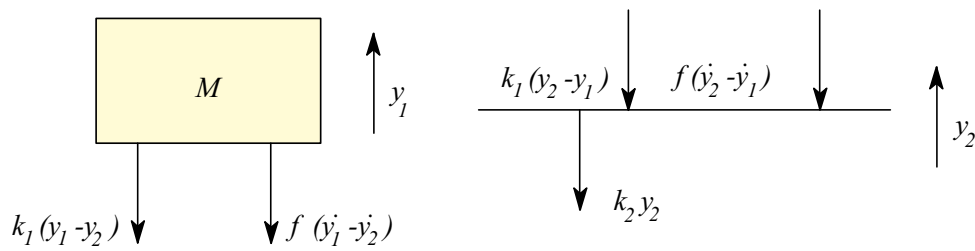
Silo, ki je potrebna, da premaknemo en konec viskoznega dušilnika s hitrostjo v relativno glede na drugi konec, podaja zveza

$$F_D = fv = f\dot{y} \quad (3.6)$$

kjer je f dušilni koeficient [Nsm^{-1}].

Tako se moramo odločiti le še o predznaku sil, kar najlaže naredimo s pomočjo diagrama sil. Pri tem se dogovorimo, da so sile v smeri predpostavljene premika (največkrat ga izberemo v smeri sile, ki vzbuja sistem) pozitivne. Zavedati pa se moramo, da se sile na vzmeteh in dušilnikih vedno upirajo kakršnemkoli premiku. Prav tako je jasno, da premiki v mehanskem vezju niso vezani le na mase, ki so vključene vanj, temveč se, kot v našem primeru, lahko pojavijo tudi ob zaporednih vezavah elementov, ali pa če vzbujevalna sila deluje na vzmet ali na dušilnik, ne pa na maso. Kadar je vsak konec elementa podvržen različnima premikoma (ni vezan na mirujočo točko), se namesto premika y oz. odvoda \dot{y} v zvezah pojavi razlika premikov oz. odvodov (v tej razliki ima prva spremenljivka isti indeks, kot ga ima premik točke, v kateri delujejo obravnavane sile). Glede na povedano lahko za naš primer narišemo diagram sil na sliki 3.17.

Iz diagrama sil na sliki 3.17 lahko direktno napišemo enačbi za oba premika tako, da sile s predznakom, kot ga narekuje diagram sil, vpišemo na desno stran enačbe (3.4)



Slika 3.17: Diagram sil za sistem avtomobilskega vzmetenja

$$M\ddot{y}_1 = -f(\dot{y}_1 - \dot{y}_2) - k_1(y_1 - y_2) \quad (3.7)$$

$$0 = -f(\dot{y}_2 - \dot{y}_1) - k_1(y_2 - y_1) - k_2y_2 \quad (3.8)$$

Enačbi (3.7) in (3.8) že predstavljata matematični model našega procesa. Ker pa nas zanima le y_1 , eliminirajmo y_2 . Vsota enačb (3.7) in (3.8) daje

$$M\ddot{y}_1 = -k_2y_2 \quad (3.9)$$

iz česar izrazimo y_2 kot

$$y_2 = -\frac{M}{k_2}\ddot{y}_1 \quad (3.10)$$

Če y_2 sedaj vnesemo v enačbo (3.7), dobimo

$$\frac{fM}{k_2}\ddot{y}_1 + M\left(1 + \frac{k_1}{k_2}\right)\ddot{y}_1 + f\dot{y}_1 + k_1y_1 = 0 \quad (3.11)$$

in končno

$$\ddot{y}_1 + \frac{k_1 + k_2}{f}\ddot{y}_1 + \frac{k_2}{M}\dot{y}_1 + \frac{k_1k_2}{Mf}y_1 = 0 \quad (3.12)$$

Dobili smo torej navadno linearno diferencialno enačbo tretjega reda s konstantnimi koeficienti, ki opisuje gibanje karoserije avtomobila.

Do sedaj nismo rekli še ničesar o vzbujanju sistema. Kot vemo, je sistem lahko vzbujan z vhodnim signalom, ali pa ima v trenutku, ko ga začnemo opazovati, določeno začetno stanje. Zunanji vhod bi lahko predstavljala podlaga po kateri vozi avto. Kakšna ovira ali jama na podlagi bi vzbudila sistem, pri čemer pa ne smemo pozabiti, da je glede na predpostavko naše kolo infinitezimalno majhno. Izberimo začetno stanje

$$y_1(0) = -y_{10} \quad (3.13)$$

Začetno stanje si lahko predstavljamo tako, da voznik vstopi v avtomobil, v trenutku $t = 0$ pa izstopi.

Predpostavimo naslednje konkretne podatke avtomobilskega vzmetenja:

$M=500 \text{ kg}$, $k_1=7500 \text{ N/m}$, $k_2=150000 \text{ N/m}$, $f=2250 \text{ N s/m}$ in $y_{10}=0.05 \text{ m}$.

Tako lahko enačbo (3.12) zapišemo v splošni obliki

$$\ddot{y}_1 + a\dot{y}_1 + by_1 + cy_1 = 0 \quad (3.14)$$

pri čemer so

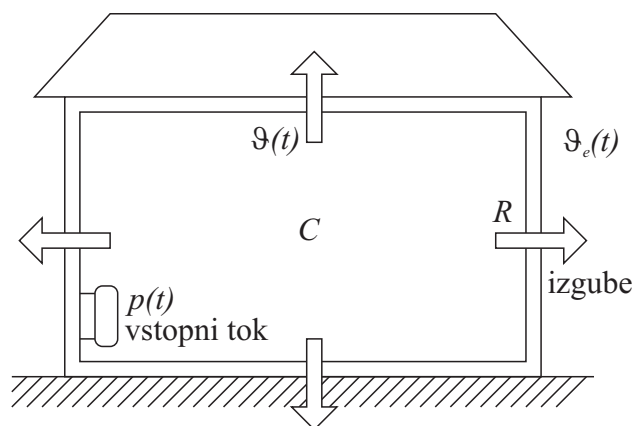
$a = 70$, $b = 300$, $c = 1000$ in $y_1(0) = -y_{10} = -0.05$

□

Primer 3.2 Modeliranje regulacije gretja prostora

Oglejmo si še možnost "intuitivnega" pristopa k modeliranju gretja prostora z grelom, ki ga vključuje ali izključuje termostatski regulator (slika 3.18).

To je klasična in cenena možnost regulacije temperature v prostoru, ki jo v vsakdanjem življenju največkrat uporabljamo. Učinki regulacije zavisijo v glavnem od treh faktorjev:



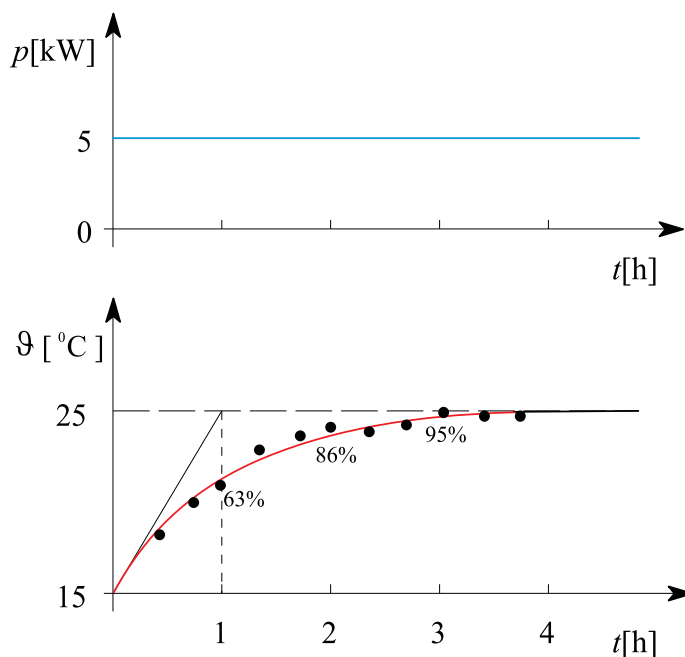
Slika 3.18: Modeliranje ogrevanja prostora

- moči grelnega telesa,
- histereze termostatskega regulatorja in
- zakasnitev, ki so prisotne v procesu.

Ti faktorji določajo nihanja temperature okrog želene vrednosti. Ker želimo majhna nihanja, bi morala biti tudi histereza termostata majhna, kar pa po drugi strani povzroča pogosto prižiganje in ugašanje grela in zopet predstavlja nezaželen pojav. Seveda pa je za čas, v katerem dosežemo želeno temperaturo, odločilna moč grela. Načrtovanje sistema gretja (dimenzioniranje gretja in izbira ter postavitve termostata) bi lahko potekalo v samem prostoru, vendar bi potrebne meritve zahtevale ogromno časa, saj so časovne konstante procesa v razredu ur. Pristop z natančno matematično analizo pa je tudi problematičen saj histereza, ki predstavlja nelinearnost, že v zelo enostavni povratnoznančni strukturi in pri do skrajnosti poenostavljenem modelu procesa povzroči komplicirano dinamično obnašanje obravnavanega sistema. Zato je pristop z modeliranjem in simulacijo najustreznejši. Ko se odločimo za enostaven model gretja prostora, lahko namreč celotno regulacijsko zanko simuliramo s pomočjo primerne simulacijskega orodja in tako mnogo hitreje kot pa na realnem sistemu s poskušanjem dobimo potrebne informacije o zaprtizančnem obnašanju sistema.

Omenjena dejstva so pomenila osnovne podatke o sistemu in namenu modeliranja. Glede na to skušajmo priti do modela gretja prostora z grelom na najenostavnejši "intuitivni" način, pri čemer pa mora dobljeni model vseeno zadovoljiti naše zahteve.

Zato v prostoru pri relativno konstantni temperaturi (npr. 15°C) prižgemo grelo moči 5kW ($t = 0$, $p = 5\text{kW}$) in v določenih časovnih intervalih (npr. 20 min) merimo temperaturo. Tako dobimo rezultate na sliki 3.19.



Slika 3.19: Gretje prostora ob vklopu grela (p ...moč gretja, ϑ ...temperatura v prostoru). Pike pomenijo merjene vrednosti, krivulja pa odziv modela.

Kot vidimo na sliki 3.19, je proces proporcionalen, kar smo tudi pričakovali, saj višanje temperature povzroča tudi večanje izgub toplote (skozi zidove, vrata, okna itd.). Izgube so namreč proporcionalne razliki med temperaturama prostora in okolice ($\vartheta - \vartheta_e$). Tako pridemo do ustaljenega stanja pri približno 25°C (pri tej temperaturi je toplota, ki jo proizvaja grelo, enaka izgubam). Glede na zakon o energijskem ravnotežju (shranjeni tok = vstopajoči tok - izstopajoči tok) lahko predpostavimo, da je shranjeni tok sorazmeren spremembi temperature v prostoru, vstopajoči tok je kar moč grela, izstopajoči tok pa predstavlja izgube in je seveda sorazmeren razliki med notranjo in zunanjo temperaturo:

$$C\dot{\vartheta} = p - \frac{\vartheta - \vartheta_e}{R} \quad (3.15)$$

R je termična upornost ovoja prostora [$^{\circ}\text{C}/\text{W}$] in je sorazmerna z debelino ovoja d [m] in obratno sorazmerna s površino ovoja S [m^2] in specifično prevodnostjo

λ ovoja [$\frac{W}{m^{\circ}C}$]

$$R = \frac{d}{S\lambda} \quad (3.16)$$

C je termična kapacitivnost [$J/^{\circ}C$] prostora. Če enačbo (3.15) delimo s konstanto C in uredimo enačbo tako, da na desni strani enačaja ostane le vhodni signal $p(t)$, dobimo enačbo

$$\dot{\vartheta} + \frac{1}{RC}(\vartheta - \vartheta_e) = \frac{1}{C}p \quad (3.17)$$

Namesto konstante RC vpeljemo časovno konstanto T , namesto konstante $\frac{1}{C}$ pa konstanto $\frac{k}{T}$, kjer je k ojačenje sistema. Tako dobimo standardno diferencialno enačbo sistema prvega reda

$$\dot{\vartheta} + \frac{1}{T}(\vartheta - \vartheta_e) = \frac{k}{T}p \quad (3.18)$$

kjer je pomen oznak naslednji:

ϑ - temperatura v prostoru [$^{\circ}C$],

ϑ_e - temperatura okolice [$^{\circ}C$], ki je približno konstantna in znaša $15^{\circ}C$,

p - moč grela [kW] v našem primeru $5 kW$,

k - ojačenje sistema prvega reda [$^{\circ}C/kW$],

T - časovna konstanta sistema prvega reda [h].

Tako dobljeni linearni model dobro opisuje realni proces za temperaturni interval $15^{\circ}C \leq \vartheta \leq 25^{\circ}C$ pri konstantni temperaturi okolice. Običajno pa uporabljamo tako imenovane deviacijske modele, ki podajajo razmere relativno na delovno točko (v našem primeru je to temperatura okolice). To storimo s pomočjo spremenljivke ϑ_w , ki jo definiramo kot razliko

$$\vartheta_w = \vartheta - \vartheta_e \quad (3.19)$$

kar daje končno obliko modela ($\dot{\vartheta} = \dot{\vartheta}_w$ ob predpostavki, da je temperatura

okolice ϑ_e konstantna)

$$\dot{\vartheta}_w + \frac{1}{T}\vartheta_w = \frac{k}{T}p \quad (3.20)$$

Model, ki ga podaja enačba (3.20), lahko zapišemo še v obliki prenosne funkcije

$$\frac{\theta_w(s)}{P(s)} = \frac{k}{Ts + 1} \quad (3.21)$$

Postavitev enačbe (3.20) predstavlja teoretični del modeliranja. Konstanti k in T pa skušajmo določiti iz merjenih rezultatov, kar pomeni neko vrsto eksperimentalnega modeliranja. To kaže, da smo pri tem problemu uporabili pravzaprav kombinirano modeliranje. Za naš sistem prvega reda lahko časovno konstanto določimo s pomočjo tangente na krivuljo ob času $t=0$ (kot kaže slika 3.19). Kjer tangenta seka premico ustaljenega stanja, odčitamo časovno konstanto T . Le-ta je v našem primeru ocenjena na

$$T = 1\text{h}$$

Ojačenje sistema pa določa razmerje med spremembo temperature v ustaljenem stanju in spremembo vhodnega signala (v našem primeru gretja)

$$k = \frac{\Delta\vartheta}{\Delta p} = \frac{25 - 15}{5 - 0} = 2^\circ\text{C/kW} \quad (3.22)$$

Če dobljene vrednosti vstavimo v enačbo (3.20), dobimo model ogrevanja prostora

$$\dot{\vartheta}_w + \vartheta_w = 2p \quad (3.23)$$

oz. v obliki prenosne funkcije

$$\frac{\theta_w(s)}{P(s)} = \frac{2}{s + 1} \quad (3.24)$$

V praksi pa se izkaže, da bi lahko ta model uporabili le v primeru, ko bi bil termostat, ki zaznava temperaturo prostora, zelo blizu grelnega telesa. Če temu

ni tako, bi problem boljše opisovala diferencialna enačba višjega reda, ki jo lahko zadovoljivo aproksimiramo z modelom prvega reda v kombinaciji z mrtvim časom. V tem primeru je potrebno zapis (3.20) spremeniti v zapis z dvema enačbama

$$\begin{aligned} \dot{\vartheta}_w + \frac{1}{T}\vartheta_w &= \frac{k}{T}p_d \\ p_d(t) &= p(t - T_d) \end{aligned} \quad (3.25)$$

kjer je T_d mrtvi čas. Prenosna funkcija takega sistema je

$$\frac{\theta_w(s)}{P(s)} = \frac{k}{Ts + 1} e^{-T_d s} \quad (3.26)$$

Zapis s pomočjo prenosne funkcije ponuja zelo ugodno in ilustrativno možnost predstavitve sistema s pomočjo bločnih diagramov. Pri študiju regulacijskih problemov (kot je to v našem primeru) lahko tako jasno prikažemo celotno strukturo regulacijske zanke.

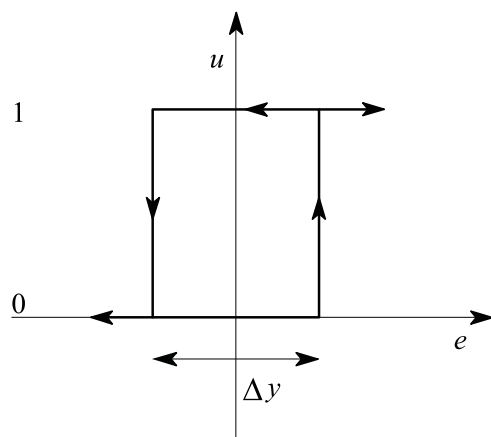
Zgradimo bločni diagram regulacije gretja prostora. Želena temperaturo prostora (ϑ_r) nastavimo na termostatskem regulatorju. Vhod v slednjega je tudi temperatura v prostoru ϑ . Termostatski regulator ustvari razliko

$$e = \vartheta_r - \vartheta \quad (3.27)$$

iz katere preko histerezne karakteristike

$$u(t) = \begin{cases} 0, & e < -\frac{\Delta y}{2} \\ 1, & e > \frac{\Delta y}{2} \\ u(t - \Delta t), & -\frac{\Delta y}{2} \leq e \leq \frac{\Delta y}{2} \end{cases} \quad (3.28)$$

določi regulirni signal. Ustrezno zakonitost prikazuje slika 3.20. Pri tem je $u(t - \Delta t)$ pretekla vrednost regulirnega signala (seveda 0 ali 1, pri tem je Δt poljubno majhen).



Slika 3.20: Histereza termostata

Regulirni signal u vklaplja in izklaplja grelec (npr. preko releja, kontaktorja). Le tega lahko modeliramo kar z nekim konstantnim ojačenjem p_{max} , oz. z enačbo

$$p = p_{max}u \quad (3.29)$$

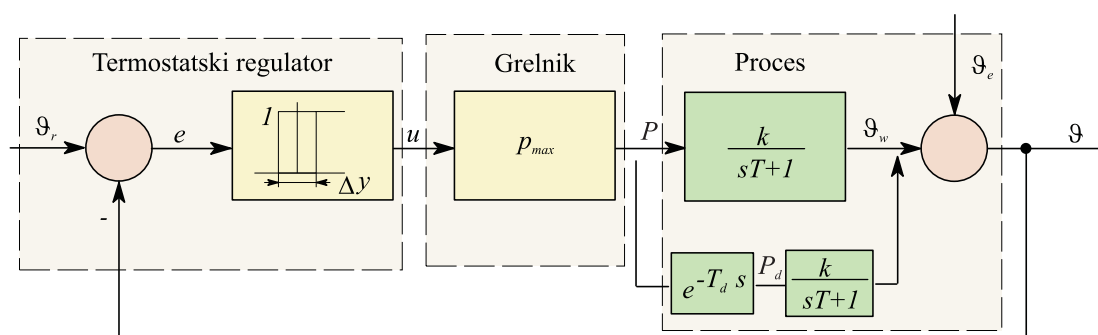
Dobljeni signal p predstavlja vhod v model procesa (enačba (3.20) za $T_d = 0$ in enačbi (3.25) za $T_d \neq 0$). Ker smo razvili deviacijski model, s simulacijo pa bi radi opazovali absolutne iznose spremenljivk, dobimo absolutni iznos temperature z vsoto temperature v okolici delovne točke (ϑ_w) in temperature delovne točke, ki je kar temperatura okolice (ϑ_e)

$$\vartheta = \vartheta_w + \vartheta_e \quad (3.30)$$

Bločni diagram regulacijske zanke pri gretju prostora tako prikazuje slika 3.21. V gradniku za proces sta navedeni obe možnosti: model prvega reda in model prvega reda z mrtvim časom.

Tako je problem pripravljen za nadaljnjo obravnavo. Naštejmo nekaj možnosti za študij različnih aspektov našega problema s pomočjo razvitega modela:

- določitev širine histereze termostata, ki daje majhna nihanja temperature ob ne preveč pogostem preklapljanju grela,
- določitev minimalne moči grela, ki še zagotavlja zadovoljivo regulacijo (sprejemljivo hitrost prilagajanja temperature novi želeni vrednosti, neobčutljivost na motnje itd.),



Slika 3.21: Bločni diagram regulacije temperature prostora

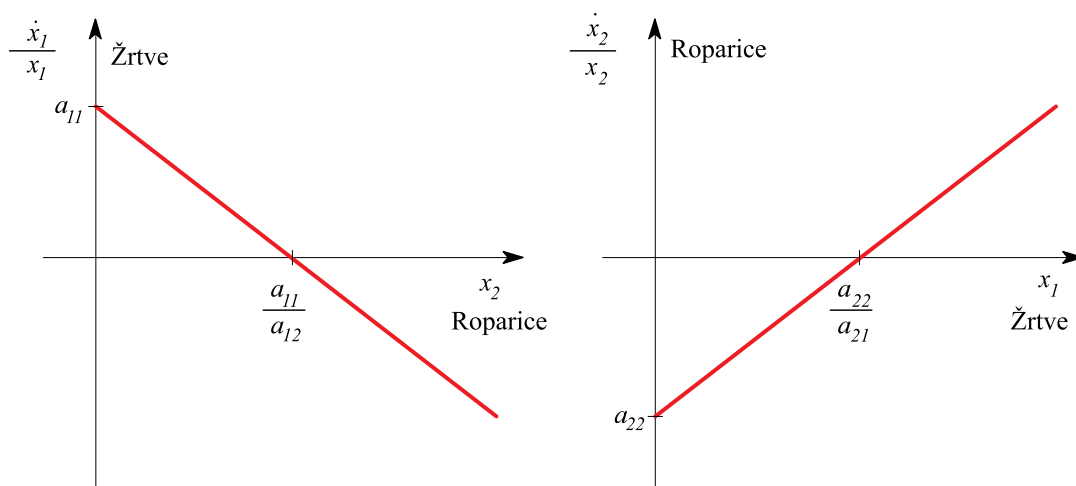
- študij vpliva motenj na regulirano temperaturo (možno je enostavno simulirati različne motnje na vhodu in izhodu procesa ali pa tudi na kakšnem drugem mestu),
- določitev optimalnega poteka želene temperature, ki obenem s primernim temperaturnim profilom v prostoru zagotavlja optimalno izrabo energije,
- študij vpliva lege termostata na temperaturo prostora (kar je posebno pomembno pri slabem mešanju zraka v prostoru),
- študij vpliva različnih časovnih konstant gretja in hlajenja prostora na temperaturo v prostoru itd.

□

Primer 3.3 Primer modeliranja populacijske dinamike

Oglejmo si še primer modeliranja enostavnega netehniškega problema. Pri populacijski dinamiki gre za študij rasti ali upadanja različnih populacij. Ta problem je dobro znan iz literature in če ga razširimo na obravnavo dveh populacij, ga imenujemo *model ekosistema Lhotka-Volterra*, še večkrat pa *problem roparic in žrtev*. Če jemljemo populaciji kot zvezni spremenljivki, je logaritmična rast ene odvisna le od druge populacije. Logaritmična rast populacije roparic je sorazmerna populaciji žrtev in logaritmična rast populacije žrtev je obratno sorazmerna populaciji roparic. Ob predpostavki, da je sorazmernost linearna, dobimo odvisnosti, kot jih prikazuje slika 3.22. Pri tem smo označili populacijo žrtev z x_1 , populacijo roparic z x_2 , a_{11} , a_{12} , a_{21} , a_{22} pa so konstante modela.

S pomočjo slike 3.22 lahko napišemo naslednji matematični model



Slika 3.22: Logaritmični rasti populacij žrtev in roparjev

$$\begin{aligned}\frac{\dot{x}_1}{x_1} &= a_{11} - a_{12}x_2 \\ \frac{\dot{x}_2}{x_2} &= a_{21}x_1 - a_{22}\end{aligned}\quad (3.31)$$

ali

$$\begin{aligned}\dot{x}_1 &= (a_{11} - a_{12}x_2)x_1 \\ \dot{x}_2 &= (a_{21}x_1 - a_{22})x_2\end{aligned}\quad (3.32)$$

Pri tem zanemarimo faktorje, ki bi povzročali upadanje rasti neke populacije in bi izvirali v tej isti populaciji. Tako se pri odsotnosti žrtev populacija roparic zmanjšuje, pri odsotnosti roparic pa se populacija žrtev zvečuje. Pozitivni konstanti a_{11} in a_{22} zavisita od hitrosti rasti, pozitivni konstanti a_{12} in a_{21} pa predstavljata vzajemne faktorje zadrževanja rasti in sta proporcionalni velikosti druge populacije.

Če bi naša razmišljanja prenesli na zajce (žrtve) in lisice (roparice), lahko konstante v enačbah (3.32) ocenimo na teoretičen način s pomočjo naslednjih predpostavk:

1. Vsak par zajcev ima povprečno 10 mladičev letno.

2. Vsaka lisica poje povprečno 25 zajcev letno.
3. Povprečna starost lisic je 5 let, kar pomeni, da letno umre 20 % lisic.
4. V povprečju število mladih lisic, ki preživijo, zavisi od dosegljive hrane (število preživelih mladih lisic je torej enako številu zajcev deljeno s 25).

Za časovno enoto eno leto lahko torej izračunamo konstanti a_{11} in a_{22} iz predpostavk 1 in 3 na naslednji način:

$$\begin{aligned} a_{11} &= \left. \frac{\dot{x}_1'}{x_1} \right|_{a_{12}=0} \approx \frac{\Delta x_1'/\Delta t}{x_1} = \frac{10}{2} = 5 \\ a_{22} &= - \left. \frac{\dot{x}_2'}{x_2} \right|_{a_{21}=0} \approx - \frac{\Delta x_2'/\Delta t}{x_2} = - \frac{-1}{5} = 0.2 \end{aligned} \quad (3.33)$$

Kot smo omenili, sta konstanti a_{12} in a_{21} odvisni tudi od področja na katerem opazujemo populaciji, kar pomeni, da sta odvisni od povprečnega števila zajcev in lisic. Če npr. opazujemo območje 50 km^2 , kjer je povprečno število zajcev $\bar{x}_1 = 500$ in lisic $\bar{x}_2 = 100$, predpostavka 2 daje

$$a_{12} = - \left. \frac{\dot{x}_1''}{\bar{x}_1 \bar{x}_2} \right|_{a_{11}=0} \approx - \frac{\Delta x_1''/\Delta t}{\bar{x}_1 \bar{x}_2} = - \frac{25 \bar{x}_2}{\bar{x}_1 \bar{x}_2} = \frac{25}{500} = 0.05 \quad (3.34)$$

predpostavka 4 pa

$$a_{21} = \left. \frac{\dot{x}_2''}{\bar{x}_1 \bar{x}_2} \right|_{a_{22}=0} \approx \frac{\Delta x_2''/\Delta t}{\bar{x}_1 \bar{x}_2} = \frac{\bar{x}_1/25}{\bar{x}_1 \bar{x}_2} = \frac{1}{25 \bar{x}_2} = \frac{1}{2500} = 0.0004 \quad (3.35)$$

Tako smo prišli do matematičnega modela za obravnavani primer:

$$\begin{aligned} \dot{x}_1 &= 5x_1 - 0.05x_1x_2 \\ \dot{x}_2 &= 0.0004x_1x_2 - 0.2x_2 \end{aligned} \quad (3.36)$$

Eksperimentirali bomo pri naslednjih začetnih pogojih: 520 zajcev in 85 lisic:

$$\begin{aligned}x_1(0) &= 520 \\x_2(0) &= 85\end{aligned}\tag{3.37}$$

Model predstavlja sistem dveh navadnih nelinearnih diferencialnih enačb prvega reda s konstantnimi koeficienti. □

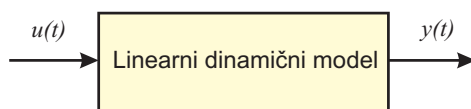
4.

Zapisi matematičnih modelov

Konvencionalni zapis matematičnih modelov temelji na diferencialno-algebrskih enačbah. Kot smo omenili, pridemo v modeliranju običajno do nelinearnih sistemov, ki jih največkrat rešujemo z numerično simulacijo. Ker pa dovršen del analize in načrtovanja sistemov vodenja temelji na linearni teoriji, skušamo priti tudi do linearnih ali lineariziranih zapisov, ki so veljavni vsaj v nekem ožjem področju delovanja. Linearne zapise uspešno rešujemo tudi analitično. Inženirji ponavadi uporabljajo Laplace-ovo transformacijo. Le-ta vodi do dveh inženirsko zelo priljubljenih zapisov linearnih modelov, t.j. do prenosnih funkcij in bločnih diagramov.

4.1 Zapis z diferencialno enačbo

Slika 4.1 prikazuje linearni dinamični model z vhodnim signalom $u(t)$ in izhodnim signalom $y(t)$.



Slika 4.1: Linearni dinamični model

Zapišemo ga z linearno diferencialno enačbo

$$a_0 y^{(n)} + a_1 y^{(n-1)} + \dots + a_{n-1} \dot{y} + a_n y = b_0 u^{(m)} + b_1 u^{(m-1)} + \dots + b_{m-1} \dot{u} + b_m u \quad (4.1)$$

Členi na levi strani enačaja pripadajo izhodni spremenljivki, členi na desni strani pa vhodni spremenljivki. Model je n -tega reda. Red sistema je namreč določen s stopnjo najvišjega odvoda izhodne spremenljivke. Velja tudi $m \leq n$. Če je $m = n$, potem ima model to lastnost, da se pri neki trenutni spremembi vhodnega signala tudi izhodni signal trenutno spremeni. To ni lastnost modelov večine realnih procesov, zato velja običajno $m < n$.

4.2 Laplace-ova transformacija

Laplace-ova transformacija je matematični pripomoček, zlasti primeren za reševanje diferencialnih enačb in je zato izjemno pomembna pri analizi in načrtovanju sistemov vodenja. Z uporabo Laplace-ove transformacije je možno prevesti funkcije kot so sinusoida, dušena sinusoida, eksponencialna funkcija, itd. v racionalne funkcije kompleksne spremenljivke s . Operacije kot npr. diferenciranje in integriranje se v kompleksni ravnini s izražajo kot algebraične operacije. Zato lahko s pomočjo Laplace-ove transformacije prevedemo diferencialne enačbe v algebrajske. Ob reševanju diferencialne enačbe s pomočjo Laplace-ove oz. inverzne Laplace-ove transformacije dobimo kompletno rešitev: ustaljeno stanje in prehodni pojav.

Laplace-ov transform časovne funkcije $f(t)$, $f(t) = 0$ za $t < 0$ je definiran z izrazom

$$\mathcal{L}[f(t)] = F(s) = \int_0^{\infty} f(t)e^{-st} dt \quad (4.2)$$

kjer je s kompleksna spremenljivka. Transformacija, ki omogoča pretvorbo Laplace-ovega transformata v časovno funkcijo, je inverzna Laplace-ova transformacija.

$$\mathcal{L}^{-1}[F(s)] = f(t) \quad (4.3)$$

Najpogosteje uporabljene Laplace-ove transforme in ustrezne časovne funkcije prikazuje tabela 4.1.

Teoremi Laplace-ove transformacije

1. Množenje s konstanto

$$\mathcal{L}[k \cdot f(t)] = k \cdot F(s) \quad (4.4)$$

2. Vsota in razlika

$$\mathcal{L}[f_1(t) \pm f_2(t)] = F_1(s) \pm F_2(s) \quad (4.5)$$

3. Odvajanje

$$\mathcal{L}\left[\frac{d}{dt}f(t)\right] = sF(s) - f(0^+) \quad f(0^+) = \lim_{t \rightarrow 0^+} f(t) \quad (4.6)$$

$$\begin{aligned} \mathcal{L}\left[\frac{d^n f(t)}{dt^n}\right] &= s^n F(s) - \lim_{t \rightarrow 0^+} \left[s^{n-1} f(t) + s^{n-2} \frac{d f(t)}{dt} + \dots + \frac{d^{n-1} f(t)}{dt^{n-1}} \right] = \\ &= s^n F(s) - s^{n-1} f(0^+) - s^{n-2} f^{(1)}(0^+) - \dots - f^{(n-1)}(0^+) \quad (4.7) \end{aligned}$$

4. Integracija

$$\mathcal{L}\left[\int_0^t f(\tau) d\tau\right] = \frac{F(s)}{s} \quad (4.8)$$

$$\mathcal{L}\left[\int_0^{t_1} \int_0^{t_2} \dots \int_0^{t_n} f(\tau) d\tau dt_1 \dots dt_{n-1}\right] = \frac{F(s)}{s^n} \quad (4.9)$$

5. Premik

$$\mathcal{L}[f(t - \tau)] = e^{-\tau s} F(s) \quad (4.10)$$

6. Začetna vrednost

$$\lim_{t \rightarrow 0} f(t) = \lim_{s \rightarrow \infty} s F(s) \quad (4.11)$$

7. Končna vrednost

$$\lim_{t \rightarrow \infty} f(t) = \lim_{s \rightarrow 0} s F(s) \quad (4.12)$$

Teorem velja, če funkcija $s F(s)$ nima nobenih polov z pozitivnimi realnimi deli.

Tabela 4.1: Laplace-ovi transformi in časovne funkcije

Številka	$F(s)$	$f(t), t \geq 0$
1	1	$\delta(t)$ (delta impulz)
2	$\frac{1}{s}$	1(t) (enotina stopnica)
3	$\frac{1}{s^2}$	t
4	$\frac{2!}{s^3}$	t^2
5	$\frac{3!}{s^4}$	t^3
6	$\frac{m!}{s^{m+1}}$	t^m
7	$\frac{1}{(s+a)}$	e^{-at}
8	$\frac{1}{(s+a)^2}$	te^{-at}
9	$\frac{1}{(s+a)^3}$	$\frac{1}{2!}t^2e^{-at}$
10	$\frac{1}{(s+a)^m}$	$\frac{1}{(m-1)!}t^{m-1}e^{-at}$
11	$\frac{a}{s(s+a)}$	$1 - e^{-at}$
12	$\frac{a}{s^2(s+a)}$	$\frac{1}{a}(at - 1 + e^{-at})$
13	$\frac{b-a}{(s+a)(s+b)}$	$e^{-at} - e^{-bt}$
14	$\frac{s}{(s+a)^2}$	$(1 - at)e^{-at}$
15	$\frac{\omega^2}{s^2 + 2\zeta\omega s + \omega^2}$	$\frac{\omega}{\sqrt{1-\zeta^2}}e^{-\zeta\omega t} \sin\omega\sqrt{1-\zeta^2}t$
16	$\frac{a^2}{s(s+a)^2}$	$1 - e^{-at}(1 + at)$
17	$\frac{(b-a)s}{(s+a)(s+b)}$	$be^{-bt} - ae^{-at}$
18	$\frac{\omega}{(s^2 + \omega^2)}$	$\sin\omega t$
19	$\frac{s}{(s^2 + \omega^2)}$	$\cos\omega t$
20	$\frac{s+a}{(s+a)^2 + \omega^2}$	$e^{-at}\cos\omega t$
21	$\frac{\omega}{(s+a)^2 + \omega^2}$	$e^{-at}\sin\omega t$
22	$\frac{a^2 + \omega^2}{s[(s+a)^2 + \omega^2]}$	$1 - e^{-at}(\cos\omega t + \frac{a}{\omega}\sin\omega t)$

4.3 Reševanje diferencialne enačbe z Laplace-ovo transformacijo

Z Laplace-ova transformacijo dobimo rešitev, ki sestoji iz splošne rešitve in partikularnih rešitev. Začetni pogoji se avtomatsko vključijo v rešitev. Dobimo prehodni pojav in ustaljeno stanje.

Če so vsi začetni pogoji nič, izvedemo Laplace-ovo transformacijo diferencialne enačbe tako, da d/dt zamenjamo z Laplace-ovim operatorjem s , d^2/dt^2 z s^2 itd. Če začetni pogoji niso nič, jih moramo vključiti v skladu s teoremom odvajanja. Za rešitev diferencialne enačbe potrebujemo dva koraka:

- Vsak člen diferencialne enačbe nadomestimo z ustreznim Laplace-ovim transformom, kar pripelje do algebrajske enačbe spremenljivke s . Z ustrežno preureditvijo pridemo do Laplace-ovega transformata odvisne spremenljivke ($Y(s)$).
- Rešitev diferencialne enačbe (časovni potek) dobimo z uporabo inverzne Laplace-ove transformacije.

Primer 4.1

$$\ddot{y} + 3\dot{y} + 2y = 0 \quad y(0) = a, \dot{y}(0) = b \quad (4.13)$$

$$\begin{aligned} \mathcal{L}[y(t)] &= Y(s) \\ \mathcal{L}[\dot{y}(t)] &= sY(s) - y(0) \\ \mathcal{L}[\ddot{y}(t)] &= s^2Y(s) - sy(0) - \dot{y}(0) \end{aligned} \quad (4.14)$$

Z ustreznimi nadomestitvami pridemo do enačbe

$$[s^2Y(s) - sy(0) - \dot{y}(0)] + 3[sY(s) - y(0)] + 2Y(s) = 0 \quad (4.15)$$

Če v to enačbo vstavimo začetne pogoje, dobimo enačbo

$$[s^2Y(s) - as - b] + 3[sY(s) - a] + 2Y(s) = 0 \quad (4.16)$$

ali

$$(s^2 + 3s + 2)Y(s) = as + b + 3a \quad (4.17)$$

ter z ustrezno preureditvijo

$$Y(s) = \frac{as + b + 3a}{s^2 + 3s + 2} = \frac{as + b + 3a}{(s + 1)(s + 2)} = \frac{2a + b}{s + 1} - \frac{a + b}{s + 2} \quad (4.18)$$

Z inverzno Laplace-ovo transformacijo dobimo rezultat

$$\begin{aligned} y(t) &= \mathcal{L}^{-1}[Y(s)] = \mathcal{L}^{-1}\left[\frac{2a + b}{s + 1}\right] - \mathcal{L}^{-1}\left[\frac{a + b}{s + 2}\right] = \\ &= (2a + b)e^{-t} - (a + b)e^{-2t} \quad t \geq 0 \end{aligned} \quad (4.19)$$

pri čemer sta a in b znana začetna pogoja . □

Primer 4.2

$$u(t) = 1(t), \quad U(s) = \frac{1}{s} \quad (4.20)$$

$$\ddot{y}(t) + 4\dot{y}(t) + 3y(t) = 2\dot{u}(t) + 3u(t) \quad (4.21)$$

$$y(0) = 0 \quad \dot{y}(0) = 0 \quad u(0) = 1 \quad (4.22)$$

$$s^2Y(s) + 4sY(s) + 3Y(s) = 2sU(s) - 2 + 3U(s) \quad (4.23)$$

$$Y(s)(s^2 + 4s + 3) = 2 + \frac{3}{s} - 2$$

$$Y(s) = \frac{3}{s(s^2 + 4s + 3)} = \frac{3}{s(s + 1)(s + 3)} \quad (4.24)$$

$$Y(s) = \frac{k_1}{s} + \frac{k_2}{s + 1} + \frac{k_3}{s + 3} \quad (4.25)$$

$$k_l = (s + s_l)Y(s) \Big|_{s=-s_l} \quad (4.26)$$

$$k_1 = 1, \quad k_2 = -\frac{3}{2}, \quad k_3 = \frac{1}{2}$$

$$Y(s) = \frac{1}{s} - \frac{3}{2} \frac{1}{s + 1} + \frac{1}{2} \frac{1}{s + 3} \quad (4.27)$$

$$y(t) = 1(t) - \frac{3}{2}e^{-t} + \frac{1}{2}e^{-3t} \quad (4.28)$$

□

Primer 4.3

$$\ddot{y}(t) + 5\dot{y}(t) + 6y(t) = 6 \cdot 1(t) \quad (4.29)$$

$$y(0) = 2, \quad \dot{y}(0) = 2 \quad (4.30)$$

$$s^2 Y(s) - sy(0) - \dot{y}(0) + 5(sY(s) - y(0)) + 6Y(s) = \frac{6}{s} \quad (4.31)$$

$$s^2 Y(s) - s \cdot 2 - 2 + 5(sY(s) - 2) + 6Y(s) = \frac{6}{s}$$

$$Y(s)(s^2 + 5s + 6) = 2s + 12 + \frac{6}{s}$$

$$Y(s) = \frac{2s + 12 + \frac{6}{s}}{s^2 + 5s + 6} = \frac{2s^2 + 12s + 6}{s(s+2)(s+3)} \quad (4.32)$$

$$Y(s) = \frac{k_1}{s} + \frac{k_2}{s+2} + \frac{k_3}{s+3} \quad (4.33)$$

$$k_l = (s + s_l)Y(s) \Big|_{s=-s_l} \quad l = 1, 2, 3 \quad (4.34)$$

$$k_1 = 1, \quad k_2 = 5, \quad k_3 = -4$$

$$Y(s) = \frac{1}{s} + \frac{5}{s+2} - \frac{4}{s+3} \quad (4.35)$$

$$y(t) = 1(t) + 5e^{-2t} - 4e^{-3t} \quad (4.36)$$

□

Primer 4.4

$$\ddot{y} + 2\dot{y} + 5y = 3 \quad y(0) = 0, \quad \dot{y}(0) = 0 \quad (4.37)$$

$$s^2 y(s) + 2sY(s) + 5Y(s) = \frac{3}{s} \quad (4.38)$$

$$\begin{aligned} Y(s) &= \frac{3}{s(s^2 + 2s + 5)} = \frac{3}{5} \cdot \frac{1}{s} - \frac{3}{5} \cdot \frac{s+2}{s^2 + 2s + 5} = \\ &= \frac{3}{5} \cdot \frac{1}{s} - \frac{3}{10} \cdot \frac{2}{(s+1)^2 + 2^2} - \frac{3}{5} \cdot \frac{s+1}{(s+1)^2 + 2^2} \end{aligned} \quad (4.39)$$

$$y(t) = \mathcal{L}^{-1}[Y(s)] = \frac{3}{5} - \frac{3}{10} e^{-t} \sin 2t - \frac{3}{5} e^{-t} \cos 2t \quad t \geq 0 \quad (4.40)$$

□

4.4 Prenosne funkcije

Prenosne funkcije predstavljajo pomembno orodje na področju vodenja sistemov in izražajo vhodno-izhodno vedenje neke komponente ali nekega sistema.

Prenosna funkcija linearne, časovno nespremenljivega sistema, ki je opisan z diferencialno enačbo, predstavlja razmerje med Laplace-ovim transformom izhodnega signala (odziva) in Laplace-ovim transformom vhodnega signala (vzbujanja) pri predpostavki, da so vsi začetni pogoji enaki nič.

Linearni časovno nespremenljivi sistem opisuje diferencialna enačba

$$a_0 y^{(n)} + a_1 y^{(n-1)} + \dots + a_{n-1} \dot{y} + a_n y = b_0 u^{(m)} + b_1 u^{(m-1)} + \dots + b_{m-1} \dot{u} + b_m u \quad (4.41)$$

kjer je y izhod sistema, u pa njegov vhod. Če naredimo Laplace-ovo transformacijo enačbe (4.41) ob ničelnih začetnih pogojih, dobimo

$$\begin{aligned} a_0 s^n Y + a_1 s^{n-1} Y + \dots + a_{n-1} s Y + a_n Y &= \\ = b_0 s^m U + b_1 s^{m-1} U + \dots + b_{m-1} s U + b_m U & \end{aligned} \quad (4.42)$$

Če izpostavimo Y na levi strani in U na desni strani in naredimo ustrežni kvocient, dobimo prenosno funkcijo v **polinomski obliki**

$$\begin{aligned} \text{Prenosna funkcija} &= G(s) = \frac{\mathcal{L}[\text{izhod}]}{\mathcal{L}[\text{vhod}]} = \frac{Y(s)}{U(s)} = \\ &= \frac{b_0 s^m + b_1 s^{m-1} + \dots + b_{m-1} s + b_m}{a_0 s^n + a_1 s^{n-1} + \dots + a_{n-1} s + a_n} \end{aligned} \quad (4.43)$$

Z uporabo prenosnih funkcij je možno predstaviti dinamiko sistemov s pomočjo algebrskih enačb v ravnini s . Če je najvišja potenca v imenovalcu enaka n , pravimo, da je sistem n -tega reda.

Prenosne funkcije se veliko uporabljajo tako pri analizi kot pri načrtovanju sistemov vodenja. Za boljše razumevanje dodajmo nekaj komentarjev:

- Prenosna funkcija sistema je matematični model in izraža ustrezne diferencialne enačbe v smislu povezave med vhomom in izhodom. Dobimo jo torej iz diferencialnih enačb ali pa direktno iz postopkov (z orodji) matematičnega ali eksperimentalnega modeliranja.
- Prenosna funkcija je lastnost samega sistema, neodvisna od narave in velikosti vhodnega signala.
- Prenosna funkcija vsebuje zvezo med enotami vhodnih in izhodnih signalov, vendar ne daje nobene informacije o fizikalni naravi sistema (prenosne funkcije fizikalno zelo različnih sistemov so lahko identične).
- Če je znana prenosna funkcija sistema, lahko proučujemo odvisnost med različnimi oblikami vhodnih signalov in ustreznim izhodnim signalom, kar izdatno pripomore k boljšemu razumevanju sistema. Velja enačba $Y(s) = G(s)U(s)$, pri čemer je Y izhod in U vhod v sistem.
- Ojačenje je pomemben parameter modelov. Pri proporcionalnih procesih določa vrednost izhodnega signala v ustaljenem stanju pri konstantni vrednosti vhodnega signala. Iz prenosne funkcije ga določimo z limito

$$K = \lim_{s \rightarrow 0} G(s) \quad (4.44)$$

Iz polinomske oblike 4.43 in enačbe 4.44 sledi, da je ojačenja $K = \frac{b_m}{a_n}$.

Razen polinomskega zapisa prenosne funkcije (enačba (4.43)) uporabljamo tudi dve faktorizirani obliki. **Prva faktorizirana oblika** je zapisana s pomočjo polov ($s = p_1, p_2, \dots, p_n$) in ničel ($s = z_1, z_2, \dots, z_m$) in multiplikativne konstante k

$$G(s) = \frac{k(s - z_1)(s - z_2) \cdots (s - z_m)}{(s - p_1)(s - p_2) \cdots (s - p_n)} \quad (4.45)$$

Poli in ničle so lahko v splošnem kompleksni. Omenimo naj, da k ni neposredno ojačenje sistema, ki pomeni relacijo med ustaljeno vrednostjo izhodnega in vhodnega signala. S pomočjo enačbe 4.44 velja $K = \frac{k(-z_1)(-z_2) \cdots (-z_m)}{(-p_1)(-p_2) \cdots (-p_n)}$.

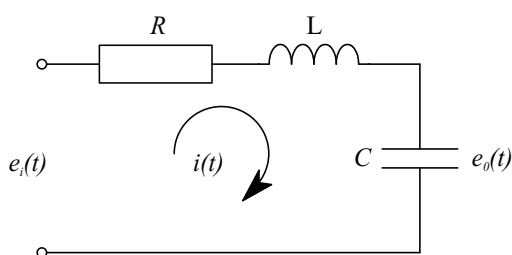
Druga faktorizirana oblika nazorno prikazuje časovne konstante sistema ($T_{a_1}, T_{a_2}, \dots, T_{a_n}$). Če so vsi poli in ničle realni in enkratni, ima obliko

$$G(s) = \frac{K(T_{b_1}s + 1)(T_{b_2}s + 1) \cdots (T_{b_m}s + 1)}{(T_{a_1}s + 1)(T_{a_2}s + 1) \cdots (T_{a_n}s + 1)} \quad (4.46)$$

kjer K predstavlja ojačenje sistema ($G(0) = K$), kajti vsi ostali multiplikativni gradniki imajo ojačenje 1. V kolikor pa ime sistem kompleksne pole in/ali ničle, potem zapišemo en konjugirano kompleksni par s pomočjo kvadratnega člena $\frac{1}{1+2\zeta\left(\frac{s}{\omega_n}\right)+\left(\frac{s}{\omega_n}\right)^2}$.

V vodenju sistemov uporabljamo prvo faktorizirano obliko pri analizi in sintezi s pomočjo diagrama lege korenov, drugo pa pri analizi in sintezi v frekvenčnem prostoru (uporaba Bodejevega in polarnega diagrama).

Primer 4.5 Slika 4.2 prikazuje serijsko RLC vezje



Slika 4.2: Serijsko RLC vezje

Z uporabo prvega Kirchofovega zakona pridemo do enačbe

$$e_i(t) = R i(t) + L \frac{di(t)}{dt} + \frac{1}{C} \int i(t) dt \quad (4.47)$$

Z Laplace-ovo transformacijo enačbe (4.47) dobimo ob upoštevanju ničelnih začetnih pogojev enačbo

$$E_i(s) = \left(R + Ls + \frac{1}{Cs} \right) I(s) \quad (4.48)$$

Če nas zanima prenosna funkcija med tokom $i(t)$ ter vhodno napetostjo $e_i(t)$, naredimo ustrezno deljenje v enačbi (4.48)

$$G_1(s) = \frac{I(s)}{E_i(s)} = \frac{1}{R + Ls + \frac{1}{Cs}} = \frac{Cs}{1 + RCs + LCs^2} = \frac{\frac{1}{L}s}{s^2 + \frac{R}{L}s + \frac{1}{LC}} \quad (4.49)$$

Če za izhodno veličino izberemo napetost na kondenzatorju, velja

$$e_o(t) = \frac{1}{C} \int i dt \quad (4.50)$$

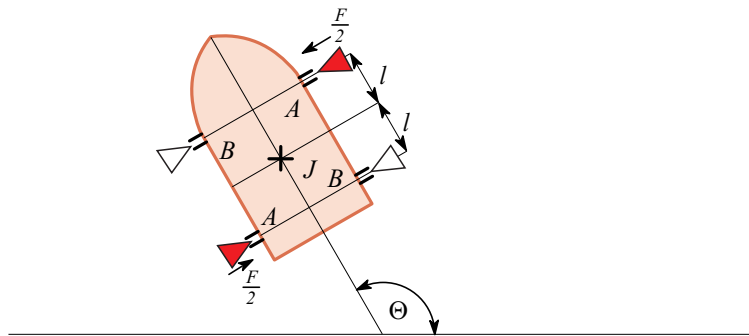
$$E_o(s) = \frac{1}{Cs} I(s) \quad (4.51)$$

oz.

$$G_2(s) = \frac{E_o(s)}{E_i(s)} = \frac{E_o(s)}{I(s)} \cdot \frac{I(s)}{E_i(s)} = \frac{\frac{1}{LC}}{s^2 + \frac{R}{L}s + \frac{1}{LC}} \quad (4.52)$$

□

Primer 4.6 Slika 4.3 prikazuje problem regulacije smeri leta nekega satelita.



Slika 4.3: Regulacija smeri leta satelita

Problem je poenostavljen in kaže regulacijo le v eni ravnini. Za regulacijo smeri leta Θ uporabljamo štiri majhne raketne motorje, od katerih dva (A in A ter B in B) delata v paru. Delovanje motorjev povzroči vrtenje okoli težišča. Vsak motor povzroča moment $\frac{F}{2} \cdot l$, oz. en sinhronizirani par moment $F \cdot l = T$. Moment je v splošnem funkcija časa, zato ga zapišemo kot $T(t)$. Vztrajnostni moment satelita označimo z J . Zanima nas zveza med spremembo kota $\Theta(t)$ in vzbujačnim momentom $T(t)$, oz. ustrežna prenosna funkcija. Z uporabo drugega Newtonovega zakona ob zanemaritvi dušenja pridemo do diferencialne enačbe

$$J \frac{d^2\Theta(t)}{dt^2} = T(t) \quad (4.53)$$

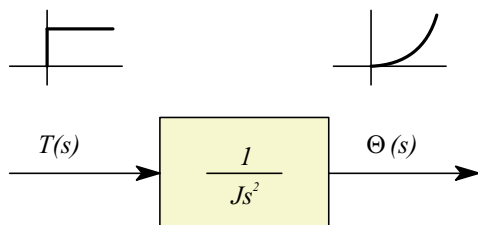
in z Laplace-ovo transformacijo do enačbe

$$Js^2\Theta(s) = T(s) \quad (4.54)$$

Prenosna funkcija je kvocient Laplace-ovih transformov izhodnega in vhodnega signala:

$$G(s) = \frac{\Theta(s)}{T(s)} = \frac{1}{Js^2} \quad (4.55)$$

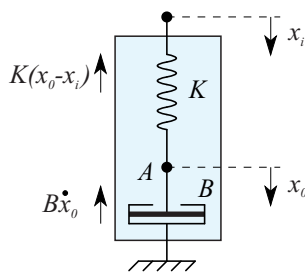
Ustrezno bločno shemo prikazuje slika 4.4. Prikazana sta tudi tipična vhodni in izhodni signal.



Slika 4.4: Bločna shema modela satelita

□

Primer 4.7 Slika 4.5 prikazuje mehanski sistem. Zanima nas prenosna funkcija med vhodnim signalom x_i in izhodnim signalom x_o .



Slika 4.5: Mehanski sistem

Velja enačba, da je vsota vseh sil v točki A enaka nič. Ker sta obe sili usmerjeni obratno od smeri pomika x_o , imata negativni predznak

$$\sum F_i = 0 \quad (4.56)$$

oz.

$$-B \dot{x}_0 - K(x_0 - x_i) = 0 \quad (4.57)$$

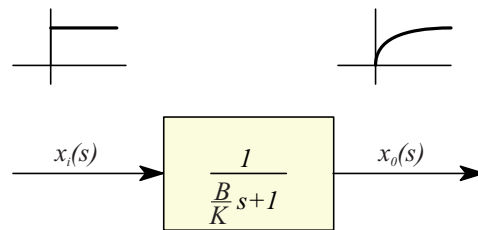
Z ustrežno Laplace-ovo transformacijo pridemo do enačb

$$B s X_o(s) + K X_o(s) = K X_i(s) \quad (4.58)$$

oz.

$$\frac{X_o(s)}{X_i(s)} = \frac{1}{\frac{B}{K} s + 1} \quad (4.59)$$

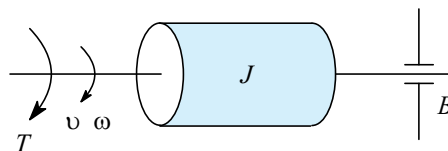
Bločno shemo s tipičnim vhodnim in izhodnim signalom prikazuje slika 4.6.



Slika 4.6: Bločna shema mehanskega sistema

□

Primer 4.8 Slika 4.7 predstavlja rotacijski mehanski sistem. Zanima nas prenosna funkcija med vzbujalnim momentom $T(t)$ in kotno hitrostjo $\omega(t)$.



Slika 4.7: Rotacijski mehanski sistem

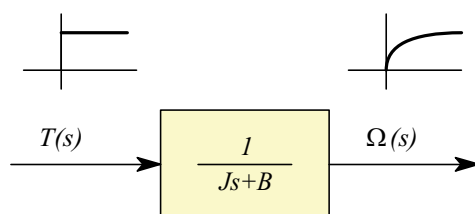
Ker velja diferencialna enačba

$$J \frac{d\omega(t)}{dt} + B\omega(t) = T(t) \quad (4.60)$$

povezuje vzbujalni moment $T(t)$ in kotno hitrost $\omega(t)$ relacija

$$\frac{\Omega(s)}{T(s)} = \frac{1}{Js + B} \quad (4.61)$$

Ustrezno bločno shemo prikazuje slika 4.8.



Slika 4.8: Bločna shema rotacijskega mehanskega sistema

□

4.5 Bločni diagrami

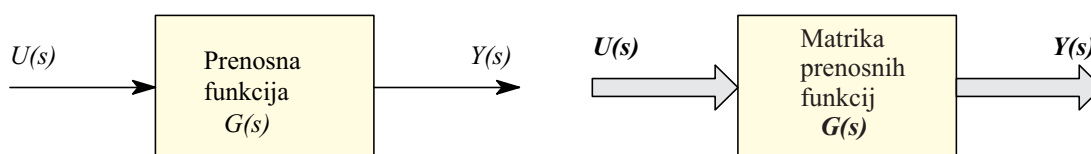
4.5.1 Splošne značilnosti bločnih diagramov

Sistem vodenja lahko sestoji iz številnih komponent. Za nazorno predstavitev funkcij, ki jih v sistem vnašajo posamezne komponente, se v inženirski praksi pogosto uporabljajo bločni diagrami. Zlasti pri linearnih sistemih je možno bločne sheme z uporabo ustreznih pravil poenostavljati. Tako iz kompleksnih shem pridemo do povezav, ki jasno kažejo prispevek določenega vhoda na izhod ali prispevek določene komponente bločnega diagrama na celotno vedenje sistema vodenja. Dandanes večina sodobnih paketov za računalniško podprto načrtovanje sistemov vodenja omogoča podajanje modelov s pomočjo bločnih diagramov za ustrezno nadaljnje procesiranje (npr. simulacija, linearizacija, ...). Inženirjem so bločni diagrami bolj nazorni kot kakršne koli sheme, ki so sicer bliže

fizikalnemu sistemu. Potrebno je poudariti, da bločni diagram vsebuje informacijo o dinamičnem obnašanju sistema, ne vsebuje pa nobene informacije o fizikalni konstrukciji sistema. Fizikalno zelo različni sistemi imajo lahko enake ali podobne bločne diagrame. Z bločnim diagramom se sistem ne da opisati enolično. Vsak sistem je namreč možno opisati z več ekvivalentnimi bločnimi diagrami, ki so različno primerni za različne oblike analize.

V bločnem diagramu so vse spremenljivke sistema povezane med seboj s t.i. funkcionalnimi bloki ali kratko bloki. Blok je simbol, ki ponazarja zvezo med njegovim vhomom in izhodom. Često je podan s prenosno funkcijo. Puščici na vhodu in izhodu določata smer signala. Signal torej poteka le v smeri puščic.

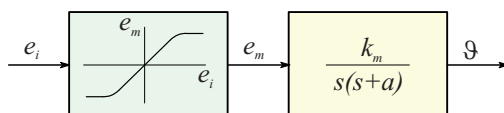
Slika 4.9 prikazuje temeljni element bločnega diagrama - funkcionalni blok. Vhodni ali izhodni signal sta lahko enodimenzionalna (pri univariabilnem sistemu)



Slika 4.9: Funkcionalni blok: a) univariabilni
b) multivariabilni

ali večdimenzionalna (pri multivariabilnem sistemu). Za večdimenzionalne signale je precej v uporabi risanje signalov, kot prikazuje slika 4.9b. Namesto prikazanih povezav lahko uporabljamo tudi poudarjene povezave. Seveda je funkcionalni blok pri linearnih sistemih lahko predstavljen tudi z drugimi možnimi zapisi, ki jih poznamo v teoriji sistemov in ne le s prenosnimi funkcijami (npr. zapis v prostoru stanj, diferencialne enačbe).

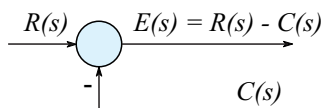
Prav tako uporaba bločnih diagramov ni omejena na linearne sisteme, ampak je povsem splošna. Slika 4.10 prikazuje tipičen nelinearni sistem z ojačevalnikom in rotorsko vzbujanim enosmernim motorjem.



Slika 4.10: Primer uporabe bločnih diagramov pri nelinearnih sistemih

Ojačevalnik je tipični nelinearni element, saj pri določeni vhodni napetosti e_i njegov izhod e_m pride v nasičenje.

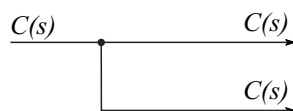
Razen funkcionalnih blokov nastopajo v bločnih diagramih, ki opisujejo linearne sisteme, še sumacijske točke in razcepišča. Sumacijska točka ima obliko kroga, v katerega vstopata dva signala, eden pa izstopa. Prikazuje jo slika 4.11. Omogoča



Slika 4.11: Sumacijska točka

seštevanje ali odštevanje dveh signalov. V bližini, kjer signala vstopata v sumacijsko točko, napišemo znaka + ali - in s tem povemo, ali se signal prišteje ali odšteje. Pozitivni znak običajno izpuščamo.

Razcepišče v bločnem diagramu uporabljamo, če mora posamezni signal sočasno delovati na različnih mestih v bločnem diagramu. Način risanja prikazuje slika 4.12.

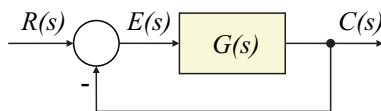


Slika 4.12: Razcepišče

Za opis nelinearnih sistemov potrebujemo tudi nekatere druge elemente kot so npr. množilnik, delilnik in člen za korenjenje.

Bločni diagram zaprtozančnega sistema

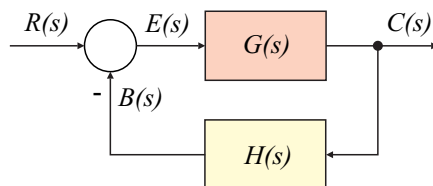
Slika 4.13 prikazuje bločni diagram zaprtozančnega sistema. Pri tem $G(s)$ pred-



Slika 4.13: Bločni diagram zaprtozančnega sistema

stavlja skupno prenosno funkcijo regulatorja in procesa. Bločni diagram jasno prikazuje, da pogrešek $E(s)$ kot razlika med referenčno in regulirano veličino, deluje na sistem $G(s)$. Izhod $C(s)$ dobimo z množenjem prenosne funkcije $G(s)$

z vhomom $E(s)$. Takemu zaprtozančnemu sistemu, ki v povratni zanki ne vsebuje nobenega elementa, pravimo zaprtozančni sistem z enotino povratno zanko. V praksi pa je potrebno običajno regulirano veličino pretvoriti v neko drugo veličino, preden jo primerjamo z referenčno veličino (npr. sprememba temperature v napetost, filtriranje, ...). Omenjeno pretvorbo opravi prenosna funkcija $H(s)$. Ustrezno povratnozančno strukturo prikazuje slika 4.14.



Slika 4.14: Zaprtozančni sistem z elementom $H(s)$

Torej $H(s)$ lahko predstavlja prenosno funkcijo tipala in merilnega pretvornika ali pa predstavlja regulator v povratni zanki. Signal, ki se primerja z referenčnim signalom, je torej $B(s) = H(s)C(s)$.

Glede na sliko 4.14 bomo definirali:

- prenosno funkcijo direktne veje,
- odprtozančno prenosno funkcijo in
- zaprtozančno prenosno funkcijo.

Prenosna funkcija direktne veje je kvocient med regulirano veličino $C(s)$ in pogreškom $E(s)$:

$$\text{prenosna funkcija direktne veje} = \frac{C(s)}{E(s)} = G(s) \quad (4.62)$$

Odprtozančna prenosna funkcija je kvocient med povratnozančnim signalom $B(s)$ in signalom pogreška $E(s)$:

$$\text{odprtozančna prenosna funkcija} = \frac{B(s)}{E(s)} = G(s)H(s) \quad (4.63)$$

V primeru enotine povratne zanke sta prenosna funkcija direktne veje in odprtozančna prenosna funkcija enaki.

Zaprtozančna prenosna funkcija je kvocient med regulirano veličino $C(s)$ in referenčno veličino $R(s)$. Zvezo izpeljemo s pomočjo naslednjih enačb, ki so razvidne iz slike 4.14

$$C(s) = G(s) E(s) \quad (4.64)$$

$$B(s) = H(s) C(s) \quad (4.65)$$

$$E(s) = R(s) - B(s) = R(s) - H(s) C(s) \quad (4.66)$$

Če iz enačb (4.64) in (4.66) izločimo $E(s)$, dobimo

$$C(s) = G(s) [R(s) - H(s) C(s)] \quad (4.67)$$

oziroma

$$\text{zaprtozančna prenosna funkcija} = \frac{C(s)}{R(s)} = \frac{G(s)}{1 + G(s) H(s)} \quad (4.68)$$

S pomočjo zaprtozančne prenosne funkcije izračunamo regulirano veličino s pomočjo enačbe

$$C(s) = \frac{G(s)}{1 + G(s) H(s)} R(s) \quad (4.69)$$

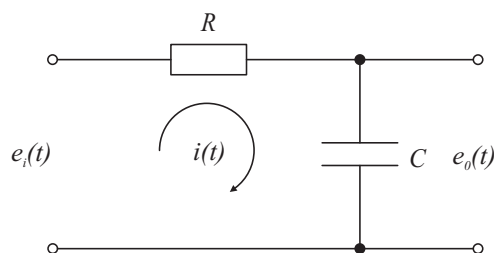
Odziv je torej odvisen od zaprtozančne prenosne funkcije in od vhodnega signala $r(t)$.

4.5.2 Postopek za risanje bločnih diagramov

Bločni diagrami predstavljajo grafično predstavitev enačb, ki jih dobimo v postopku matematičnega modeliranja. Ker pa omenjeni postopek presega okvir tega dela, bomo k risanju pristopili bolj intuitivno.

Izhajamo iz enačb, v katerih upoštevamo fizikalne zakone (enačbe masnega in energijskega ravnotežja, Kirchofovi zakoni, ...). Pomembno je tudi, da pravilno izberemo vhode in izhode sistema, t.j. kaj so vplivne veličine in kaj želimo opazovati. Pri preoblikovanju tako dobljenih osnovnih enačb vedno začnemo z enačbami, ki opisujejo 'shranjevalnike energije' (npr. tuljava, kondenzator, ...). V teh enačbah izrazimo na levi strani tiste veličine, ki imajo t.i. vztrajnost (npr. tok skozi tuljavo, napetost na kondenzatorju, ...). To so stanja sistema. Matematično se ta preureditev kaže tako, da dobimo na desni strani enačb operator integriranja. Bločna predstavitev preurejenih enačb torej vsebuje funkcionalne bloke integratorje. Ostale enačbe preoblikujemo sproti med risanjem, ko nazorno vidimo, katere spremenljivke nam bločni diagram že ponuja in katere spremenljivke je potrebno izraziti iz preostalih enačb. Definiramo t.i. vzročnost, ki je zgolj matematična in ne fizikalna lastnost.

Primer 4.9 Narišimo bločni diagram RC vezja, ki ga prikazuje slika 4.15.



Slika 4.15: RC vezje

Ustrezni enačbi, ki opisujeta sistem, sta

$$e_o = \frac{Q}{C} = \frac{\int i dt}{C} \quad (4.70)$$

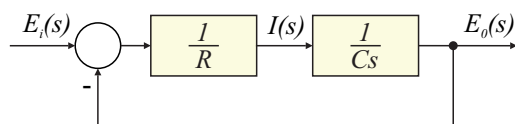
$$i = \frac{e_i - e_o}{R} \quad (4.71)$$

in ustrezna Laplace-ova transformata

$$E_o(s) = \frac{I(s)}{Cs} \quad (4.72)$$

$$I(s) = \frac{E_i(s) - E_o(s)}{R} \quad (4.73)$$

Prva enačba predstavlja blok $\frac{1}{Cs}$ (integrator z ojačenjem $\frac{1}{C}$). Vhod v blok $I(s)$ določa druga enačba. Le-to izvedemo s pomočjo sumacijske točke in bloka z ojačenjem $\frac{1}{R}$. Celotni bločni diagram prikazuje slika 4.16.



Slika 4.16: Bločni diagram RC vezja (uporabljen operator integriranja)

Če ne bi upoštevali navedenih napotkov, bi lahko namesto napetosti izrazili tok skozi tuljavo

$$i = C \frac{de_o}{dt} \quad (4.74)$$

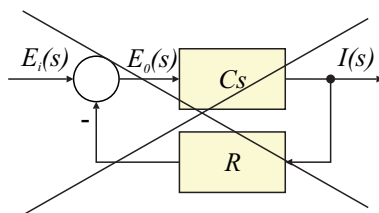
$$e_o = e_i - iR \quad (4.75)$$

oz.

$$I(s) = CsU_C(s) \quad (4.76)$$

$$E_o(s) = E_i(s) - I(s)R \quad (4.77)$$

Ustrezni bločni diagram prikazuje slika 4.17.

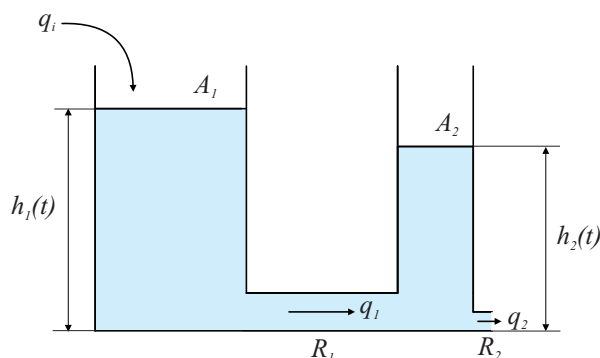


Slika 4.17: Bločni diagram RC vezja (uporabljen operator diferenciranja)

Bločni diagram, ki ga prikazuje slika 4.17, je v smislu dobrega modeliranja in simulacije nesprijemljiv. Tako npr. onemogoča numerično ugodno realizacijo (simulacijo) modela na računalniku kakor tudi konsistentno vključitev začetnih pogojev.

□

Primer 4.10 Poiščimo bločni diagram hidravličnega sistema, ki ga prikazuje slika 4.18.



Slika 4.18: Hidravlični sistem z dvema nivojskima posodama

Vhodni pretok je označen z $q_i(t)$, nivo v prvi posodi s $h_1(t)$, površina gladine v prvi posodi pa je A_1 . Pretok med obema posodama je $q_1(t)$, $h_2(t)$ je nivo v drugi posodi, A_2 je površina gladine v drugi posodi, $q_2(t)$ pa je iztok iz druge posode. Za prvo posodo velja ravnotežna enačba

$$A_1 \frac{d h_1(t)}{dt} = q_i(t) - q_1(t) \quad (4.78)$$

Pretok med obema posodama je proporcionalen razliki obeh nivojev

$$q_1(t) = \frac{h_1(t) - h_2(t)}{R_1} \quad (4.79)$$

Ravnotežna enačba za drugo posodo je

$$A_2 \frac{d h_2(t)}{dt} = q_1(t) - q_2(t) \quad (4.80)$$

iztok pa je odvisen od nivoja h_2

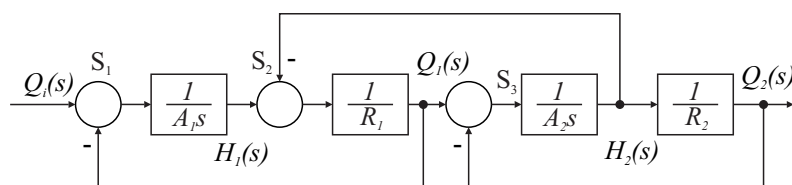
$$q_2(t) = \frac{h_2(t)}{R_2} \quad (4.81)$$

Z Laplace-ovo transformacijo gornjih enačb dobimo

$$\begin{aligned} A_1 s H_1(s) &= Q_i(s) - Q_1(s) \Rightarrow H_1(s) = \frac{1}{A_1 s} [Q_i(s) - Q_1(s)] \\ Q_1(s) &= \frac{H_1(s) - H_2(s)}{R_1} \end{aligned} \quad (4.82)$$

$$\begin{aligned} A_2 s H_2(s) &= Q_1(s) - Q_2(s) \Rightarrow H_2(s) = \frac{1}{A_2 s} [Q_1(s) - Q_2(s)] \\ Q_2(s) &= \frac{H_2(s)}{R_2} \end{aligned} \quad (4.83)$$

Iz obeh enačb masnih shranjevalnikov smo izrazili nivoja H_1 in H_2 , ker sta to veličini z vstrajnostjo (ne moreta se hipoma spremeniti). Če povežemo bloke ob upoštevanju gornjih enačb, dobimo shemo, ki jo prikazuje slika 4.19.



Slika 4.19: Bločni diagram hidravličnega sistema

□

4.5.3 Poenostavljanje bločnih diagramov

Kompleksne bločne diagrame, ki so sestavljeni iz večih povratnih zank, je možno poenostaviti z uporabo pravil algebre bločnih shem. Pri tem se zveza med vhodi in izhodi sistema ne spremeni. Pravila za poenostavljanje prikazuje tabela 4.2.

Do enakega rezultata lahko pridemo na različne načine. Postopek običajno začnemo s premikom sumacijske točke ali razcepišča tako, da dobimo strukture, ki jih je možno poenostaviti s pravili redukcije vzporedne ali zaporedne vezave in zanke.

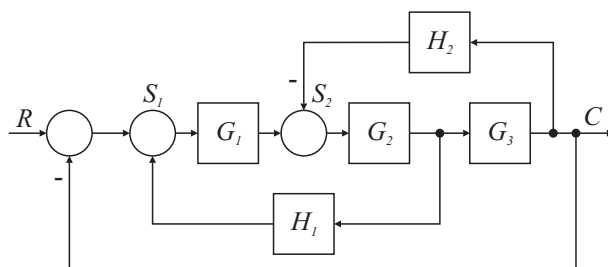
Algebra bločnih shem pa ni vedno namenjena poenostavljanju. Včasih namreč želimo spremeniti strukturo v obliko, ki nam za določen namen bolj odgovarja.

Tabela 4.2: Pravila algebre bločnih shem

1. Zamenjava vrstnega reda		
2. Zamenjava sumacijskih točk (S)		
3. Zamenjava razcepišč (R)		
4. Premik S pred blok		
5. Premik S za blok		
6. Premik R pred blok		
7. Premik R za blok		
8. Premik R pred S		
9. Premik R za S		

10. Redukcija zaporedne vezave		
11. Redukcija vzporedne vezave		
12. Redukcija zanke		
13. Odstranitev bloka iz direktne veje		
14. Vstavitve bloka v direktno vejo		
15. Odstranitev bloka iz povratne zanke		
16. Vstavitve bloka v povratno zanko		
17. Razširitev bloka v zanko		

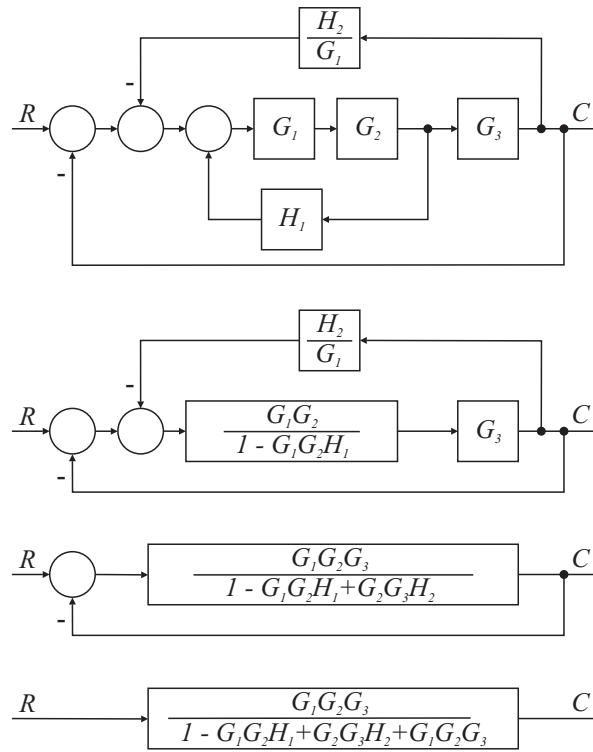
Primer 4.11 Poenostavimo bločni diagram na sliki 4.20.



Slika 4.20: Bločni diagram

Bistveni korak pri rešitvi sheme je v tem, da sumacijsko točko S_2 prestavimo pred blok G_1 (pravilo 4) in pred sumacijsko točko S_1 (pravilo 2). Nato uporabljamo pravila redukcije zaporedne vezave blokov (pravilo 10) ter redukcije zanke (pravilo

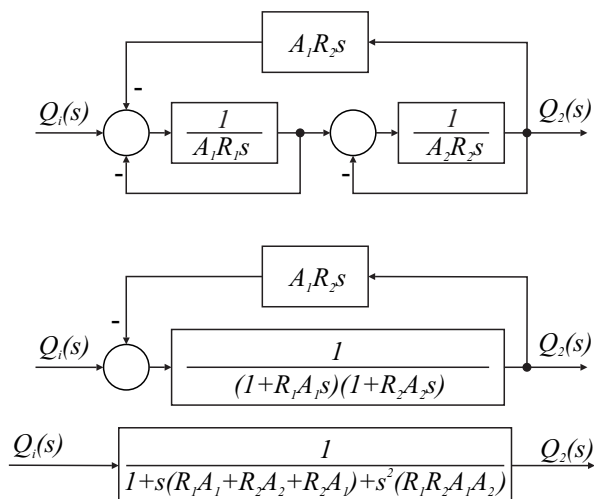
12). Postopek prikazuje slika 4.21.



Slika 4.21: Redukcija bločnega diagrama

□

Primer 4.12 Poenostavimo bločni diagram iz primera 4.10 (slika 4.19). Postopek prikazuje slika 4.22.



Slika 4.22: Redukcija bločnega diagrama hidravličnega procesa

Najprej prestavimo S_2 pred blok $\frac{1}{A_1 s}$ (pravilo 4) ter razcepišče R za blok $\frac{1}{R_2}$ (pravilo 7). Nato uporabimo pravilo redukcije zaporedne vezave (pravilo 10). Postopek prikazuje prvi diagram na sliki 4.22. Da dobimo drugi diagram, uporabimo pravilo redukcije zanke (pravilo 12) in pravilo redukcije zaporedne vezave (pravilo 10). Tretji diagram pa dobimo tako, da še enkrat upoštevamo pravilo redukcije zanke (pravilo 12). \square

5.

Računalniško podprta analiza in simulacija sistemov

Dandanes si ne moremo več predstavljati uspešne analize in načrtovanja sistemov in vodenja brez ustrezne računalniške podpore. Najbolj je v rabi okolje Matlab z mnogimi dodatki. Na tem mestu si bomo ogledali dodatek **Control Toolbox**, ki je namenjen predvsem obravnavi linearnih sistemov. Nelinearne sisteme pa je možno učinkovito proučevati s simulacijo in z Matlabovim okoljem Simulink. Metode in orodje bomo spoznali v naslednjem poglavju. Ostali paketi, ki omogočajo podporo pri analizi, simulaciji in načrtovanju pa so: Signal processing, Nonlinear, Robust control, Optimization, Identification, Data acquisition, Fuzzy logic Neural networks, Symbolic Math, ...

5.1 Vnos linearnih dinamičnih sistemov

Linearni dinamični sistem lahko podamo v prostoru stanj ali s prenosno funkcijo.

Prostor stanj

Sistem

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}u(t) \\ y(t) &= \mathbf{C}\mathbf{x}(t) + \mathbf{D}u(t)\end{aligned}$$

podamo v Matlabu s pomočjo matrik \mathbf{A} , \mathbf{B} , \mathbf{C} in \mathbf{D} , nato pa uporabimo funkcijo `ss`

`sys=ss(A,B,C,D)`

Prenosna funkcija v polinomski obliki

Prenosno funkcijo

$$G(s) = \frac{B(1)s^{m-1} + B(2)s^{m-2} + \dots + B(m-1)s + B(m)}{A(1)s^{n-1} + A(2)s^{n-2} + \dots + A(n-1)s + A(n)}$$

podamo v Matlabu tako, da definiramo dva vektorja, ki vsebujeta koeficiente števca in imenovalca

`B=[B(1) B(2) ... B(m-1) B(m)]`
`A=[A(1) A(2) ... A(n-1) A(n)]`

in uporabimo ukaz

`sys=tf(B,A)`

Prenosna funkcija v faktorizirani obliki

Prenosno funkcijo

$$G(s) = k \frac{(s - z(1))(s - z(2)) \dots + (s - z(m))}{(s - p(1))(s - p(2)) \dots + (s - p(n))}$$

podamo v Matlabu tako, da definiramo dva vektorja, ki vsebujeta pole in ničle prenosne funkcije (korene števca in imenovalca) in multiplikativno konstanto k

`z=[z(1) z(2) ... z(m-1) z(m)]`
`p=[p(1) p(2) ... p(n-1) p(n)]`

Uporabimo ukaz

`sys=zpk(z,p,k)`

Prevedbe med zapisi

Funkcije `ss`, `tf` in `zpk` uporabljamo tudi za poljubne medsebojne prevedbe

Prevedba v prostor stanj (**ss**) iz oblike **tf** ali **zpk**

```
sys_ss=ss(sys)
```

Prevedba v polinomsko obliko prenosne funkcije (**tf**) iz oblike **ss** ali **zpk**

```
sys_tf=tf(sys)
```

Prevedba v faktorizirano obliko (**zpk**) iz oblike **ss** ali **tf**

```
sys_zpk=zpk(sys)
```

5.2 Osnovne zmožnosti dela s sistemi

Izračun polov

```
poli = pole(sys)
```

Izračun ničel

```
nicle = zero(sys)
```

Izris grafa lege polov in ničel v ravnini *s*

```
pzmap(sys)
```

Določitev lastnih vrednosti (polov), dušilnega koeficienta, lastne frekvence

```
damp(sys)
```

```
[wn,z]=damp(sys)
```

```
[wn,z,p]=damp(sys)
```

Zaporedna vezava sistemov *sys1* in *sys2*

```
sys=series(sys1,sys2)
```

Vzporedna vezava sistemov *sys1* in *sys2*

```
sys=parallel(sys1,sys2)
```

Izračun zaprtozankne prenosne funkcije (sys1 v direktni veji, sys2 v povratni zanki)

```
sys=feedback(sys1,sys2,-1)    negativna povratna zanka
sys=feedback(sys1,sys2,+1)    pozitivna povratna zanka
```

5.3 Simulacija s pomočjo Matlabovih funkcij

Razen simulacije v Simulinku nudi Matlab in dodatek Control Toolbox še nekaj drugih možnosti:

- odziv linearnih sistemov na enotino stopnico s pomočjo funkcije `step` ali odziv na delta impulz s pomočjo funkcije `impulse`,
- simulacijo linearnih časovno nespremenljivih sistemov s pomočjo funkcije `lsim`,
- simulacijo linearnih in nelinearnih, časovno nespremenljivih in časovno spremenljivih sistemov s pomočjo vgrajenih funkcij za numerično integracijo,
- simulacijo Simulink modela iz okolja Matlab s pomočjo funkcije `sim`.

Določitev odziva linearnega sistema s funkcijama `step` in `impulse`

Funkcija `step` izračuna odziv na enotino stopnico linearnega sistema (LTI), ki je opisan s prenosno funkcijo v polinomski (`tf`) ali faktorizirani obliki (`zpk`) ali v prostoru stanj (`ss`).

```
step(sys)
```

izračuna in nariše odziv sistema `sys` na enotino stopnico. Časovno območje in časovni korak se izbereta avtomatsko.

```
step(sys,tf)
```

izračuna in nariše odziv na časovnem intervalu `[0 tf]`.

```
step(sys,t)
```

izračuna in nariše odziv v trenutkih, ki jih določa vektor `t`, npr. `t=0:dt:tf`

`y=step(sys,t)`

izračuna odziv y v trenutkih, ki jih določa vektor t , npr. $t=0:dt:tf$. Funkcija ne nariše odziva, pač pa ga lahko narišemo z ukazom `plot(t,y)`.

`[y,t]=step(sys)`

izračuna odziv y v trenutkih, ki jih določa t . Funkcija ne nariše odziva, pač pa ga lahko narišemo z ukazom `plot(t,y)`. Časovno območje in časovni korak se izbereta avtomatsko.

`[y,t]=step(sys,tf)`

izračuna odziv y v trenutkih, ki jih določa t (trenutke določi računalnik, ne uporabnik!) na časovnem intervalu $[0 \text{ } tf]$. Funkcija ne nariše odziva, pač pa ga lahko narišemo z ukazom `plot(t,y)`.

`[y,t,x]=step(sys)`

funkcija vrne še stanja x , kar je predvsem uporabno pri sistemih v prostoru stanj.

Na podoben način se uporablja funkcija `impulse` za določitev odziva na delta impulz.

Primer 5.1 Primer prikazuje izračun in izris odziva prenosne funkcije sistema 2. reda z ojačenjem 2, lastno frekvenco 1 in dušilnim koeficientom 1 na enotino stopnico. Čas opazovanja je 10s. Program v Matlabu

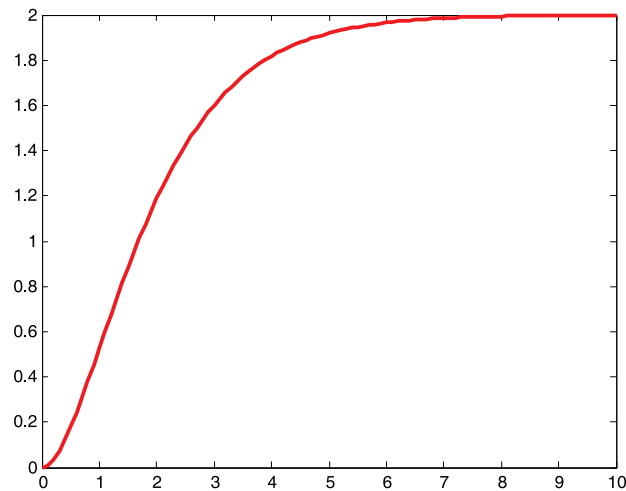
```
G=tf([2],[1 2 1]); tf=10; [y,t]=step(G,tf); plot(t,y)
```

nariše sliko 5.1.

□

Simulacija s funkcijo `lsim`

S funkcijo `lsim` simuliramo zvezne (ali diskretne) dinamične sisteme, ki jih v Matlab vnesemo s funkcijami `ss` (prostor stanj), `tf` (prenosna funkcija v polinomski obliki) in `zpk` (prenosna funkcija v faktorizirani obliki). To so t.i. LTI modeli (linear time invariant model). Vhodni signal definiramo s pomočjo numerično podane funkcijske odvisnosti (s točkami podana funkcija).

Slika 5.1: Odziv s pomočjo funkcije `step`

Brez argumentov na levi strani enačaja lahko uporabljamo naslednji obliki:

```
lsim(sys,u,t)
lsim(sys,u,t,x0)
```

Funkcija `lsim` na zaslonu nariše odziv LTI modela. Pri tem je `sys` sistem, ki ga podamo s funkcijami `ss`, `tf` ali `zpk`. Par `u` in `t` je sestavljen iz dveh vektorjev, ki določata vhodni signal $u(t)$. `t` torej določa trenutke, v katerih je določen vhodni signal, `u` pa so vrednosti signala v ustreznih trenutkih. Trenutke lahko definiramo z naslednjo Matlabovo definicijo vektorja:

```
t = 0:dt:t_koncni
```

sinusni signal pa z

```
u=sin(t)
```

Vektorja `u` in `t` morata imeti enako število elementov (`length(t)=length(u)`).

Med dvema zaporednima točkama vhodnega signala se upošteva linearna interpolacija.

`x0` je matrika začetnih stanj (vsaka kolona opisuje časovni potek enega stanja). Predvsem se začetna stanja uporabljajo v zvezi z modelom v prostoru stanj.

Če kličemo funkcijo `lsim` le z enim argumentom - imenom modela

```
lsim(sys)
```

se odpre uporabniški vmesnik - Linear Simulation Tool. Uporabniški vmesnik zahteva vnos vhodnega signala in začetnih stanj, nakar lahko poženemo simulacijo.

Lahko pa navedemo tudi spremenljivke na levi strani enačaja:

```
y = lsim(sys,u,t)
[y,t] = lsim(sys,u,t)
[y,t,x] = lsim(sys,u,t)
[y,t,x] = lsim(sys,u,t,x0)
```

`y` je odziv LTI sistema `sys`. Če na levi strani enačaja navedemo `t`, potem se v določenih primerih lahko dogodi, da se vektor `t` po simulaciji spremeni (npr. v primeru, če je opisan z malo elementi). `x` je matrika stanj (vsaka kolona opisuje časovni potek enega stanja), `x0` pa matrika začetnih stanj. Slednja argumenta imata pomen predvsem pri zapisu modela v prostoru stanj.

Primer 5.2 Primer prikazuje določitev odziva sistema 2. reda z ojačenjem 2, lastno frekvenco 1 in dušilnim koeficientom 1 na sinusni vhodni signal. Čas opazovanja je 10s. Z uporabo programa v jeziku Matlab

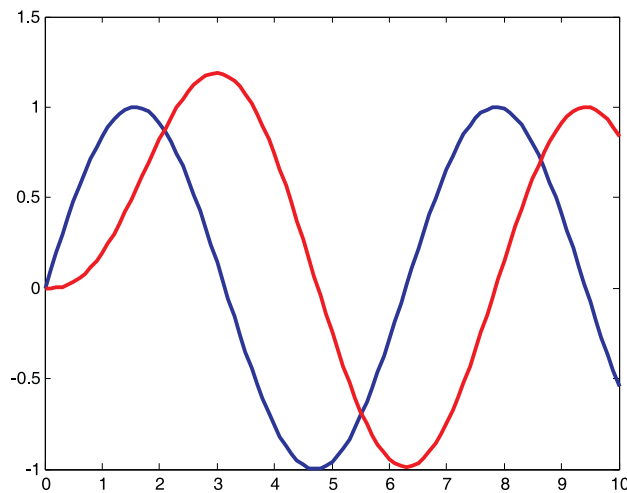
```
G=tf([2],[1 2 1]);t=0:0.1:10; u=sin(t); y=lsim(G,u,t); plot(t,u,t,y)
```

□

dobimo sliko 5.2.

Simulacija s funkcijo `sim`

S funkcijo `sim` zaženemo model, ki je opisan s Simulink shemo. Tako lahko v okolju Matlab programiramo kompleksne eksperimente, ki vključujejo simulacijske teke modela v Simulinku. Ponavadi je koristno, da modelu v Simulinku z out bloki iz knjižnice Sinks označimo izhode oz. signale, ki jih opazujemo.

Slika 5.2: Odziv s pomočjo funkcije `lsim`

`sim('model')`

ukaz simulira Simulink model `model.mdl`. Veljajo krmilni parametri, kot so nastavljeni v Simulink shemi.

`[t,x,y]=sim('model', timespan)`

`t` je vektor, ki vsebuje trenutke, v katerih se izračunajo rezultati simulacije. `x` je matrika stanj (vsaka kolona opisuje eno stanje)- stanja so izhodi integratorjev. `y` je matrika izhodov (vsaka kolona opisuje en izhod, izhodi so določeni z out bloki v Simulink shemi). `timespan` je v splošnem vektor. Če ima vektor en element, je to končni čas simulacije `tf`, če ima dva elementa, sta to začetni in končni čas simulacije `[t0 tf]`, če pa je več elementov, so to trenutki, v katerih želimo dobiti rezultate simulacije `[t0 t1 t2 ... tf]`.

`[t,x,y]=sim('model', timespan, options)`

`options` vsebuje vse mogoče nastavitve, ki se sicer lahko nastavijo v Simulink shemi, npr. tolerance, integracijsko metodo, minimalni in maksimalni dopustni računski korak,... Za nastavitev parametra `options` se uporablja posebna funkcija `simset`. Če želimo izbrati integracijsko metodo `ode45`, uporabimo ukaz

`options=simset('solver','ode45')`.

Primer 5.3 Primer prikazuje parametrizacijo sistema 2. reda pri različnih vrednostih parametra ζ . Parameter ζ spreminjamo od 0 do 2 s korakom 0.1. Odziv sistema določimo s pomočjo funkcije `step in` s pomočjo funkcije `sim`. Slednja

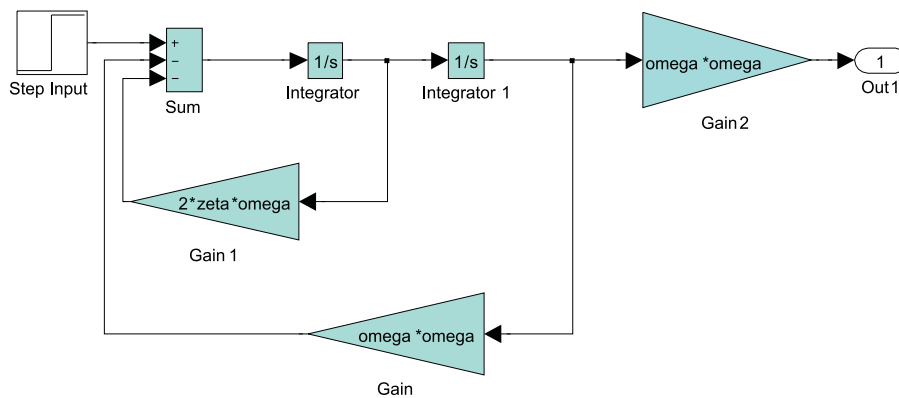
požene model `secor1.mdl` v Simulinku. Prepričamo se, da dobimo po obeh metodah enake rezultate. Program v jeziku Matlab

```

omega=1; tf=20; for zeta=0:0.1:2;
    [t,x,y]=sim('secor1',tf);
    Gs=tf(1,[1 2*zeta*omega omega*omega]);
    [y1,t1]=step(Gs,tf);
    subplot(2,1,1);
    hold on
    plot(t,y,'r');
    subplot(2,1,2);
    plot(t1,y1,'b');
    hold on;
end

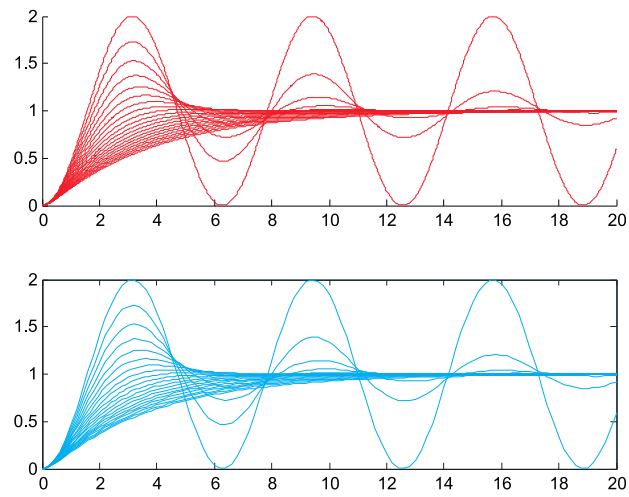
```

uporabi Simulink shemo `secor1.mdl` (slika 5.3) in nariše sliko 5.4.



Slika 5.3: Simulink shema sistema 2.reda

□



Slika 5.4: Parametrizacija sistema 2. reda

6.

Osnovne metode pri reševanju problemov s simulacijo

V tem delu bomo pokazali osnovne metode pri reševanju problemov s simulacijo. *Indirektni postopek* pri reševanju diferencialnih enačb prikazuje bistvo zvezne simulacije. Ogleдали si bomo tudi metodo za simulacijo prenosne funkcije. Postopki nas bodo pripeljali do simulacijskih shem, ki so neodvisne od uporabljenega simulacijskega orodja in predstavljajo osnovo za računalniško simulacijo.

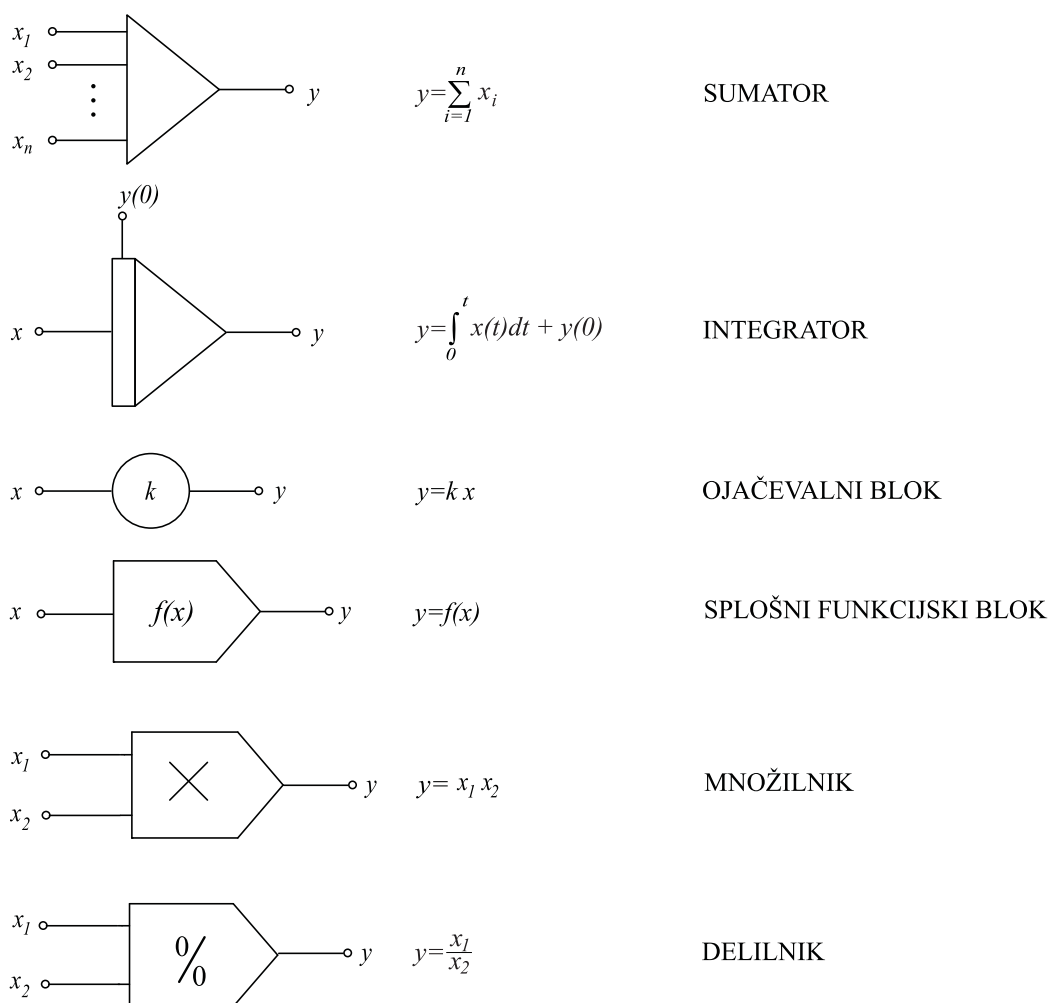
6.1 Simulacijska shema

Osnova za zvezno simulacijo nekega modela je posebna grafična predstavitev, ki jo bomo imenovali *simulacijska shema*. Le-ta ima veliko skupnega z bločnimi diagrami, ki jih zlasti uporabljamo pri zapisu sistemov vodenja. Zato bomo tudi osnovne gradnike simulacijskih shem imenovali *bloke*. Blok predstavlja funkcijo določenega gradnika simulacijske sheme. Grafično predstavitev bloka pa imenujemo *ikona*.

Ker osnove simulacije izvirajo iz konceptov analogne simulacije, izhajajo tudi simulacijske sheme iz analognih simulacijskih shem. Lord Kelvin je imenoval ustrezno shemo "shemo diferencialne analize". Danes se v praksi uporabljajo zelo različne oblike simulacijskih shem tako, da praktično vsak izdelovalec simulacijskega orodja uvede kakšne svoje bloke oz. ikone. S čim manjšim naborom

blokov oz. ikon bomo skušali uvesti shemo, ki bo uporabna za kakršno koli simulacijsko orodje. Tako bodo take sheme uporabne za simulacijo z digitalnim simulacijskim jezikom kot tudi za simulacijo z analognim računalnikom, kjer pa je potrebno dodatno upoštevati, da elementi analognega računalnika obračajo signalom predznak.

Osnovne bloke, ki jih bomo uporabljali v simulacijski shemi, prikazuje slika 6.1. Po potrebi pa bomo kasneje definirali še kakšen nov blok oz. ikono.



Slika 6.1: Pogosto uporabljeni bloki oz. ikone v simulacijski shemi

Predznake, ki učinkujejo v posameznih blokih, lahko vpeljemo preko ustreznih parametrov (npr. predznak konstante k pri ojačevalnem bloku, ki signal x pomnoži s konstanto k), lahko pa ga definiramo v bližini, kjer vhodni signal vstopa v

blok (če predznaka ni, privzamemo pozitivni predznak). Sumator ima poljubno število vhodov, integrator pa le enega. Splošni funkcijski blok je najsplošnejši. Predstavlja lahko vir signalov (takrat običajno ne rišemo vhodnega priključka) ali pa poljubne nelinearne zakonitosti med vhomom in izhodom. V konkretnih shemah namesto $f(x)$ vpišemo v blok ustrezno matematično relacijo (npr. SIN za funkcijo sinus) ali pa vrišemo grafični simbol, ki nazorno pove, za kakšen tip bloka gre. Vlogo bloka lahko nadalje posplošimo, če definiramo vhod in izhod kot vektorska signala ($\mathbf{y} = \mathbf{f}(\mathbf{x})$).

6.2 Indirektna metoda

Ker so matematični modeli dinamičnih sistemov običajno opisani s sistemom diferencialnih enačb, predstavlja indirektna metoda za reševanje diferencialnih enačb osnovni simulacijski pristop. Po tej metodi je potrebno najvišji odvod integrirati tolikokrat, kolikor je njegov red. S tem indirektno generiramo vse nižje odvode in samo spremenljivko. V tem načinu se skriva bistvo simulacije. Analitična rešitev diferencialne enačbe in tabeliranje rešitve v določenih točkah neodvisne spremenljivke nima nobene zveze s simulacijo. Včasih sicer na ta način lahko pridemo hitreje do bolj točnih rezultatov, vendar je v praksi uporabnost takega analitičnega pristopa zelo omejena (npr. le za linearne sisteme).

Indirektna metoda je uporabna, če je možno iz diferencialne enačbe izraziti najvišji odvod in če ne nastopajo višji odvodi vhodnega signala. Indirektno metodo opišimo za sistem

$$y^{(n)} + f(y^{(n-1)}, y^{(n-2)}, \dots, y', y, u; t) = 0 \quad (6.1)$$

y je izhodni signal, u je vhodni signal, t pa je neodvisna spremenljivka simulacije (čas). Postopek opišemo v treh točkah:

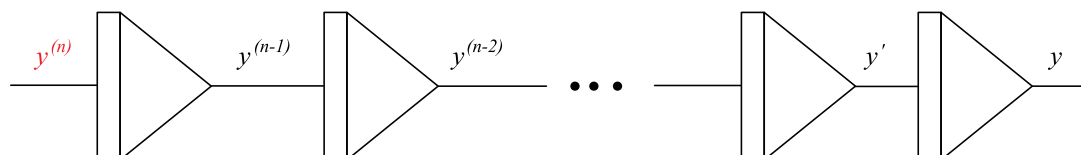
1. korak

Preuredimo diferencialno enačbo tako, da ostane na levi strani najvišji odvod, vse ostalo pa prenesemo na desno stran. Če je sistem zapisan v prostoru stanj (s sistemom diferencialnih enačb 1. reda), je zapis že ustrezen in prvi korak odpade

$$y^{(n)} = -f(y^{(n-1)}, y^{(n-2)}, \dots, y', y, u; t) \quad (6.2)$$

2. korak

Narišemo kaskado n integratorjev, če je n red najvišjega odvoda. 2. korak prikazuje slika 6.2.

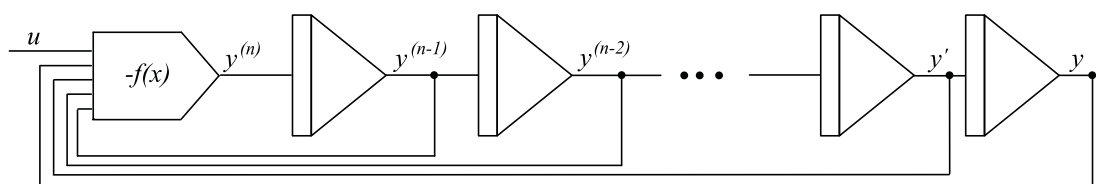


Slika 6.2: 2. korak indirektnega postopka

Predpostavimo, da je vhod v prvi integrator (najvišji odvod) znan, preostali integratorji pa generirajo nižje odvode in samo spremenljivko oz. rešitev diferencialne enačbe.

3. korak

Z upoštevanjem (virtualnih) nižjih odvodov in rešitve diferencialne enačbe generiramo negativno funkcijsko odvisnost, ki realizira desno stran enačbe (6.2). Izhod bloka, ki generira negativno funkcijsko odvisnost, je enak najvišjemu odvodu, zato ga moramo povezati na vhod prvega integratorja. Pri generaciji funkcije $-f$ uporabljamo različne bloke (razen integratorja), kar zavisi od oblike diferencialne enačbe. Dobljeni obliki, ki jo prikazuje slika 6.3, pravimo tudi kanonična oblika.



Slika 6.3: 3. korak indirektnega postopka

Postopek, ki smo ga opisali, je neposredno uporaben, če diferencialna enačba ne vsebuje odvodov vhodnega signala. Če pa le-ti nastopajo, je v primeru linearnih sistemov bolj smiselno sistem simulirati po konceptu prenosnih funkcij (podpoglavje 6.3).

Uporabnost postopka bomo prikazali na primerih.

Primer 6.1 Temperaturni proces

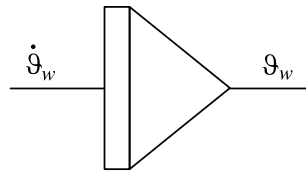
Model temperaturnega procesa je opisan v primeru 3.2. Matematični model opisuje diferencialna enačba

$$\dot{\vartheta}_w + \frac{1}{T}\vartheta_w = \frac{k}{T}p \quad (6.3)$$

Z upoštevanjem 1. koraka indirektnega postopka moramo enačbo (6.3) preurediti v obliko

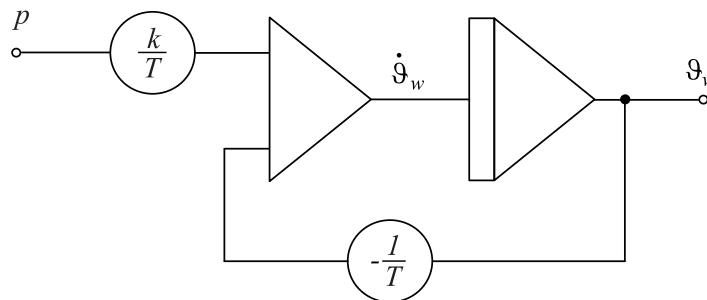
$$\dot{\vartheta}_w = -\frac{1}{T}\vartheta_w + \frac{k}{T}p \quad (6.4)$$

V drugem koraku narišemo le en integrator, ker je diferencialna enačba 1. reda. Ustrezen korak prikazuje slika 6.4.



Slika 6.4: Simulacijska shema za 2. korak

V 3. koraku generiramo desno stran enačbe (6.4). Končno simulacijsko shemo prikazuje slika 6.5.



Slika 6.5: Simulacijska shema temperaturnega procesa

Kot lahko vidimo na sliki 6.5, smo desno stran enačbe (6.4) realizirali s sumatorjem in dvema ojačevalnima blokom. Potrebna predznaka sta vključena v ojačevalnih blokih. \square

Primer 6.2 Avtomobilsko vzmetenje

Model avtomobilskega vzmetenja smo predstavili v primeru 3.1. Matematični model opisuje diferencialna enačba

$$\ddot{y}_1 + \frac{k_1 + k_2}{f} \dot{y}_1 + \frac{k_2}{M} y_1 + \frac{k_1 k_2}{M f} y_1 = 0 \quad y_1(0) = -y_{10} \quad (6.5)$$

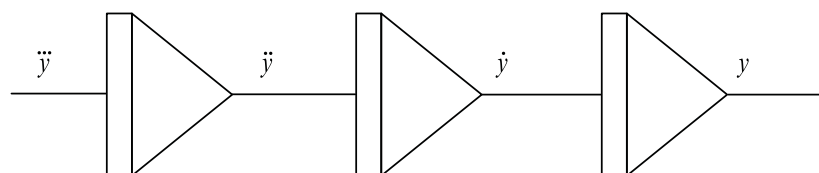
ki jo lahko poenostavimo v obliko

$$\ddot{y} + a\dot{y} + by + cy = 0 \quad y(0) = -d \quad (6.6)$$

1. korak: Preuredimo enačbo (6.6)

$$\ddot{y} = -a\dot{y} - by - cy \quad y(0) = -d \quad (6.7)$$

2. korak: Narišemo kaskado treh integratorjev, kar prikazuje slika 6.6



Slika 6.6: Simulacijska shema za 2. korak

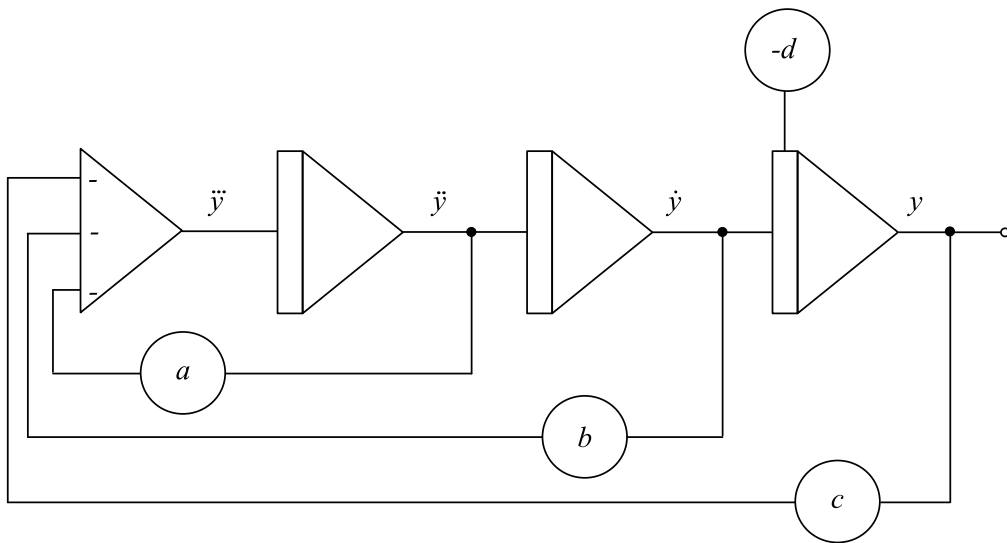
3. korak: Zaključimo simulacijsko shemo, kot prikazuje slika 6.7. V tem primeru smo nekatere predznake podali v sumacijskem bloku.

□

Primer 6.3 Ekološki sistem žrtev in roparjev

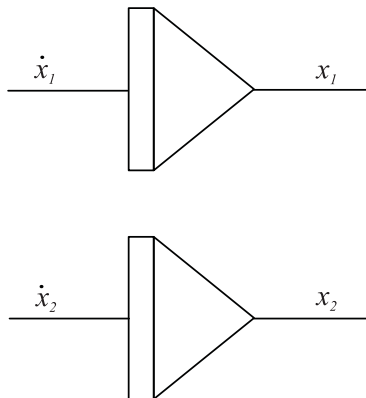
Model ekološkega sistema, v katerem nastopajo roparji in žrtve, smo predstavili v primeru 3.3. Primer prikazuje uporabnost indirektna metode pri reševanju sistema nelinearnih diferencialnih enačb. Matematični model ima obliko

$$\begin{aligned} \dot{x}_1 &= a_{11}x_1 - a_{12}x_1x_2 & x_1(0) &= x_{10} \\ \dot{x}_2 &= a_{21}x_1x_2 - a_{22}x_2 & x_2(0) &= x_{20} \end{aligned} \quad (6.8)$$



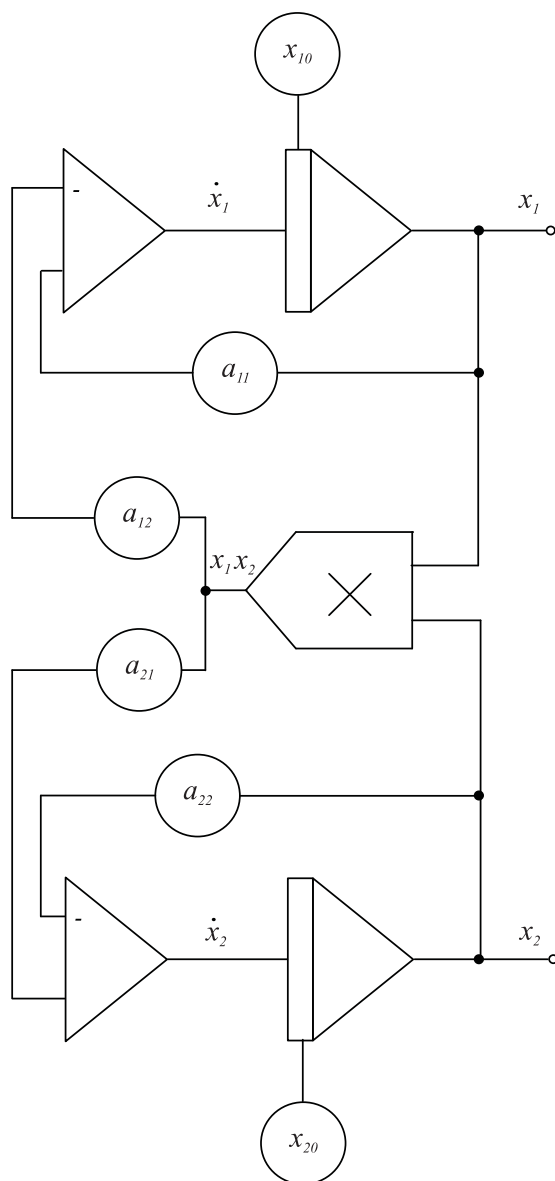
Slika 6.7: Simulacijska shema sistema avtomobilskega vzmetenja

1. korak: Ni potreben, ker imata enačbi že predpisano obliko.
2. korak: Za vsako enačbo narišemo en integrator, kar prikazuje slika 6.8.
3. korak: Zaključimo simulacijsko shemo, kot to prikazuje slika 6.9.



Slika 6.8: Simulacijska shema za 2. korak

Kot vidimo iz slik 6.8 in 6.9, je postopek simulacije nelinearnega sistema enako enostaven kot pri linearnih sistemih. Razen zank okoli posameznih integratorjev dobimo v tem primeru tudi ustrezne križne povezave. Nelinearnost vnaša v shemo množilnik. □



Slika 6.9: Simulacijska shema ekološkega sistema

6.3 Simulacija prenosnih funkcij

Prenosne funkcije imajo pomembno vlogo pri analizi in načrtovanju sistemov. Zlasti je učinkovita nazornost bločnih diagramov, v katerih nastopajo med ostalimi funkcionalnimi bloki tudi prenosne funkcije.

Od številnih metod za simulacijo prenosnih funkcij bomo prikazali eno, ki je najbolj nazorna in se največ uporablja. To je **delitvena metoda**.

Postopek bomo prikazali na primeru 6.4.

Primer 6.4 Prenosno funkcijo

$$G(s) = \frac{Y(s)}{U(s)} = \frac{as^3 + bs^2 + cs + d}{s^3 + es^2 + fs + g} \quad (6.9)$$

simuliramo z delitveno metodo s pomočjo naslednjih korakov:

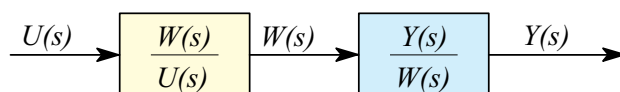
- Če koeficient pri členu z najvišjo potenco v imenovalcu ni enak 1, je potrebno vse člene polinomov v števcu in imenovalcu deliti s tem koeficientom.
- Prenosno funkcijo $\frac{Y(s)}{U(s)}$ razdelimo s pomočjo pomožne spremenljivke $W(s)$ v dve prenosni funkciji; prva predstavlja realizacijo imenovalca

$$\frac{W(s)}{U(s)} = \frac{1}{s^3 + es^2 + fs + g} \quad (6.10)$$

druga pa realizacijo števca

$$\frac{Y(s)}{W(s)} = as^3 + bs^2 + cs + d \quad (6.11)$$

Delitveni postopek nazorno prikazuje slika 6.10.



Slika 6.10: Bločna shema, ki predstavlja delitveni postopek

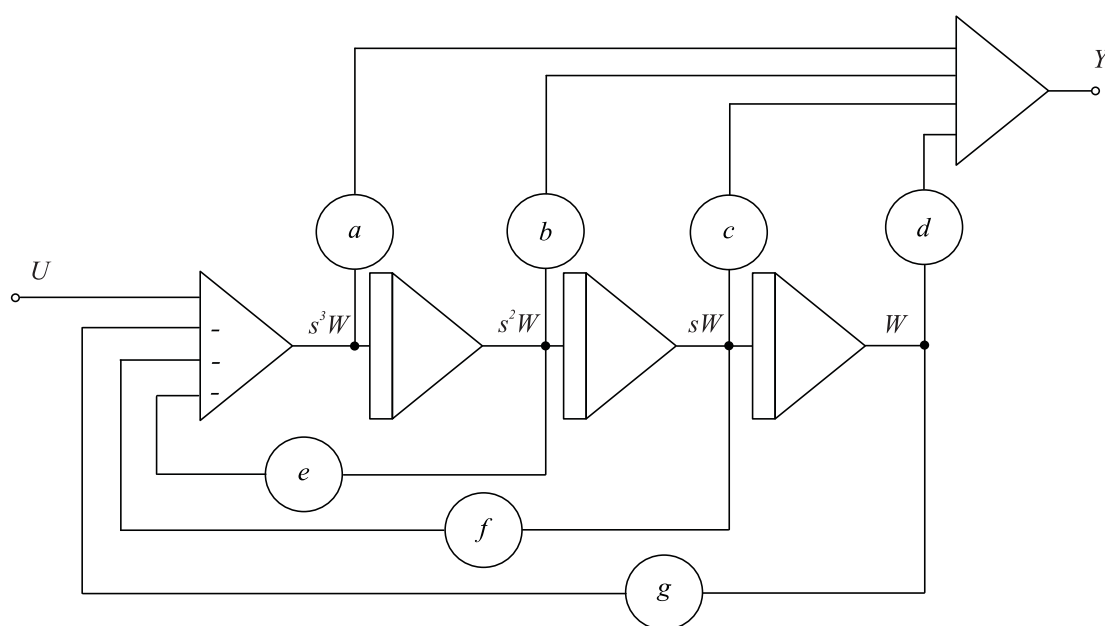
- Preuredimo enačbo (6.10) v obliko

$$s^3W = U - es^2W - fsW - gW \quad (6.12)$$

- Preuredimo enačbo (6.11) v obliko

$$Y = as^3W + bs^2W + csW + dW \quad (6.13)$$

- S pomočjo enačb (6.12) in (6.13) realiziramo simulacijsko shemo, ki jo prikazuje slika 6.11.



Slika 6.11: Simulacijska shema ob uporabi delitvene metode

Če koeficient pri členu s^3 ne bil 1, bi bile vrednosti ojačevalnih blokov v sliki 6.11 kvocienti koeficientov a , b , c , d , e , f in g s tem koeficientom. Kot vidimo iz slike 6.11, sestoji struktura iz dveh delov. Eden realizira imenovalac z indirektno metodo (le ta namreč omogoča, da so razen izhodne spremenljivke dostopni tudi odvodi), drugi del pa generira števec s pomočjo direktne metode. Metoda je zelo primerna za simulacijo več prenosnih funkcij z enakim imenovalcem, saj le enkrat simuliramo prenosno funkcijo $\frac{W}{U}$.

Shema na sliki 6.11 predstavlja realizacijo prenosne funkcije, ki je v teoriji vodenja znana kot *vodljivostna kanonična oblika* (Ogata, 2010). \square

Metoda potrebuje toliko integratorjev, kolikor je red prenosne funkcije. Uporabljamo jo tudi pri simulaciji sistemov, ki so opisani z diferencialnimi enačbami, pri čemer nastopajo tudi višji odvodi vhodnega signala (npr. $\ddot{y} + e\dot{y} + fy + gy = a\ddot{u} + b\dot{u} + c\dot{u} + du$).

6.4 Princip modularnosti

Pri simulaciji kompleksnih sistemov je treba že v fazi modeliranja upoštevati principe modularnosti, kar pomeni, da večje sklope realiziramo z manjšimi zaključenimi sklopi (podmodeli, prenosne funkcije, ...). Medtem ko je pri analitičnih metodah analize in načrtovanja potrebno take podsklope združevati oz. poenostavljati (npr. poenostavljanje bločnih diagramov s pomočjo algebre bločnih diagramov), pa je pomembna prednost pri simulaciji v tem, da vsak podsklop ločeno simuliramo po eni izmed obravnavanih metod in nato posamezne podsklope ustrezno povežemo. Na ta način so dosegljive tudi vse spremenljivke podsklopov, simulacijske sheme so pregledne in uspešno služijo tudi za dokumentacijo.

Primer 6.5 Regulacijski sistem vsebuje regulator PI

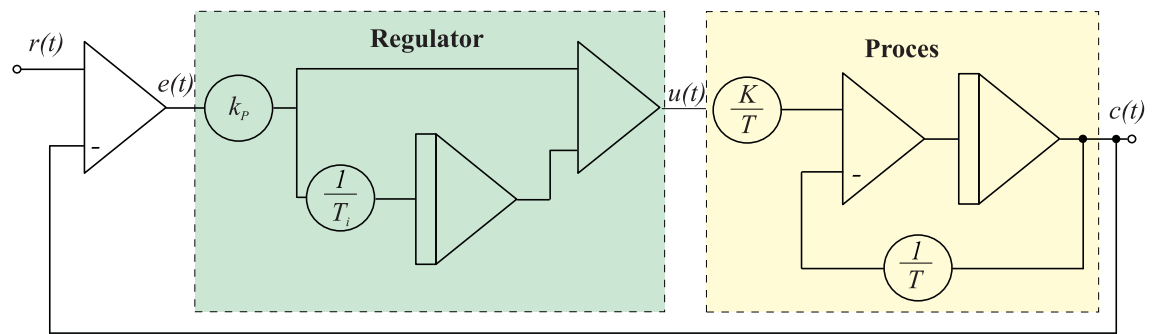
$$G_R(s) = \frac{U(s)}{E(s)} = K_P \left(1 + \frac{1}{T_I s}\right) \quad (6.14)$$

in proces 1. reda

$$G_P(s) = \frac{C(s)}{U(s)} = \frac{K}{T s + 1} \quad (6.15)$$

Ob upoštevanju nazornosti in modularnosti pri simulaciji kompleksnejših sistemov dobimo simulacijsko shemo, ki jo prikazuje slika 6.12. Slika pregledno kaže regulacijsko strukturo. Če bi npr. simulirali regulacijski sistem s pomočjo ene zaprtozančne prenosne funkcije $\frac{C(s)}{R(s)}$, ne bi bila dostopna regulirna veličina $u(t)$.

□



Slika 6.12: Simulacijska shema regulacijskega sistema

7.

Orodja za simulacijo dinamičnih sistemov

Metode za simulacijo zveznih dinamičnih sistemov lahko učinkovito uporabljamo le v povezavi z modernimi računalniškimi orodji.

7.1 Osnovne lastnosti simulacijskih sistemov

Moderna simulacijska orodja omogočajo simulacijo dinamičnih modelov na tak način, da se uporabnik lahko osredotoči na samo problematiko (modeliranje, simulacija), ne pa na programiranje problema. Posamezne vrste orodij se z uporabniškega vidika med seboj razlikujejo glede na:

- potreben čas, da postane uporabnik večč simulacijskega orodja,
- uporabniško prijaznost,
- potreben čas za razvoj simulacijskega modela,
- možnosti pri spreminjanju simulacijskega modela,
- sposobnosti pri izvajanju simulacije (hitrost, interaktivnost, numerična robustnost, možnost simulacije v realnem času, ...),
- možnosti dokumentiranja modelov, rezultatov simulacije in dr.

V nadaljevanju si bomo ogledali simulacijsko okolje **Simulink**, ki deluje znotraj programskega okolja Matlab (Simulink, 2009, Oblak, Škrjanc, 2008, Dabney, Harman, 2004). Simulink je v osnovi splošno namenski bločno orientirani simulator. Model podamo s pomočjo grafično zasnovanega uporabniškega vmesnika.

7.2 Simulacijsko okolje Matlab-Simulink

Simulink je orodje, ki ga lahko uporabljamo v okviru programskega paketa Matlab in omogoča simulacijo dinamičnih sistemov. Nastal je kot dopolnilo Matlab-a, saj je za uporabnika zelo priročno, če lahko v enem programskem okolju izvaja različne operacije in pri tem ni potreben prenos podatkov iz oz. v druge programe (filtriranje signalov, opazovanje meritev in primerjava meritev z odzivi modela, načrtovanje vodenja, simulacijsko opazovanje delovanja načrtanih regulacijskih sistemov, itd.) in tako ponuja določene prednosti pri simulaciji, hkrati pa ohranja funkcionalnost Matlaba.

7.2.1 Osnovna uporaba Simulinka

Opišimo uporabo osnovnih funkcij orodja Simulink na primeru reševanja modela drugega reda naslednje oblike:

$$\ddot{y}(t) + 3\dot{y}(t) + 2y(t) = 2u(t) \quad (7.1)$$

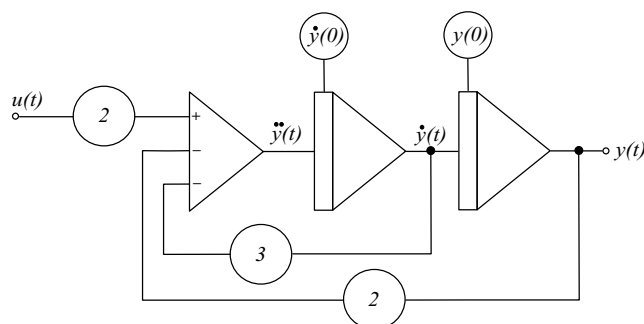
Enačbo (7.1) preoblikujmo za realizacijo simulacijske sheme po indirektni metodi:

$$\ddot{y}(t) = -3\dot{y}(t) - 2y(t) + 2u(t) \quad (7.2)$$

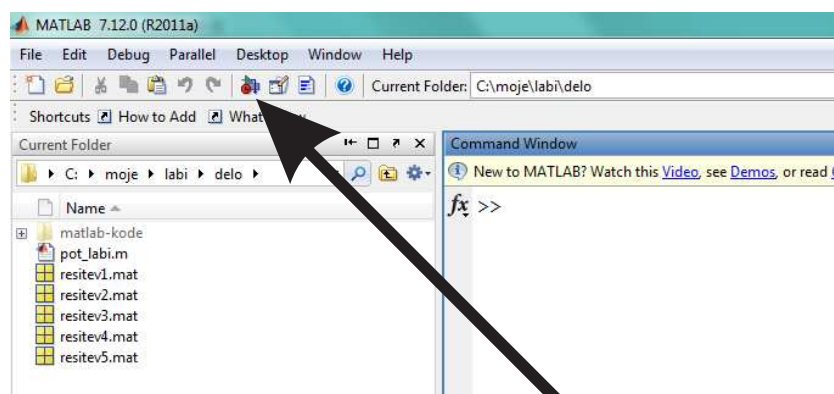
in narišimo pripadajočo splošno simulacijsko shemo, kot jo prikazuje slika 7.1.

Sedaj se lotimo izvedbe simulacijske sheme v Simulinku. Orodje zaženemo tako, da v komandnem oknu odtipkamo ukaz **simulink** ali pa kliknemo na ikono v orodni vrstici, kot prikazuje slika 7.2.

Ta ukaz odpre brskalnik Simulinkove knjižnice, kot prikazuje slika 7.3. Najprej s klikom na ikono iz orodne vrstice, kot kaže puščica na sliki 7.3, odpremo novo okno (nov model). Ko ta model shranimo, dobimo datoteko *.mdl.



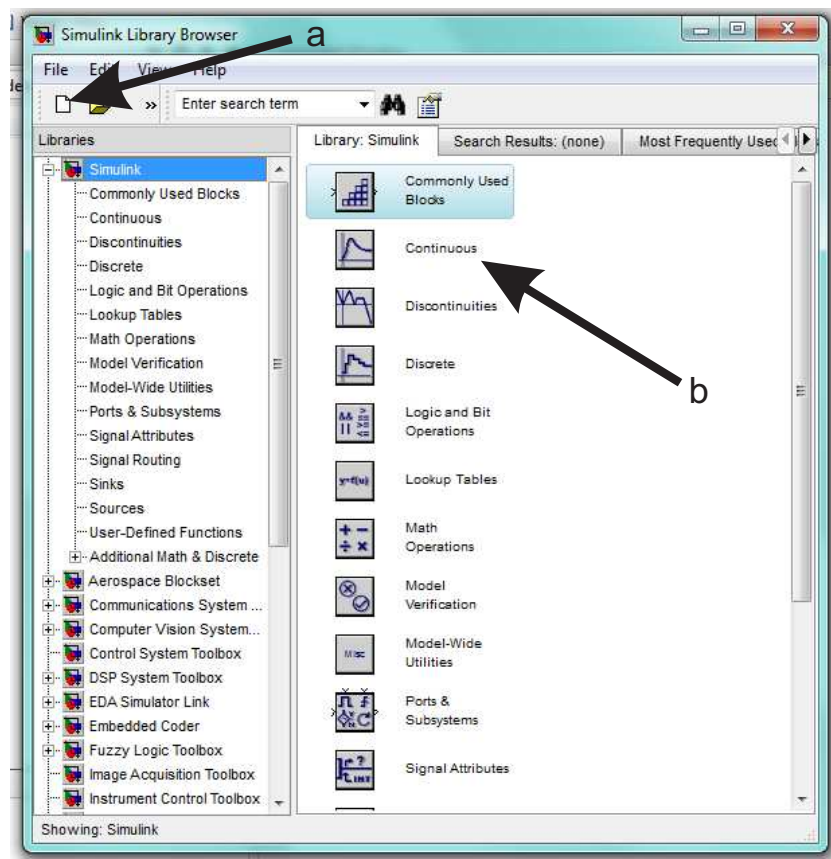
Slika 7.1: Simulacijska shema sistema 2. reda



Slika 7.2: Zagon Simulinka iz orodne vrstice

Simulinkova knjižnica vsebuje veliko število blokov, ki jih lahko uporabljamo za gradnjo simulacijskih modelov. Zaradi preglednosti so urejeni po skupinah (podknjižnicah). Oglejmo si najprej skupino zveznih modelov. To odpremo z dvojnim klikom na oznako **Continuous**, kot kaže puščica b na sliki 7.3. Pri tem se odpre okno, kot je prikazano na levi strani slike 7.4.

Med bloki najdemo tudi takega, z oznako integrator, ki ga prenesemo v odprti simulacijski model. To storimo z ukazoma **ctrl/c** in **ctrl/v**, ali pa tako, da kliknemo na blok, držimo levi gumb na miški in blok vlečemo na ustrezno mesto v okno modela. Ker potrebujemo dva integratorja, bi lahko postopek ponovili. Lahko pa naredimo kopijo bloka kar znotraj modela z ukazoma **ctrl/c** in **ctrl/v**, ali pa tako, da kliknemo integrator z desnim gumbom in vlečemo na mesto, kjer želimo imeti kopijo.

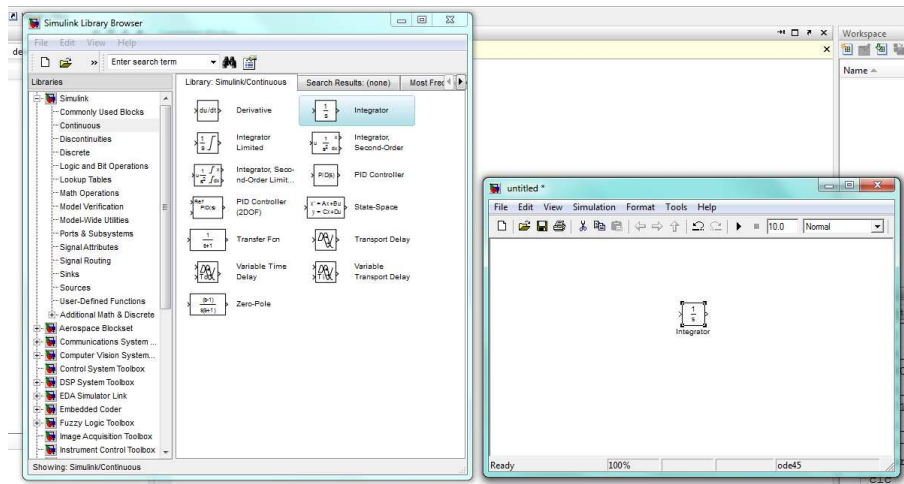


Slika 7.3: Brskalnik Simulinkove knjižnice

Blokom, ki smo jih prenesli v model, nastavljamo lastnosti oz. parametre. V ta namen dvakrat kliknemo na prvi integrator in opazujemo vsebino okna, ki se odpre (slika 7.5).

Vidimo, da je pri integratorju mogoče nastaviti zunanji reset, izvor začetnega stanja, (v našem primeru izberimo notranji) in vrednost začetnega stanja (Initial condition). Predlagana začetna vrednost je nič in v našem primeru na obeh integratorjih ohranimo to vrednost. Lahko tudi omejimo izhod na določeno želeno območje in nastavimo t.i. absolutno toleranco ter omogočimo detekcijo prehoda ničle. Slednji dve lastnosti nastavljamo le pri zahtevnejših simulacijah. V našem primeru pustimo vse privzete vrednosti in to potrdimo s klikom na gumb OK v spodnjem delu okna.

Sedaj potrebujemo v simulacijski shemi še sumator, ki ga najdemo v skupini



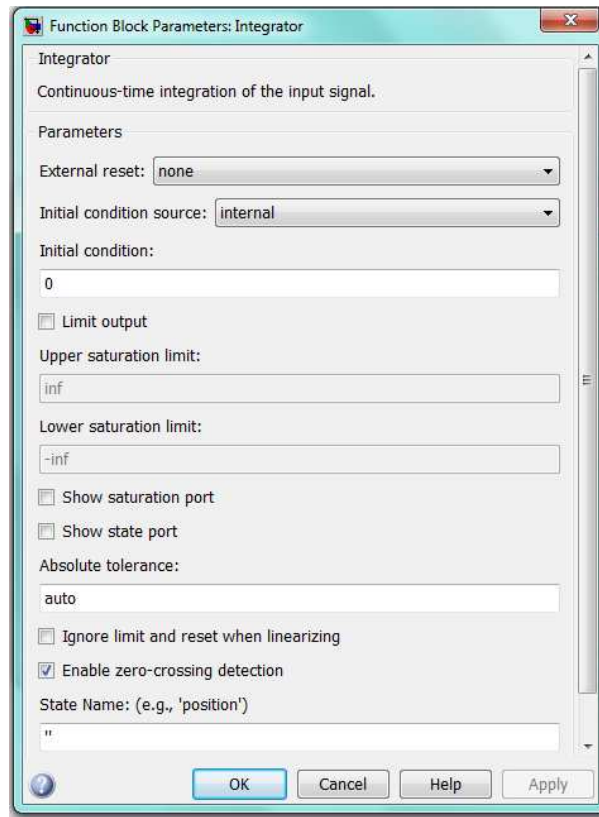
Slika 7.4: Skupina zveznih modelov (Continuous) Simulinkove knjižnice

blokov z oznako **Math Operations**. Na enak način kot integrator, sedaj v simulacijsko shemo prenesemo blok z oznako **Add**, to je sumator. V tej skupini blokov se nahaja še ojačevalni blok **Gain**. Ko prenesemo v model tudi ojačevalni blok, dobimo prikaz na zaslonu na sliki 7.6).

Kliknimo dvakrat na sumator, da se odpre pripadajoče komunikacijsko okno, kot je prikazano na sliki 7.7 in v vrstico, ki nosi oznako *List of signs*, natipkajmo en plus (+) in dva minusa (-) ter potrdimo s pritiskom na gumb OK. To povzroči, da se na ikoni sumatorja v naši datoteki pojavijo trije vhodi, ki so ustrezno predznačeni.

Glede na razmere na sliki 7.1 bomo potrebovali tri ojačevalne bloke. Zato ojačevalni blok, ki ga že imamo v shemi, še dvakrat skopirajmo. S klikom na vsakega od ojačevalnih blokov odprimo pripadajoče komunikacijsko okno ter v vrstici z oznako *Gain* nastavimo pri dveh ojačevalnikih vrednost 2, pri enem pa 3.

Zaradi preglednosti včasih priročno, da katerega od elementov v simulacijski shemi zasukamo ali spremenimo njegovo velikost. Element lahko zavrtimo za 180 stopinj tako, da ga najprej izberemo (kliknemo nanj), nato pa v orodni vrstici simulacijske sheme kliknemo na gumb z oznako *Format* in izberemo možnost *Flip block*. Blok lahko povečamo ali pomanjšamo, oz. preoblikujemo tako, da najprej nanj kliknemo (ga izberemo). Pri tem se na robovih prikažejo majhni črni



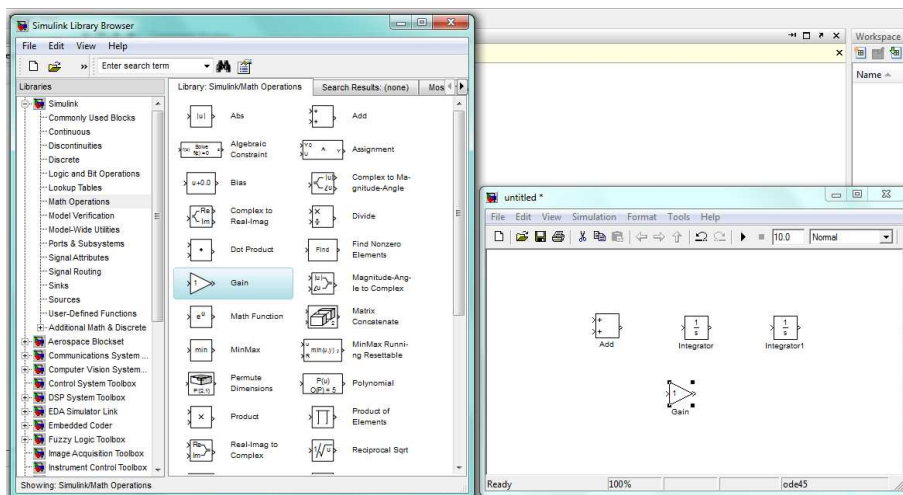
Slika 7.5: Komunikacijsko okno integratorja

kvadratki. Če postavimo kurzor na takšen črn kvadratik in držimo levi gumb na miški, bomo oblikovali velikost ikone tega bloka. To seveda vpliva le na prikaz bloka, ostale lastnosti pa ostajajo nespremenjene.

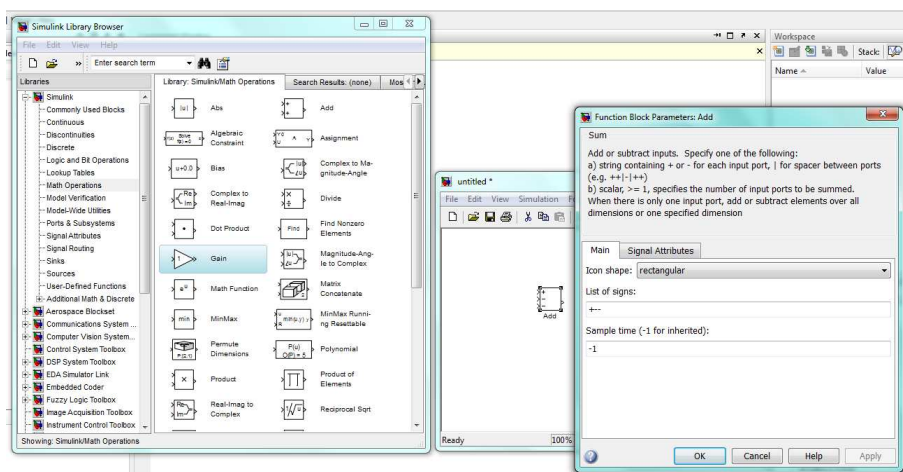
Povezovanje blokov je zelo preprosto. S kurzorjem se postavimo na izhod izbranega bloka in pritisnemo levi gumb na miški. Gumb držimo in pomikamo kurzor do vhoda bloka, ki ga želimo povezati.

Slika 7.8 prikazuje simulacijsko shemo v Simulinku za model, ki ga podaja enačba 7.1.

Opazimo, da je Simulinkov simulacijski model zelo podoben splošni simulacijski shemi na sliki 7.1.

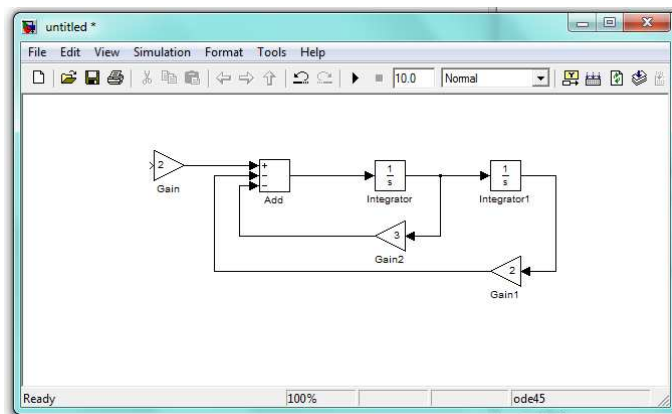


Slika 7.6: Skupina blokov z oznako Matematične operacije (**Math Operations**) (levo okno) in gradnja lastnega simulacijskega modela (desno okno)

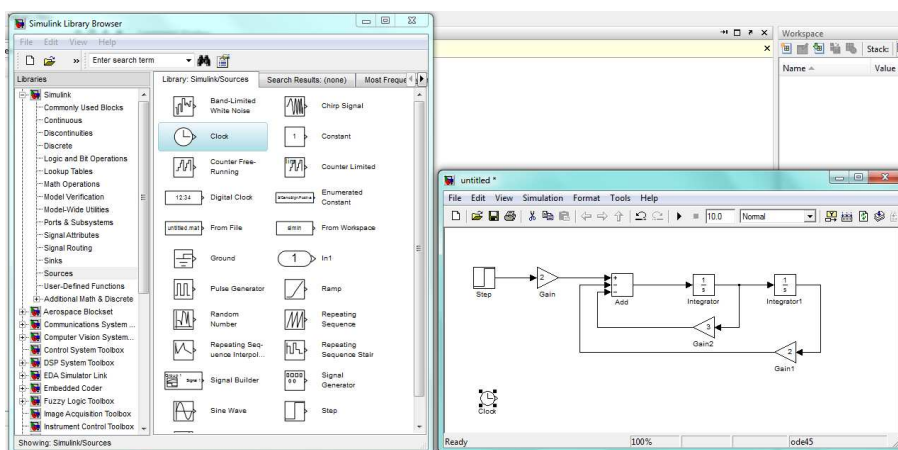


Slika 7.7: Komunikacijsko okno sumatorja (desno okno na sliki)

Predno bomo lahko pričeli s simulacijo, pa moramo narediti še tri pomembne korake. Najprej moramo dodati vzbujalni signal. Predpostavimo, da želimo uporabiti enotino stopnico. V ta namen odprimo skupino z oznako **Sources**, kjer najdemo številne bloke, s pomočjo katerih lahko generiramo potrebne signale (glej levo okno na sliki 7.9). Iz te skupine prenesemo blok z oznako **Step in Clock**. Stopničasti signal (blok **Step**) povežemo z vhodom v model (glej desno okno na sliki 7.9)



Slika 7.8: Simulink shema za sistem 2. reda

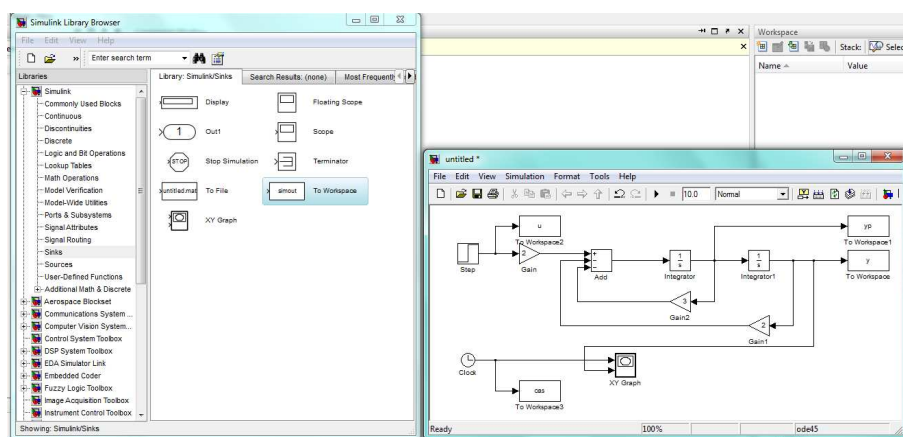


Slika 7.9: Vzbujaalni signali (levo okno) in dopolnjen simulacijski model (desno okno)

Z dvojnimi klikom na blok **Step** odpremo pripadajoče komunikacijsko okno, kjer nastavimo parametre tega bloka na naslednje vrednosti: čas nastopa stopničastega signala (*Step time*) je 0, začetna vrednost (*Initial value*) je 0 in končna vrednost (*Final value*) je 1. Parameter *Sample time* oz. čas vzorčenja pustimo na vrednosti 0, saj imamo opravka z zveznim modelom.

Rezultate simulacije lahko opazujemo na dva osnovna načina in sicer sprotno, med samo simulacijo in po končanem simulacijskem teku. V ta namen uporabljamo bloke za iz skupine **Sinks** (levo okno na sliki 7.10). V model prenesemo bloka

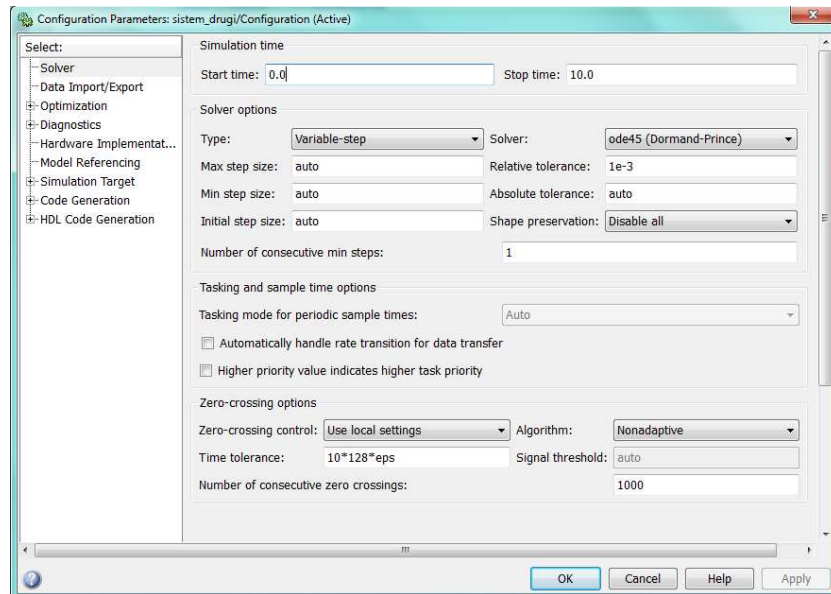
z oznako **XY Graph** in **To Workspace**. Blok **XY Graph** služi za sproten prikaz signalov. Na prvi vhod tega bloka priključimo izhod bloka **Clock**, to je bloka, ki poskrbi za možnost uporabe vektorja časa, ki ga program generira med simulacijskim tekom. Na drugi vhod pa priključimo signal, ki bi ga želeli opazovati. V našem primeru se odločimo za opazovanje izhodnega signala $y(t)$ (desno okno na sliki 7.10). Z dvojnimi klikom na ta blok ugotovimo, da je mogoče izbrati območje opazovanja signala po x in y osi.



Slika 7.10: Bloki za opazovanje (levo okno) in dopolnjen simulacijski model (desno okno)

Zelo uporaben je tudi blok **To Workspace**, s pomočjo katerega lahko poskrbimo, da se bodo po končanem simulacijskem teku v delovnem prostoru Matlab nahajali signali, ki bi jih želeli uporabiti. Če odpremo komunikacijsko okno tega bloka, ugotovimo, da lahko izberemo ime spremenljivke in tudi obliko oz. *format* podatkovne strukture. Običajno uporabimo format na *Array*, kar pomeni, da bomo po končanem simulacijskem teku imeli v delovnem prostoru celoten vektor določenega signala z izbranim imenom in ga bomo lahko direktno uporabili v okolju Matlab. V našem primeru smo se odločili za uporabo štirih tovrstnih blokov, ki bodo poskrbeli, da se bo po končanem simulacijskem teku v delovnem prostoru nahajala neodvisna spremenljivka čas pod imenom *cas*, vhodni signal *u*, izhod *y* in odvod izhodnega signala pod imenom *yp* (glej desno okno na sliki 7.10).

Sedaj pa moramo ustrezno nastaviti samo še parametre simulacije. To najlažje storimo tako, da kliknemo v orodni vrstici simulacijske sheme na gumb *Simulation* in izberemo možnost *Configuration Parameters*. Pri tem se odpre okno uporabniškega vmesnika, kot je prikazano na sliki 7.11.



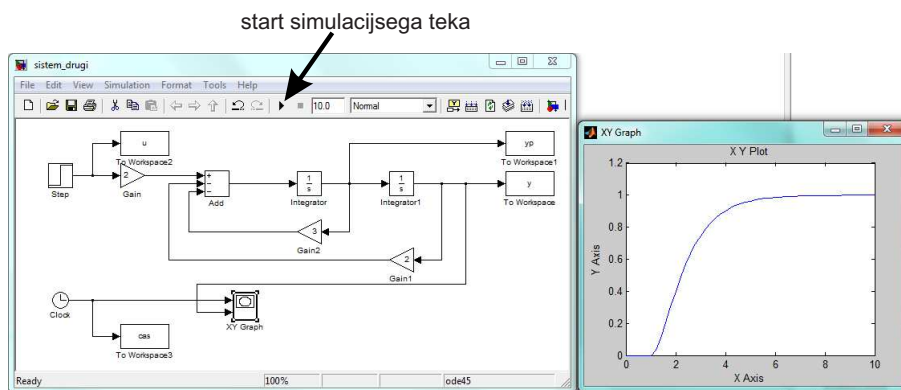
Slika 7.11: Okno uporabniškega vmesnika za nastavitve parametrov simulacije

Najprej se odločimo za primerno dolžino simulacijskega teka. Ker je daljša od obeh časovnih konstant enaka 1 sekundo, naj simulacija traja desetkrat tako (nastavitve v okencu *Stop time*). Nastavitve ostalih parametrov običajno pri enostavnejših simulacijah ni potrebno spreminjati. Omenimo pa, da v tem oknu lahko izbiramo med različnimi numeričnimi (integracijskimi) algoritmi, nastavljammo zahteve za natančnost izračunov in podobno.

Na koncu shranimo model pod zelenim imenom in sprožimo izvajanje simulacijskega teka. To lahko naredimo na več načinov. Najpreprostejša možnost je ta, da kliknemo na črno puščico v orodni vrstici simulacijske sheme (glej levo okno na sliki 7.12). Blok za sprotni prikaz (XY Graph) poskrbi, da se ob sproženju simulacijskega teka odpre okence, kamor se med simulacijskim tekom izrisuje izbrani signal (desno okno na sliki 7.12).

Po končanem simulacijskem teku se v delovnem prostoru nahajajo spremenljivke cas , u , y in yp , ki jih lahko uporabimo za grafični prikaz simulacijskih rezultatov.

Oglejmo si še enkrat simulacijsko shemo, ki smo jo pravkar zgradili v Simulinku. Parametre modela smo v tej shemi določili kar z ustreznimi vrednostmi, ki smo jih vnesli preko komunikacijskih oken uporabljenih blokov. Namesto konkret-



Slika 7.12: Sprotni grafični prikaz rezultatov (desno okno), ki ga je povzročil blok **XY Graph** v simulacijski shemi (levo okno)

nih vrednosti pa lahko uporabimo tudi imena spremenljivk ali celo matematične izraze. Pri tem je pomembno, da v delovnem prostoru Matlaba priredimo uporabljenim spremenljivkam želene vrednosti, preden zaženemo simulacijo. V nasprotnem primeru izračun seveda ne bo mogoč in Simulink bo javil napako.

Da bi ilustrirali nakazano možnost, preuredimo simulacijsko datoteko, kot je prikazano na sliki 7.13 in jo shranimo pod imenom `sistemdrugi.mdl`. Parametre smo preimenovali v spremenljivke $k1$, $k2$ in $k3$, v delovni prostor pa smo napeljali tudi drugi odvod $\ddot{y}(t)$. Poleg tega smo označili dolžino simulacijskega teka s $Tmax$.

Namesto da v komandnem oknu Matlaba odtipkamo več komand, lahko komande vpišemo v Matlabovo tekstovno datoteko `*.m` -t.j. program v jeziku Matlab.

```
% Simulacija sistema 2. reda
%
% Nastavitev parametrov modela:
k1=2;
k2=3;
k3=2;
%
% Dolžina simulacijskega teka:
Tmax=10;
%
%Proženje simulacijskega teka:
```

Slika 7.13: Simulacijska shema sistemdrugi.mdl

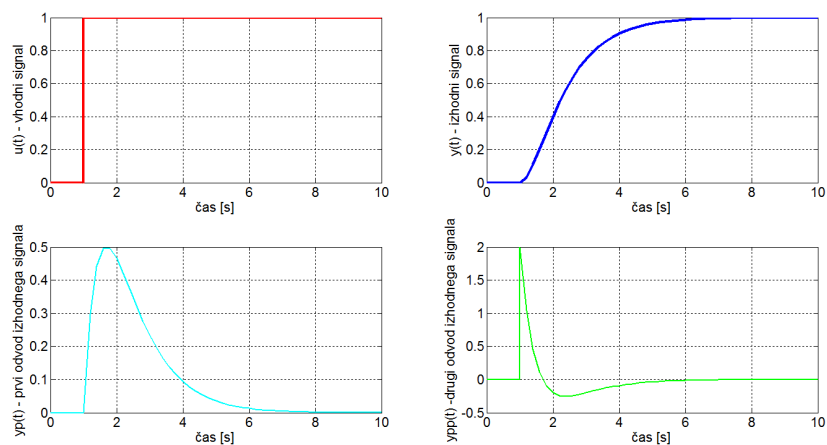
```
sim('sistemdrugi.mdl',Tmax);
%
subplot(2,2,1)
plot(cas,u,'r','Linewidth',3);
xlabel('čas [s]','FontSize',16)
ylabel('u(t) - vhodni signal','FontSize',16)
grid
subplot(2,2,2)
plot(cas,y,'b','Linewidth',3);
xlabel('čas [s]','FontSize',16)
ylabel('y(t) - izhodni signal','FontSize',16)
grid
subplot(2,2,3)
plot(cas,yp,'c','Linewidth',3);
xlabel('čas [s]','FontSize',16)
ylabel('yp(t) - prvi odvod izhodnega signala','FontSize',16)
grid
subplot(2,2,4)
plot(cas,ypp,'g','Linewidth',3);
xlabel('čas [s]','FontSize',16)
ylabel('ypp(t) -drugi odvod izhodnega signala','FontSize',16)
grid
```

Spomnimo se, da je desno od oznake % v vrstici komentar, ki smo ga dodali zaradi preglednosti. Opozorimo posebej samo na vrstico z naslednjo vsebino:

```
sim('sistemdrugi.mdl',Tmax);
```

Uporaba funkcije **sim** omogoča proženje simulacije iz okolja Matlab. Pri tem smo kot vhodni parameter najprej navedli ime simulacijskega modela, nato pa še spremenljivko, ki definira dolžino simulacijskega teka.

Če sedaj v komandnem oknu natipkamo ime komandne datoteke oz. Matlabovega programa, se izvrši simulacijski tek in narišejo prikaz rezultati simulacije, kot prikazuje slika 7.14.



Slika 7.14: Rezultati simulacijskega izračuna

7.2.2 Analiza modelov v Simulinku

Analiziranje modelov zgrajenih v Simulinku je možno na več različnih načinov: z uporabo orodja **Linear Analysis**, če v orodni vrstici izberemo (**Tools Control design Linear Analysis**) ali iz Matlab-a z uporabo funkcij **linmod** in **trim**.

Linear Analysis

Orodje omogoča linearno analizo dela modela, ki ga označimo z vhodno in izhodno točko (določimo z desnim klikom ustreznega signala in izberemo Linearization points/Input ali Output Point). Linearni model med vhodno in izhodno točko dobimo v obliki prenosne funkcije ali prostora stanj. Nadalje lahko izberemo nekaj osnovnih eksperimentov v časovnem in frekvenčnem prostoru s pomočjo lineariziranega modela: odziv na stopnico, impulzni odziv, Bodejev diagram, Nyquistov diagram, izris polov in ničel v s-ravnini,

Linmod

Linmod je funkcija s pomočjo katere lahko lineariziramo splošni nelinearni sistem diferencialnih enačb okrog delovne točke. Rezultat linearizacije je zapis lineariziranega modela v prostoru stanj.

Če lineariziramo Simulink model, potem moramo definirati vhode in izhode iz dela, ki ga želimo linearizirati, s pomočjo blokov **In1** (v skupini Sources) in **Out1** (v skupini Sinks).

Trim

Funkcija `trim` omogoča izračun ustaljenega stanja obravnavanega modela. Ustaljeno stanje lahko izračunamo pri določeni vrednosti vhoda, lahko pa izračunavamo vhodni signal in stanja pri znanem izhodu v ustaljenem stanju. Vgrajen je optimizacijski postopek, ki minimizira vrednosti odvodov.

7.2.3 Poglobljena uporaba Simulinka

V tem delu bo natančneje predstavljeno delovanje Simulinka ter zgradba S-funkcij.

Delovanje Simulinka

Modeli v Simulinku so predvsem enostavni za gradnjo, s stališča učinkovitost izvajanja simulacije pa niso optimalni. Prvi korak procesiranja modela je generiranje podatkovne strukture, ki je optimalna za simulacijo. Vključijo se ustrezni bloki, ki se tudi razvrstijo. Simulacija poteka s pomočjo numerične integracije diferencialnih enačb. Simulacijo lahko zaganjamo iz Simulink okna ali pa iz Matlabovega ukaznega okna z uporabo ukaza `sim`.

S-funkcije

Ko zgradimo Simulink model in ga shranimo, sistem zgradi t.i. S-funkcijo, ki je dostopna iz Matlaba. V tej funkciji je definirana dinamika modela. Tako definiran model lahko simuliramo (integriramo), lineariziramo in iščemo ustaljena stanja (ravnotežne točke) modela. S-funkcija deluje enako kot katerakoli Matlab funkcija, ima pa določeno sintakso:

```
sys=proces(t,x,u,flag)
```

kjer je proces ime modela in parameter flag določa informacijo, ki jo funkcija vrne v izhodno spremenljivko sys. Na primer ob vrednosti flag=1 funkcija vrne odvode stanj modela pri vektorju stanj x in vhodni spremenljivki u. S-funkcijo lahko napišemo tudi v obliki navadne M-datoteke ali MEX-datoteke, ki je C-jevska ali Fortranska funkcija. Tako lahko ločimo tri različne načine pisanja S-funkcij:

- grafični GUI Simulink,
- M-datoteka Matlab,
- MEX-datoteka C ali Fortran funkcija.

Način podajanja S-funkcij je odvisen od zahtevnosti podajanja, uporabe S-funkcije in hitrosti, ki je zahtevana za izvajanje.

Uporabniško definirano S-funkcijo lahko uvedemo v Simulink shemo tako, da jo pridružimo posebnemu bloku (S-function block). Z maskiranjem lahko takemu bloku dodamo še običajni vmesnik, ki ga imajo ostali Simulink bloki. Na ta način je uporabniku omogočeno lastno generiranje dinamičnih modelov.

Kako delujejo S-funkcije? Funkcije so zgrajene tako, da omogočajo izgradnjo modelov za reševanje časovno zveznih, diskretni in hibridnih problemov. Zato ima S-funkcija točno določeno strukturo. Ključni element S-funkcije je parameter **flag**, ki definira informacijo, ki jo daje funkcija ob klicu na izhodu:

- flag=0 S-funkcija vrne velikost parametrov in začetnih pogojev,
- flag=1 S-funkcija vrne odvode stanj (dx/dt),
- flag=2 S-funkcija vrne predikcijo diskretnih stanj ($x(k+1)$),
- flag=3 S-funkcija vrne izhodno spremenljivko,
- flag=4 S-funkcija vrne čas naslednjega časovnega trenutka.

Vsaka situacija posreduje del informacije, ki je potreben za izvajanje simulacije.

V primeru flag=0 kličemo funkcijo takole:

```
>>flag=0  
>>[sizes,x0]=sfun([],[],[],flag)
```

Vektor x0 je vektor začetnih stanj, v vektorju **sizes** pa je naslednja informacija:

- sizes(1) število zveznih stanj,
- sizes(2) število diskretnih stanj,
- sizes(3) število izhodnih spremenljivk,
- sizes(4) število vhodnih spremenljivk,
- sizes(5) število nezveznosti,
- sizes(6) število algebrajskih zank.

V primeru zveznega sistema sta za simulacijo potrebni le informaciji o odvodih stanj v vsakem trenutku simulacije (flag=1) in izhodi sistema (flag=3). Opozoriti

velja, da se za razliko od klica funkcije z zastavico `flag=0`, ostali klici pojavljajo v vsakem računskem koraku simulacije.

Pretvorba S-funkcije v Simulink blok. Katerokoli obliko S-funkcije lahko pretvorimo v Simulink blok. V ta namen je potrebno v funkcijski blok **S-function block** oz. v njen uporabniški vmesnik vpisati ime funkcije in parametre. Blok namreč omogoča prenos določenih parametrov, ki sledijo formalni strukturi parametrov `t`, `x`, `u` in `flag`. Medsebojno jih ločimo z vejico. Na koncu postopka običajno tak blok tudi maskiramo.

Izgradnja S-funkcije v obliki M-datoteke je najlažja, če si pomagamo s predlogo **sfuntmpl**, ki se nahaja na poddirektoriju toolbox \simulink systemskega Matlabovega direktorija.

7.3 Primeri v okolju Matlab - Simulink

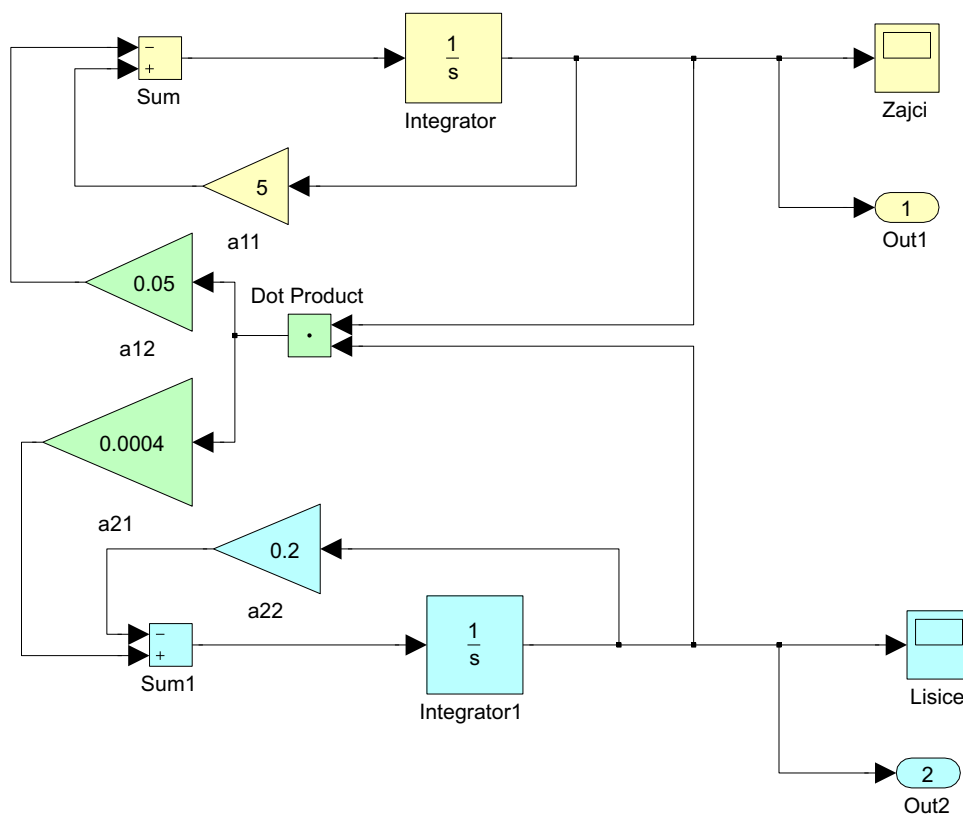
Uporabnost bločno orientiranih simulacijskih jezikov bomo prikazali na razvoju treh simulacijskih programov -model ekološkega sistema roparjev in žrtev, regulacijski sistem ogrevanja prostora in model avtomobilskega vzmetenja bomo simulirali v okolju Matlab-Simulink.

Primer 7.1 Ekološki sistem roparjev in žrtev

Primer 6.3 obravnava razvoj simulacijske sheme modela ekološkega sistema roparjev in žrtev, ki je predhodno opisan v Primeru 3.3. Model želimo simulirati pri konstantah $a_{11} = 5$, $a_{12} = 0.05$, $a_{21} = 0.0004$, $a_{22} = 0.2$ in pri začetnem številu zajcev in lisic $x_{10} = 520$, $x_{20} = 85$. V pomoč nam bo simulacijska shema, ki jo prikazuje slika 6.9 in enačbe (6.8). Ker okolje Matlab-Simulink zahteva vnos modela na grafični, bločno orientirani način, prenesemo na zaslon vse potrebne bloke, ki jih vsebuje slika 6.9. Potrebujemo dva integratorja (na zaslon jih povlečemo iz knjižničnega okna Continuous, privzeti imeni sta **Integrator** in **Integrator1**), štiri ojačevalne bloke (privzeta imena so **Gain**, **Gain1**, **Gain2** in **Gain3**, a smo ta imena v shemi nadomestili z imeni konstant **a11**, **a12**, **a21** in **a22**), dva sumatorja (privzeti imeni **Sum** in **Sum1**) in en množilnik (**DotProduct**). Vse te bloke prenesemo iz knjižničnega okna Math Operations. Na koncu iz knjižničnega okna Sinks dodamo dva prikazovalna bloka (privzeti imeni **Scope** in **Scope1** smo zamenjali z besedama **Zajci** in **Lisice**). Bloke razporedimo tako, kot kaže slika 6.9 in jih nato ustrezno povežemo (z miško potegnemo od izhoda bloka proti ustreznemu

vhodu bloka). Ko je shema povezana, nastavimo parametre blokov z dvojnimi klikom, ki odpre uporabniški vmesnik ustreznega bloka. Ojačevalnim blokom podamo vrednost ojačenja, integratorjema pa začetna pogoja. Ker je problem elementaren, ni potrebno nastaviti nobenih simulacijskih parametrov razen trajanja simulacijskega teka - vrednost 10 nastavimo v posebnem okencu v opravilni vrstici desno zgoraj. V kakšnih enotah je neodvisna spremenljivka, lahko ve le uporabnik, ki je problem tudi modeliral. Spomnimo se, da je enota neodvisne spremenljivke eno leto, torej traja simulacija deset let.

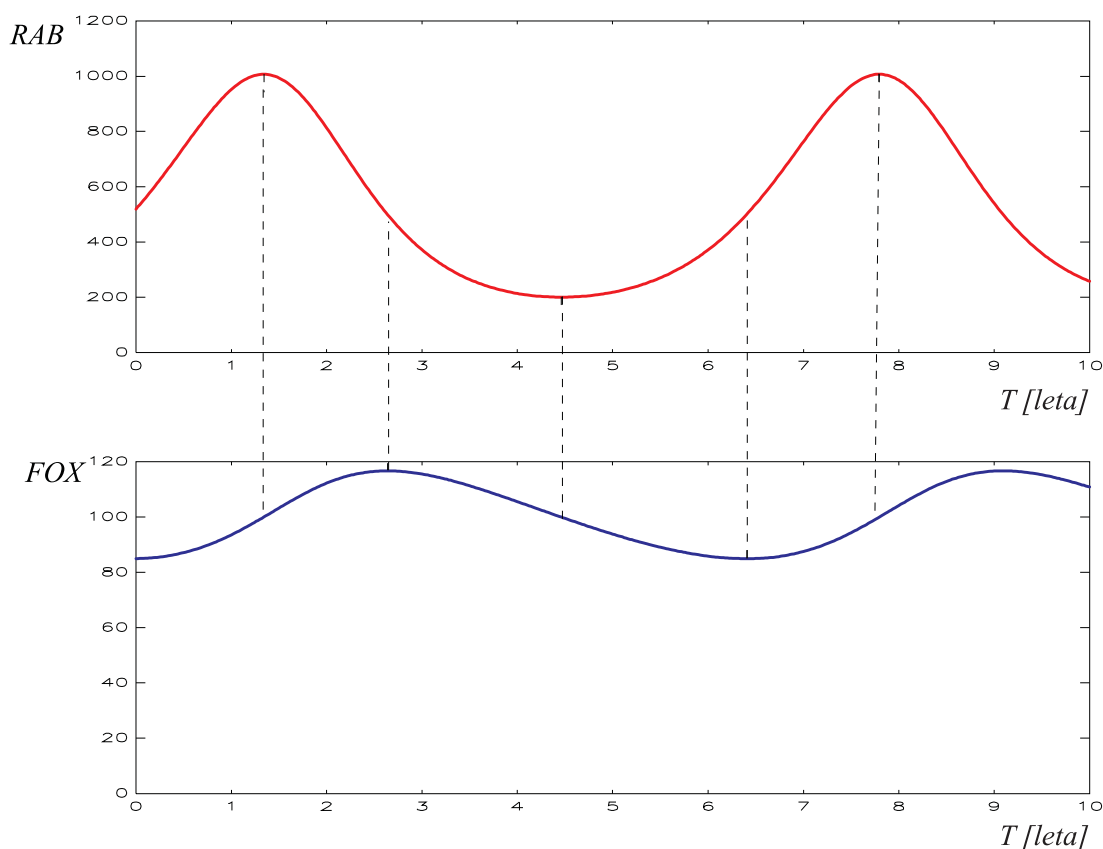
Simulink shemo prikazuje slika 7.15.



Slika 7.15: Simulacijska shema v Simulinku za ekološki primer zajcev in lisic

Rezultate simulacije prikazuje slika 7.16.

Opazimo, da je ekološka perioda približno sedem let, da število lisic najhitreje narašča ob največjem številu zajcev in najhitreje upada ob najmanjšem številu zajcev.



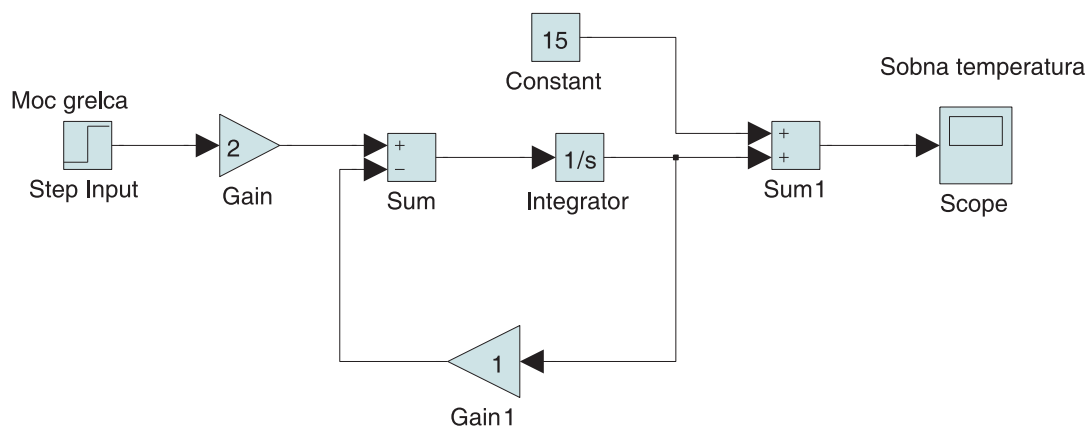
Slika 7.16: Časovna poteka števila zajcev in lisic

□

Primer 7.2 Regulacijski sistem ogrevanja prostora

Primer 6.1 obravnava razvoj simulacijske sheme temperaturnega procesa, ki je predhodno opisan v Primeru 3.2. Simulacijsko shemo razvijemo s pomočjo slike 6.5. Potrebujemo sumator, integrator in dva ojačevalna bloka. Zaradi veljavnosti enačbe (3.19) smo dodali še en sumator, s pomočjo katerega smo iz temperature v delovni točki in temperature okolice izračunali absolutno temperaturo. Simulink shemo prikazuje slika 7.17.

Simulacijski model procesa nadgradimo v shemo za regulacijo po sliki 3.21. Oporabimo blok `Signal builder` iz knjižnice `Sources` za generacijo spremenljivega referenčnega signala, ki ga opisuje tabela 7.1. Z blokom `Signal builder` opišemo referenčni signal s pomočjo grafičnega uporabniškega vmesnika. Iz knjižnice



Slika 7.17: Simulacijska shema v Simulinku za model ogrevanja prostora

Tabela 7.1: Želena sobna temperatura

t[h]	0	6	6	9	9	15	15	21	21
$\vartheta_r[^\circ C]$	15	15	20	20	18	18	20	20	15

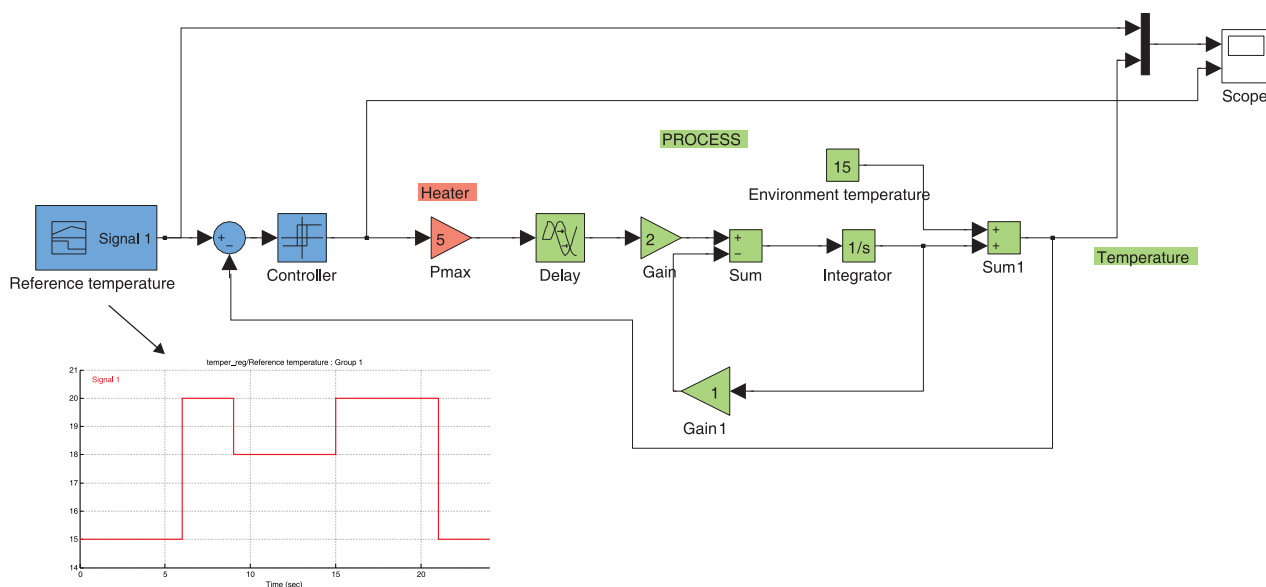
Discontinuous uporabimo **Relay** blok (privzeto ime smo nadomestili z imenom **Controller**). Bloku podamo vrednosti, pri katerih vklopi oz. izklopi ter vklopno in izklopno vrednost. Grelo simuliramo z ojačevalnim blokom **pmax**, dodatno zakasnitev pa z blokom **Transport Delay** iz knjižnice **Continuous**. Simulink shemo prikazuje slika 7.18.

Temperatura v prostoru ter vklapljanje in izklapljanje grela sta prikazana na sliki 7.19.

□

Primer 7.3 Avtomobilsko vzmetenje

Primer 6.2 obravnava razvoj simulacijske sheme avtomobilskega vzmetenja, ki je predhodno opisan v Primeru 3.1. Simulacijsko shemo razvijemo s pomočjo slike 6.7. Potrebujemo tri integratorje, tri ojačevalne bloke in sumator. Tokrat v ojačevalne bloke ne vpišemo številskih vrednosti konstant ampak kar aritmetične izraze, ki se prikažejo tudi znotraj blokovih ikon, če le te dovolj povečamo. Vrednosti za k_1 , k_2 , m in f pa podamo v komandnem oknu Matlaba. Simulacijsko



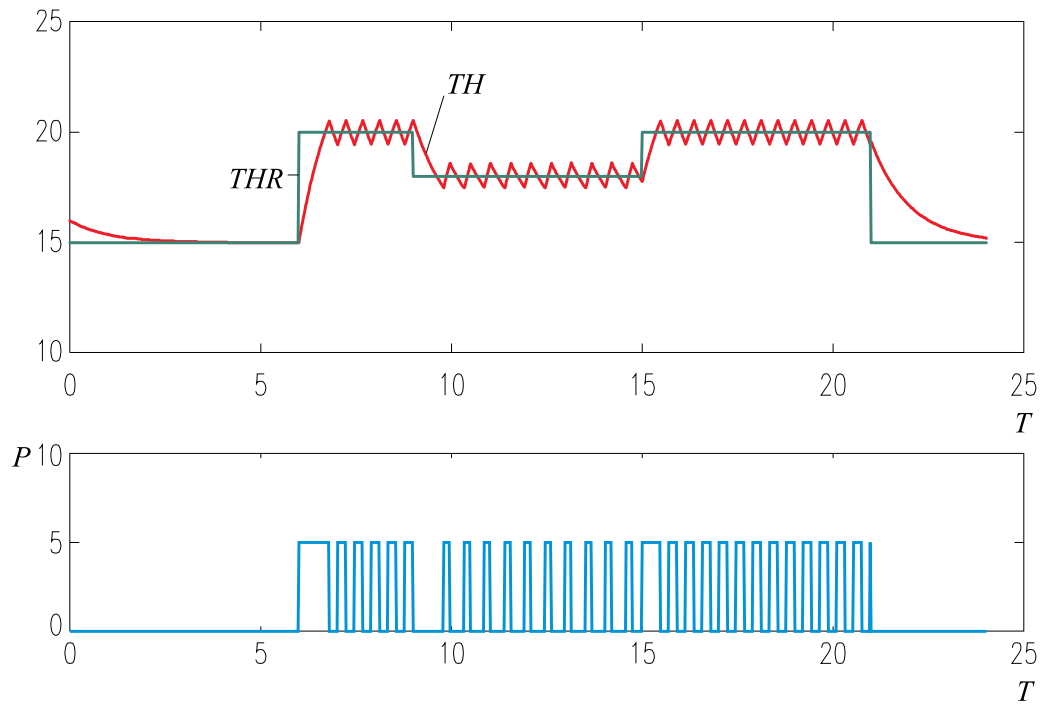
Slika 7.18: Simulacijska shema regulacije temperature v prostoru in prikaz referenčnega signala

shemo prikazuje slika 7.20.

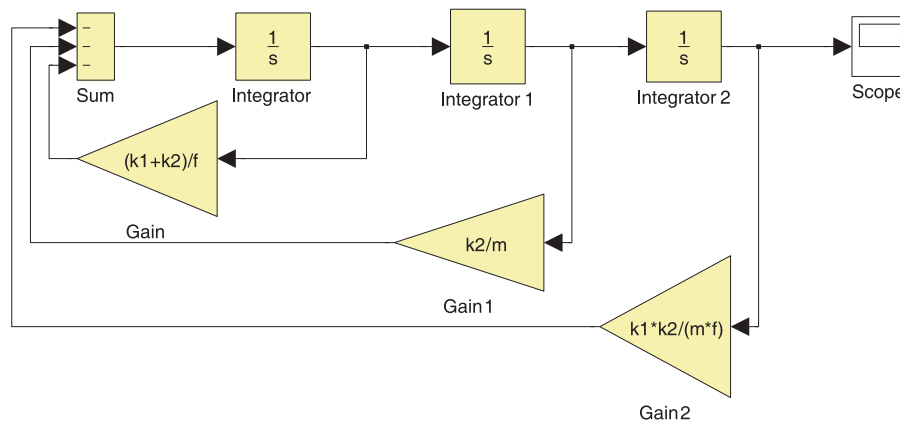
Opazujemo pomik karoserije pri negativnem začetnem pogoju $y(0) = -0.05$ m. Praktično to lahko pomeni, da se je voznik vsedel v avtomobil in v trenutku $t=0$ izstopil iz vozila. Opazujemo, kako se karoserija vrne v mirovno lego. Rezultate simulacije prikazuje slika 7.21.

Izhodiščni podatki za študijo so bili: $m=500$ kg, $k_1=7500$ N/m, $k_2=150000$ N/m, $f=2250$ Ns/m. Ker je prišlo do premalo dušenega odziva, smo povečali togost vzmeti na $k_1=15000$ N/m (2x povečanje), dušenje pa smo tudi 2x povečali ($f=4500$ Ns/m). Tako smo dobili optimalnejše delovanje avtomobilskega vzmetenja.

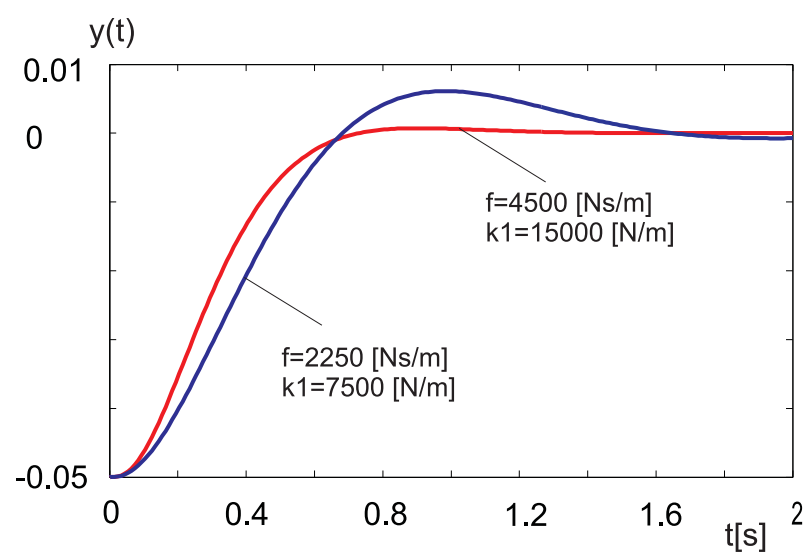
□



Slika 7.19: Rezultati temperaturnega regulacijskega problema



Slika 7.20: Simulink shema modela avtomobilskega vzmetenja



Slika 7.21: Pomik karoserije

8.

Analiza sistemov v časovnem prostoru

Pri analizi dinamičnih sistemov nas pogosto zanima, kakšni so časovni poteki določenih signalov pri vzbujanju sistema. V tem poglavju bomo obravnavali klasične metode časovne analize.

8.1 Osnovne lastnosti časovnega odziva

Časovni odziv dinamičnega sistema delimo v dva dela:

- v prehodni pojav in
- v ustaljeno (stacionarno) stanje,

kar lahko zapišemo z enačbo

$$y(t) = y_t(t) + y_{ss}(t) \tag{8.1}$$

Pri tem je $y_t(t)$ prehodni pojav, $y_{ss}(t)$ pa ustaljeno stanje. Prehodni pojav je del odziva, ki s časom izzveni. Dobimo ga z upoštevanjem homogene diferencialne

enačbe (levo stran diferencialne enačbe izenačimo z nič) in začetnih pogojev. Pod ustaljenim stanjem pa si predstavljamo vedenje sistema, ko gre čas proti neskončnosti. Torej je to del odziva, ki nastane potem, ko prehodni pojav izzveni. Izračunamo ga s pomočjo enačbe

$$y_{ss} = \lim_{t \rightarrow \infty} y(t) = \lim_{s \rightarrow 0} s Y(s) \quad (8.2)$$

ali pa s pomočjo diferencialne enačbe, tako da vse odvode signalov izenačimo z nič. Ni pa nujno, da se časovni signal v ustaljenem stanju s časom ne spreminja. Ta lastnost je odvisna od vrste sistema in od vrste vzbujanja.

Odziv sistema lahko določimo le, če poznamo matematični model (diferencialna enačba, prenosna funkcija, prostor stanj) in vhodni signal. Za razliko od mnogih sistemov (npr. komunikacijskih, električnih, kemijskih, ...) pa pri regulacijskih sistemih običajno vhodnega signala vnaprej ne poznamo. Le-ta je včasih tudi naključnega značaja. Pri radarskem sledilnem sistemu se položaj in hitrost tarče, ki jo je potrebno slediti, spreminja na nepredvidljiv način, ki ga ni možno izraziti z determinističnim opisom. Za načrtovalca predstavlja to določen problem, saj je zelo težko načrtati ustrezen regulacijski sistem, ki bi se zadovoljivo odzival na poljuben vhodni signal. Zato pri analizi in načrtovanju uporabljamo take vhodne signale, ki čim bolj posnemajo dejanske signale med obratovanjem, hkrati pa omogočajo poenostavljen postopek analize in načrtovanja. Tako uporabljamo t.i. polinomske signale (δ impulz - slika 2.29, stopnica - slika 2.21, linearno naraščajoči signal - slika 2.20, parabola, ...), periodične signale (sinusni ali kosinusni signal - sliki 2.24 in 2.25, pravokotni impulz - slika 2.28, vlak impulzov - slika 2.30), pa tudi nekatere naključne signale (pseudonaključni binarni šum, gaussov šum, šum z enakomerno verjetnostno porazdelitvijo, ...).

Pri analizi prehodnega pojava in ustaljenega stanja običajno največ informacij nudi odziv na stopničasto vzbujanje. Odziv na stopničasto vzbujanje ali na δ impulz analiziramo v primeru, če je dinamični sistem podvržen hitrim spremembam vhodnega signala. Če pa so te spremembe počasne, je smiselno uporabiti linearno naraščajoči ali sinusni signal ustrezno nizke frekvence. Posebno problematiko pa predstavlja izbira ustreznega preizkusnega signala pri identifikaciji sistemov. Izbrati je potrebno take preizkusne signale, ki čim bolj vzbuja celotno frekvenčno področje sistema.

S pomočjo znanega zapisa (modela) sistema in znanega vhodnega signala lahko izračunamo odziv na različne načine. Če je podana prenosna funkcija sistema,

dobimo odziv s pomočjo enačbe

$$Y(s) = G(s)U(s) \quad (8.3)$$

kjer sta $Y(s)$ in $U(s)$ Laplace-ova transformata izhodnega in vhodnega signala, $G(s)$ pa prenosna funkcija. Odziv $y(t)$ dobimo z inverzno Laplace-ovo transformacijo:

$$y(t) = \mathcal{L}^{-1}[Y(s)] \quad (8.4)$$

Pomembno vlogo ima inverzni Laplace-ov transform prenosne funkcije $g(t)$

$$g(t) = \mathcal{L}^{-1}[G(s)] \quad (8.5)$$

saj predstavlja odziv sistema pri vzbujanju z δ impulzom. Laplace-ov transform δ impulza je namreč enak 1

$$Y(s) = G(s) \cdot 1 \Rightarrow g(t) = y(t) \quad \text{pri} \quad u(t) = \delta(t) \quad \text{oz.} \quad U(s) = 1 \quad (8.6)$$

Odzivu sistema na δ impulz pravimo naravni odziv sistema.

8.2 Vpliv polov in ničel na časovni odziv

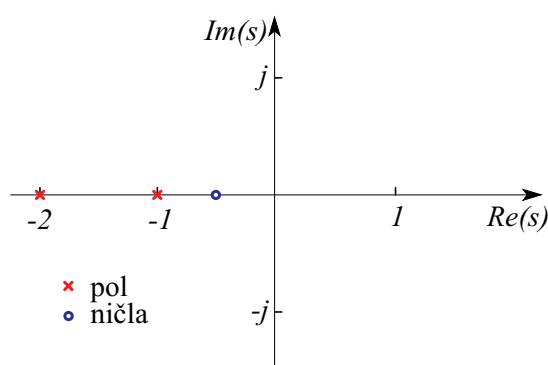
Omenili smo, da je prenosna funkcija kvocient dveh polinomov spremenljivke s . Oglejmo si sistem, ki ga opisuje diferencialna enačba

$$\ddot{y} + 3\dot{y} + 2y = 2\dot{u} + u \quad (8.7)$$

oz. prenosna funkcija

$$G(s) = \frac{Y(s)}{U(s)} = \frac{B(s)}{A(s)} = \frac{2s + 1}{s^2 + 3s + 2} = \frac{2(s + \frac{1}{2})}{(s + 1)(s + 2)} \quad (8.8)$$

Če predpostavljamo, da $B(s)$ in $A(s)$ nimata skupnih korenov (to je običajen primer), potem vrednosti s , za katere velja $A(s) = 0$, naredijo vrednost prenosne funkcije $G(s)$ neskončno. Te vrednosti s imenujemo pole prenosne funkcije $G(s)$. V našem primeru sta pola pri $s = -1$ in $s = -2$. Vrednosti s , za katere pa velja $B(s) = 0$, naredijo vrednost prenosne funkcije $G(s) = 0$, zato jih imenujemo ničle prenosne funkcije. V našem primeru je ničla pri $s = -\frac{1}{2}$. Poli in ničle do multiplikativne konstante natančno določajo prenosno funkcijo, zato le to lahko predstavimo s sliko polov in ničel v ravnini s , kar prikazuje slika 8.1. Tudi to je torej možna predstavitev sistema.



Slika 8.1: Prikaz polov in ničel v s ravnini

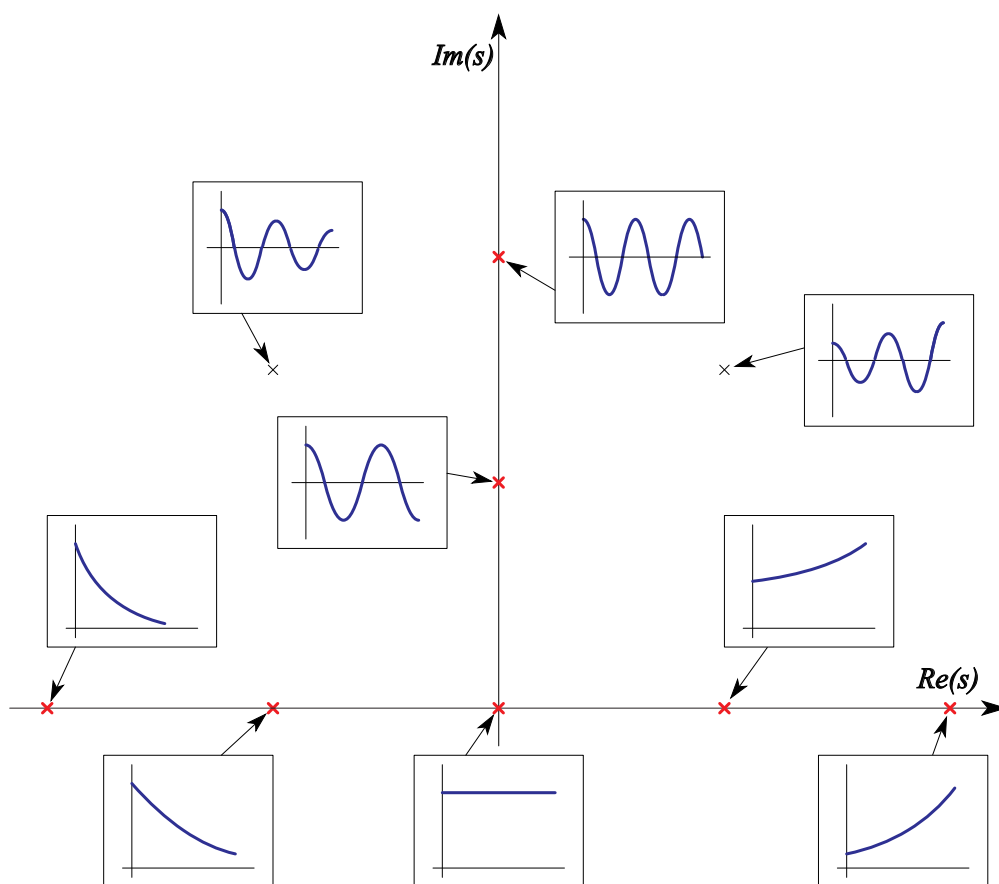
S pomočjo inverzne Laplace-ove transformacije izračunajmo iz prenosne funkcije odziv sistema na δ impulz. Slednje omogoča analizo vpliva polov in ničel na časovni odziv. Prenosno funkcijo $G(s)$ razvijemo v parcialne ulomke

$$G(s) = \frac{2s + 1}{(s + 1)(s + 2)} = \frac{-1}{s + 1} + \frac{3}{s + 2} \quad (8.9)$$

Ustrezni inverzni Laplace-ov transform pa se glasi

$$g(t) = -e^{-t} + 3e^{-2t} \quad (8.10)$$

Vidimo, da je oblika eksponencialnih funkcij, ki sestavljata odziv, odvisna le od polov pri $s = -1$ in $s = -2$. To velja v splošnem tudi za bolj kompleksne sisteme. Ničla sistema, ki oblikuje števec prenosne funkcije, pa skupaj s poli vpliva le na uteži, preko katerih oba člena (e^{-t} in e^{-2t}) vplivata na končni odziv. Slika 8.2 prikazuje vpliv lege polov na naravni odziv sistema.



Slika 8.2: Vpliv polov na naravni odziv sistema

Ker člen e^{-2t} izzveni hitreje kot člen e^{-t} , pravimo, da je pol pri $s = -2$ hitrejši kot pol $s = -1$. Zato včasih govorimo o hitrih in počasnih polih. Zapomniti si velja, da poli, ki so bolj odmaknjeni od imaginarne osi, predstavljajo prehodni pojav, ki hitreje izzveni.

Dominantni poli

Poli v prenosni funkciji so dominantni, če pretežno vplivajo na odziv sistema. V primeru, da sistem nima ničel, je relativna dominantnost definirana kot razmerje realnih delov polov. Če je razmerje realnih delov polov večje kot 4 in če v bližini pola, ki je bližje koordinatnemu izhodišču, ni ničel, potem pol, ki je bližje koordinatnemu izhodišču, dominantno vpliva na časovni potek, kajti predstavlja prehodni pojav, ki počasneje izzveni. Take pole imenujemo dominantni poli.

Često nastopajo v konjugirano kompleksnih parih.

Prenosno funkcije poenostavimo tako, da ohranimo le dominantne pole, ostale pa zanemarimo. Če imamo faktorizirano obliko, nedominantne pole izločimo. Vendar je smiselno, da spremenimo multiplikativno konstanto tako, da se ojačenje prenosne funkcije ne spremeni.

Analiza stabilnosti

Stabilnost dinamičnih sistemov lahko določimo glede na lego polov prenosne funkcije v ravnini s . Če katerikoli pol leži v desnem delu ravnine s , potem časovni signali naraščajo monotono ali pa nihajo z naraščajočo amplitudo. Taki sistemi so nestabilni. Če v takem sistemu ni ustrezne zaščite (nasičenje), potem je nevarnost, da se sistem poškoduje. Zato v dinamičnih sistemih oz. njihovih modelih, zlasti pa v regulacijskih sistemih, poli v desni polravnini ravnine s niso dovoljeni. Prav tako običajno niso dovoljeni poli na imaginarni osi. Čeprav taki poli idealno gledano včasih predstavljajo signal, ki niti ne narašča, niti ne upada, pa moramo računati s tem, da v praksi vedno nastopajo motnje, ki se lahko zaradi pola na imaginarni osi preveč ojačijo.

Stabilnost linearne dinamičnega sistema je lastnost samega sistema in ni odvisna od oblike, velikosti in mesta nastopa vhodnega signala. Sistem je absolutno stabilen, če ima vse pole v levi polravnini.

Z izbiro polov v levi polravnini pa še nismo zagotovili zadovoljivega prehodnega pojava. Ugotovili smo še, da poli v bližini imaginarne osi predstavljajo počasne prehodne pojave. Zato je za primeren prehodni pojav potrebna ustrezna lega polov v levi polravnini s , pomembno vlogo pa imajo tudi ničle. Ustrezna lega polov in ničel je zlasti pomembna pri načrtovanju regulacijskih sistemov.

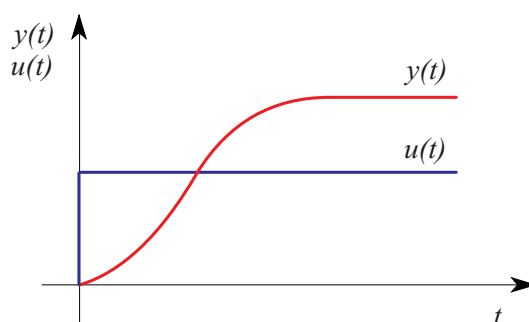
8.3 Razvrstitev sistemov

Proporcionalni, integrirni in diferencirni sistemi

Proporcionalni sistem z vhomom $u(t)$ in izhodom $y(t)$ podaja prenosna funkcija

$$G(s) = \frac{Y(s)}{U(s)} = \frac{B(s)}{A(s)} = \frac{b_0 s^m + b_1 s^{m-1} + \dots + b_m}{s^n + a_1 s^{n-1} + \dots + a_n} \quad m \leq n \quad (8.11)$$

pri čemer velja $B(0) \neq 0$ in $A(0) \neq 0$, torej polinoma v števcu in v imenovalcu ne vsebujeta korenov v koordinatnem izhodišču. Za take sisteme je značilen princip t.i. samoregulacije, kar pomeni, da pri stopničastem vzburjanju preidejo v neko novo, ustaljeno stanje, za katerega je značilna konstantna vrednost izhodnega signala. Tipičen primer predstavlja ogrevanje prostora, ko v njem vključimo grelnik neke konstantne moči. Karakteristični odziv proporcionalnega sistema $y(t)$ pri stopničastem vzburjanju $u(t)$ predstavlja slika 8.3.



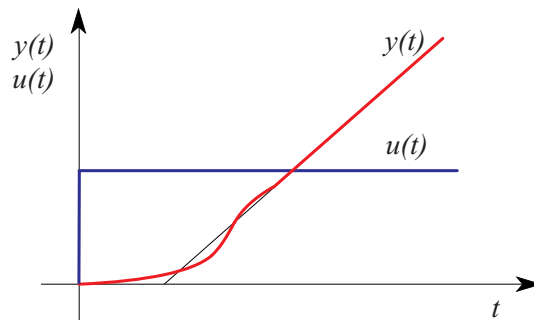
Slika 8.3: Karakteristični odziv proporcionalnega sistema

Če narišemo do integratorjev razgrajeni bločni diagram proporcionalnega sistema, opazimo, da imajo vsi integratorji povratne vezave.

Za *integrirne sisteme* pa je značilno, da vsebuje imenovalec prenosne funkcije enega ali več korenov v koordinatnem izhodišču. Prenosna funkcija ima obliko

$$G(s) = \frac{Y(s)}{U(s)} = \frac{1}{s^j} \frac{N(s)}{D(s)} \quad (8.12)$$

Velja $N(0) \neq 0$ in $D(0) = 0$. Taki sistemi so mejno stabilni in nimajo pravega ustaljenega stanja, saj odziv po prehodnem pojavu narašča pri stopničastem vzburjanju. Tipičen integrirni proces predstavljajo razne vrste enosmernih motorjev, katerih zasuk po prehodnem pojavu linearno narašča pri konstantni vzbujaalni napetosti. Slika 8.4 predstavlja karakterističen odziv integrirnega procesa.



Slika 8.4: Karakteristični odziv integrirnega sistema

Glede na število polov prenosne funkcije (8.12) v koordinatnem izhodišču (j), delimo sisteme na

- $j = 1$ sistemi prve vrste
- $j = 2$ sistemi druge vrste
- \vdots

V skladu s tem lahko proporcionalne sisteme obravnavamo kot sisteme ničte vrste ($j = 0$).

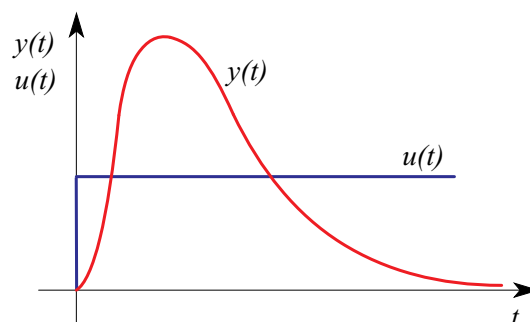
Vrsta sistema odločilno vpliva na vedenje regulacijskih (povratnozančnih) sistemov. Z višanjem vrste se zmanjšuje pogrešek v ustaljenem stanju oz. povečuje točnost regulacijskega sistema. Hkrati pa se povečujejo stabilnostni problemi, tako da je potrebno najti nek kompromis.

Pri *diferencirnih sistemih* vsebuje števec prenosne funkcije enega ali več korenov v koordinatnem izhodišču. Prenosna funkcija ima obliko

$$G(s) = \frac{Y(s)}{U(s)} = \frac{s^j N(s)}{D(s)} \quad (8.13)$$

pri čemer je $N(0) \neq 0$ in $D(0) \neq 0$. Taki sistemi kažejo močno reagiranje na stopničasti vhodni signal takoj na začetku prehodnega pojava. Slika 8.5 prikazuje tipični odziv diferencirnega sistema na stopničasto vzbujanje.

Diferencirni značaj imajo predvsem regulatorji, ki morajo reagirati čim hitreje,



Slika 8.5: Karakteristični odziv diferencirnega sistema

medtem ko za procese to ni značilni tip reagiranja, saj imajo le-ti zaradi t.i. shranjevalnikov mase ali energije v odzivih precejšnje zakasnitve.

Obstajajo tudi sistemi, pri katerih preteče določen čas, predno se pokaže vpliv vhodne veličine na izhodno. Tej lastnosti rečemo *mrtvi čas* (transportna zakasnitev). Vsi od prej obravnavanih vrst sistemov imajo lahko dodatno mrtvi čas.

Red sistema

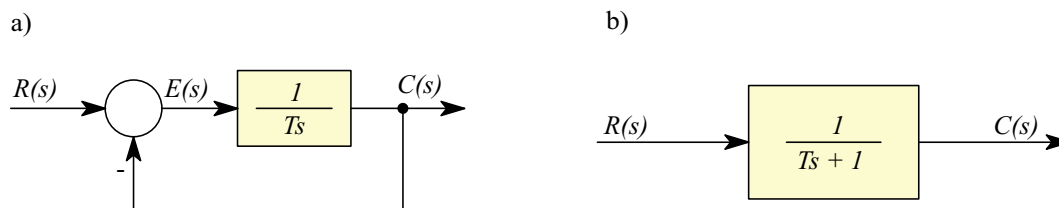
V časovnem prostoru je red sistema določen z redom najvišjega odvoda izhodne spremenljivke v diferencialni enačbi ali s številom spremenljivk stanja pri zapisu v prostoru stanj. Pri zapisu s prenosno funkcijo pa je red sistema določen s stopnjo polinoma v imenovalcu. Glede na to definicijo imamo sisteme ničtega, prvega, drugega in višjih redov.

8.4 Proporcionalni sistemi

8.4.1 Sistem prvega reda

Po zakonitostih sistema prvega reda se vedejo mnogi realni procesi in regulacijski sistemi. V literaturi srečamo tudi oznaki $P1$ ali $PT1$, kjer 1 pomeni eno časovno konstanto. Fizikalno gledano je sistem prvega reda lahko RC vezje, termični sistem, hidravlični sistem itd. Tipični regulacijski sistem (model motorja kot procesa v regulacijski zanki), ki ga je možno ponazoriti s sistemom prvega reda,

prikazuje slika 8.6.



Slika 8.6: Bločni diagram sistema 1. reda: a) povratnozančni bločni diagram
b) poenostavljeni bločni diagram

Sistem 1. reda predstavimo v splošni obliki z diferencialno enačbo

$$T \dot{y}(t) + y(t) = k u(t) \quad (8.14)$$

ali s prenosno funkcijo

$$\frac{Y(s)}{U(s)} = \frac{k}{Ts + 1} \quad (8.15)$$

kjer je k ojačenje sistema, T pa njegova časovna konstanta.

Odziv sistema prvega reda na stopničasto vzbujanje

Ena od temeljnih značilnosti sistema 1. reda je njegov odziv na stopničasto vzbujanje $\frac{U_0}{s}$. Ustrezni Laplace-ov transform se glasi

$$Y(s) = \frac{k}{Ts + 1} \cdot \frac{U_0}{s} \quad (8.16)$$

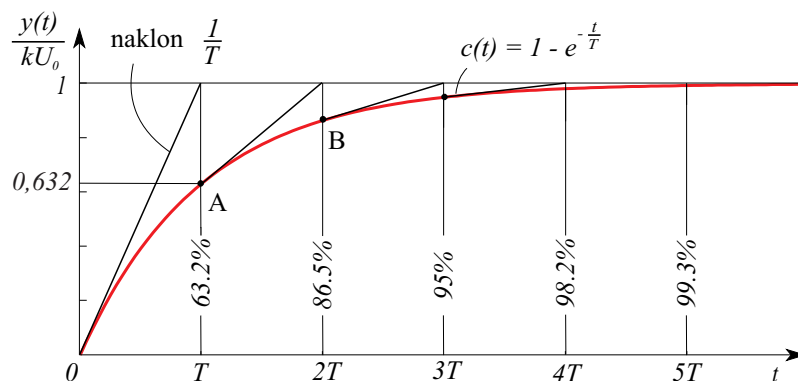
Če razvijemo $Y(s)$ po parcialnih ulomkih, dobimo

$$Y(s) = k U_0 \left[\frac{1}{s} - \frac{T}{Ts + 1} \right] \quad (8.17)$$

oz. odziv

$$y(t) = k U_0 (1 - e^{-\frac{t}{T}}) \quad t \geq 0 \quad (8.18)$$

Slika 8.7 predstavlja karakteristične vrednosti odziva sistema 1. reda, kar s pridom uporabljamo v inženirski praksi.



Slika 8.7: Odziv sistema 1. reda

Značilno je, da ima tangenta na krivuljo v koordinatnem izhodišču naklon $\frac{1}{T}$, ter da odziv doseže 63.2%, 86.5%, 95%, ... končne vrednosti, ko je čas enak $T, 2T, 3T, \dots$. V času $t \geq 4T$ ostane odziv znotraj 2% področja okoli končne vrednosti.

Ojačenje sistema določimo iz meritev tako, da ustaljeno vrednost izhodnega signala delimo s spremembo stopničastega vhodnega signala. Če je $y(0) = 0$, velja

$$k = \frac{y(\infty)}{U_0} = \frac{\lim_{t \rightarrow \infty} k U_0 (1 - e^{-\frac{t}{T}})}{U_0} \quad (8.19)$$

V primeru regulacijskega sistema na sliki 8.6 ($k = 1$) je regulirana veličina

$$c(t) = R_0 (1 - e^{-\frac{t}{T}}) \quad (8.20)$$

in pogrešek

$$e(t) = R_0 - c(t) = R_0 e^{-\frac{t}{T}} \quad (8.21)$$

oz. ustaljeni pogrešek

$$e(\infty) = 0 \quad (8.22)$$

Regulacijski sistem torej nima pogreška v ustaljenem stanju.

Odziv sistema prvega reda na linearno naraščajoči vhodni signal

Ker je Laplace-ov transform linearno naraščajočega vhodnega signala $U_0 t$ enak $\frac{U_0}{s^2}$, je odziv sistema 1. reda

$$Y(s) = \frac{k}{Ts + 1} \cdot \frac{U_0}{s^2} \quad (8.23)$$

oz.

$$Y(s) = k U_0 \left[\frac{1}{s^2} - \frac{T}{s} + \frac{T^2}{Ts + 1} \right] \quad (8.24)$$

kar da rešitev

$$y(t) = k U_0 \left[t - T + T e^{-\frac{t}{T}} \right] \quad t \geq 0 \quad (8.25)$$

V primeru regulacijskega sistema na sliki 8.6 ($k=1$) je regulirana veličina

$$c(t) = R_0 \left[t - T + T e^{-\frac{t}{T}} \right] \quad (8.26)$$

in pogrešek

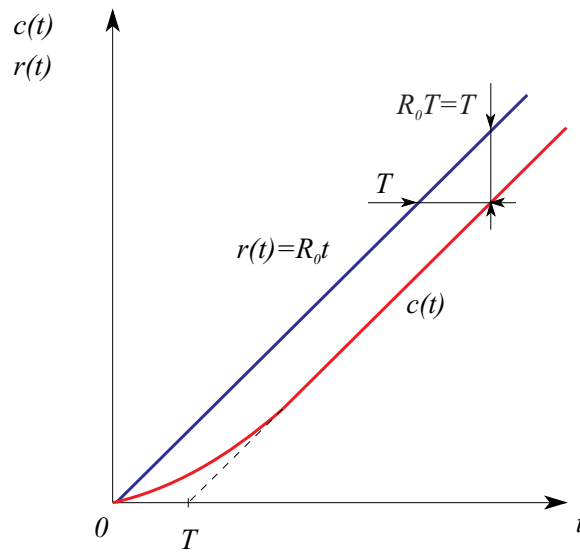
$$e(t) = R_0 t - c(t) = R_0 T (1 - e^{-\frac{t}{T}}) \quad (8.27)$$

oz. ustaljeni pogrešek

$$e(\infty) = R_0 T \quad (8.28)$$

ki ima torej konstantno vrednost. Slika 8.8 prikazuje odziv pri vzbujanju z linearno naraščajočim signalom.

Manjša kot je časovna konstanta T (večje je ojačenje zanke), manjši je pogrešek v ustaljenem stanju.



Slika 8.8: Odziv pri linearno naraščajočem vhodnem signalu

Odziv sistema prvega reda na δ impulz (naravni odziv)

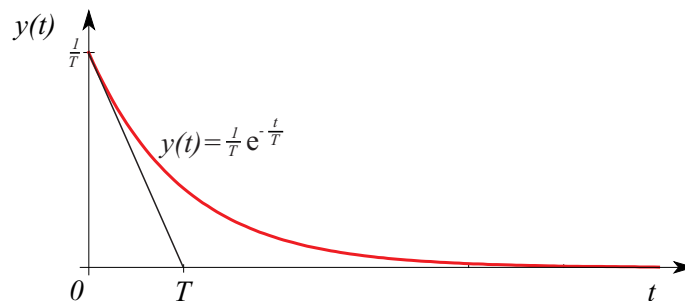
Ker je Laplace-ov transform δ impulza 1, določimo izhodni signal z enačbo

$$Y(s) = \frac{k}{Ts + 1} \quad (8.29)$$

oz.

$$y(t) = \frac{k}{T} e^{-\frac{t}{T}} \quad t \geq 0 \quad (8.30)$$

Ustrezen odziv pri ojačenju $k = 1$ prikazuje slika 8.9.

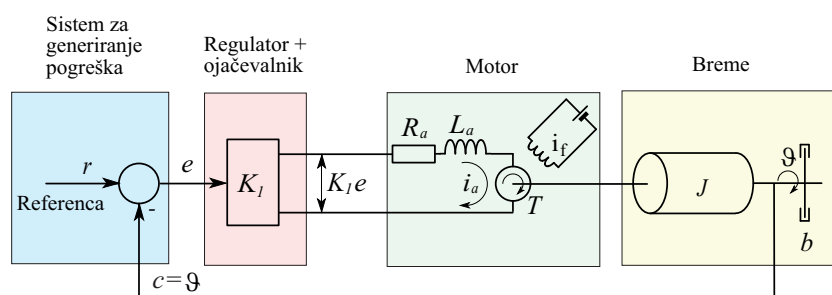
Slika 8.9: Odziv sistema 1. reda na δ impulz

8.4.2 Sistem drugega reda

Zakoni sistema drugega reda ($P2$ sistem) si bomo ogledali na primerih 8.1, 8.2 in 8.3.

Primer 8.1 Regulacije zasuka s pomočjo enosmernega motorja

Regulacije zasuka s pomočjo enosmernega motorja v servosistemu prikazuje slika 8.10.



Slika 8.10: Regulacija zasuka

Pri konstantnem statorskem toku i_f je moment T , ki ga generira motor, proporcionalen rotorskemu toku

$$T = k_2 i_a \quad (8.31)$$

pri čemer je k_2 momentna konstanta motorja. Za rotorski tokokrog pa velja enačba

$$L_a \frac{di_a}{dt} + R_a i_a + k_3 \frac{d\vartheta}{dt} = k_1 e \quad (8.32)$$

kjer je k_3 induksijska konstanta in $k_3 \frac{d\vartheta}{dt}$ ustrezna inducirana napetost, ki je seveda proporcionalna hitrosti vrtenja motorja. L_a je induktivnost, R_a pa upornost rotorskega tokokroga. Momentna ravnotežna enačba pa ima obliko

$$J \frac{d^2\vartheta}{dt^2} + b \frac{d\vartheta}{dt} = T = k_2 i_a \quad (8.33)$$

Pri tem je J vztrajnostni moment motorja in bremena, b pa je ustrezeni koeficient viskoznega dušenja. S pomočjo enačb (8.32) in (8.33) izpeljemo prenosno funkcijo med zasukom in pogreškom, tako da eliminiramo tok i_a

$$G(s) = \frac{\Theta(s)}{E(s)} = \frac{k_1 k_2}{s(L_a s + R_a)(Js + b) + k_2 k_3 s} \quad (8.34)$$

Ob zanemaritvi induktivnosti rotorja L_a , ki je običajno majhna, lahko prenosno funkcijo direktne veje napišemo v obliki

$$G(s) = \frac{\frac{k_1 k_2}{R_a}}{Js^2 + (b + \frac{k_2 k_3}{R_a})s} \quad (8.35)$$

Če označimo

$$\begin{aligned} K &= \frac{k_1 k_2}{R_a} \\ B &= b + \frac{k_2 k_3}{R_a} \end{aligned} \quad (8.36)$$

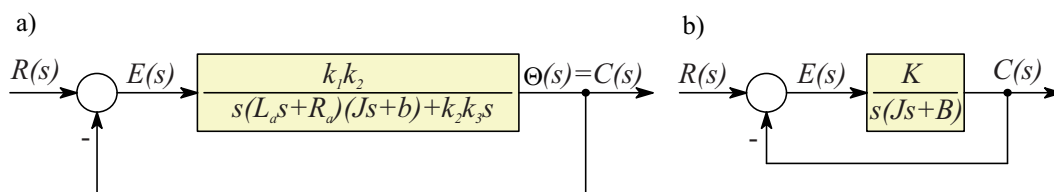
velja prenosna funkcija direktne veje (motorja)

$$G(s) = \frac{K}{s(Js + B)} \quad (8.37)$$

Slika 8.11 prikazuje bločni in poenostavljeni bločni diagram servosistema. Ob upoštevanju zveze $E(s) = R(s) - \Theta(s) = R(s) - C(s)$ oz. z uporabo bločnega diagrama 8.11b pridemo do zaprtozančne prenosne funkcije

$$\frac{C(s)}{R(s)} = \frac{\frac{K}{J}}{s^2 + \frac{B}{J}s + \frac{K}{J}} \quad (8.38)$$

Faktorizirana oblika prenosne funkcije (8.38) pa se glasi



Slika 8.11: a) bločni diagram
b) poenostavljeni bločni diagram

$$\frac{C(s)}{R(s)} = \frac{\frac{K}{J}}{\left[s + \frac{B}{2J} + \sqrt{\left(\frac{B}{2J}\right)^2 - \frac{K}{J}}\right] \left[s + \frac{B}{2J} - \sqrt{\left(\frac{B}{2J}\right)^2 - \frac{K}{J}}\right]} \quad (8.39)$$

Vidimo, da so povratnozančni poli kompleksni, če je $D = B^2 - 4JK < 0$ oz. realni, če je $D = B^2 - 4JK \geq 0$. Definirajmo

$$\frac{K}{J} = \omega_n^2, \quad \frac{B}{J} = 2\zeta\omega_n, \quad \frac{B}{2J} = \zeta\omega_n = \sigma = \frac{1}{\tau} \quad (8.40)$$

Pri tem je ω_n lastna frekvenca nedušenega sistema in ζ dušilni koeficient, ki predstavlja razmerje med dušenjem sistema B proti kritičnemu dušenju $B_{kr} = 2\sqrt{JK}$ ($B_{kr}^2 - 4JK = 0$)

$$\zeta = \frac{B}{B_{kr}} = \frac{B}{2\sqrt{JK}} \quad (8.41)$$

Z upoštevanjem omenjenih definicij lahko prenosno funkcijo zapišemo v standardni obliki za sistem drugega reda ($C(s)$ in $R(s)$ zamenjamo z $Y(s)$ in $U(s)$)

$$\frac{Y(s)}{U(s)} = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \quad (8.42)$$

□

Primer 8.2 RLC vezje

Prenosna funkcija RLC vezja med priključeno napetostjo in napetostjo na kondenzatorju je izpeljana v primeru 4.5 in jo podaja enačba 4.52

$$G_2(s) = \frac{E_o(s)}{E_i(s)} = \frac{\frac{1}{LC}}{s^2 + \frac{R}{L}s + \frac{1}{LC}} \quad (8.43)$$

Poli so kompleksni, če je $D = (\frac{R}{L})^2 - \frac{4}{LC} < 0$ oz. realni, če je $D = (\frac{R}{L})^2 - \frac{4}{LC} \geq 0$.

Definirajmo

$$\frac{1}{LC} = \omega_n^2 \quad \frac{R}{L} = 2\zeta\omega_n \quad (8.44)$$

Pri tem je ω_n lastna frekvenca nedušenega sistema in ζ dušilni koeficient, ki predstavlja razmerje med upornostjo R proti kritični upornosti (dušenju) $R_{kr} = 2\sqrt{\frac{C}{L}}$ ($R_{kr}^2 - 4\frac{L}{C} = 0$)

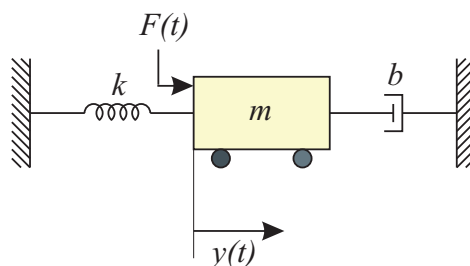
$$\zeta = \frac{R}{R_{kr}} = \frac{R}{2} \sqrt{\frac{C}{L}} \quad (8.45)$$

Če vstavimo definiciji 8.44 v enačbo 8.43 in uporabimo posplošena imena ($E_o(s)$ in $E_i(s)$ zamenjamo z $Y(s)$ in $U(s)$), dobimo normirano obliko 8.42.

□

Primer 8.3 Mehanski sistem

Slika 8.12 pa prikazuje mehanski sistem z vzmetjo, maso in dušilnikom.



Slika 8.12: Mehanski sistem

S pomočjo teoretičnega modeliranja pridemo v nekaj korakih do prenosne funkcije $G(s) = \frac{Y(s)}{F(s)}$

$$-ky(t) - by(t) + F(t) = m\ddot{y}(t) \quad (8.46)$$

$$\ddot{y}(t) + \frac{b}{m}\dot{y}(t) + \frac{k}{m}y(t) = \frac{1}{m}F(t) \quad (8.47)$$

$$s^2Y(s) + \frac{b}{m}sY(s) + \frac{k}{m}Y(s) = \frac{1}{m}F(s) \quad (8.48)$$

$$G(s) = \frac{Y(s)}{F(s)} = \frac{\frac{1}{m}}{s^2 + \frac{b}{m}s + \frac{k}{m}} \quad (8.49)$$

Ponovno smo dobili enako strukturo. Prenosna funkcija se loči le po tem, da nima ojačenja 1, kot pri prejšnjih dveh primerih.

Poli so kompleksni, če je $b^2 - 4mk < 0$ oz. realni, če je $b^2 - 4mk \geq 0$.

Definirajmo

$$\frac{k}{m} = \omega_n^2 \quad \frac{b}{m} = 2\zeta\omega_n \quad (8.50)$$

Pri tem je ω_n lastna frekvenca nedušenega sistema in ζ dušilni koeficient, ki predstavlja razmerje med dušenjem b in kritičnim dušenjem $b_{kr} = 2\sqrt{mk}$ ($b_{kr}^2 - 4mk = 0$)

$$\zeta = \frac{b}{b_{kr}} = \frac{b}{2\sqrt{mk}} \quad (8.51)$$

Če vstavimo definiciji 8.50 v enačbo 8.49 in uporabimo posplošena imena ($F(s)$ zamenjamo z $U(s)$), dobimo ponovno normirano obliko 8.42.

□

Analiza časovnega odziva

Torej na dinamično obnašanje sistema drugega reda vplivata dva parametra: dušilni koeficient ζ in lastna frekvenca nedušenega nihanja ω_n . Z ozirom na velikost parametra ζ lahko dinamično obnašanje razdelimo v štiri razrede:

$\zeta = 0$	nedušeno nihanje	$(B = 0, \quad R = 0)$
$\zeta < 1$	dušeno nihanje	$(B < B_{kr}, \quad R < R_{kr})$
$\zeta = 1$	meja aperiodičnosti	$(B = B_{kr}, \quad R = R_{kr})$
$\zeta > 1$	aperiodični odziv	$(B > B_{kr}, \quad R > R_{kr})$

a) **Dušeno nihanje** (podkritično dušenje $0 < \zeta < 1$)

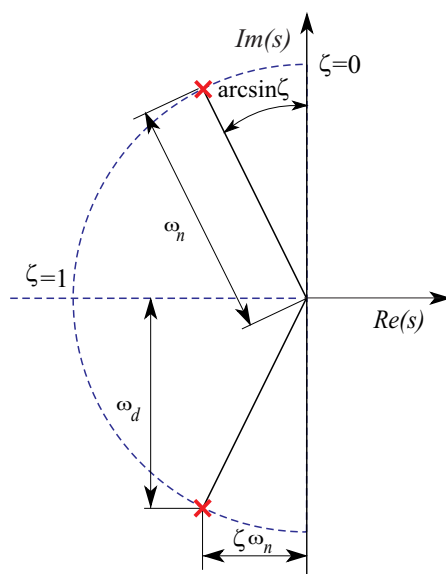
V tem primeru lahko zapišemo $Y(s)/U(s)$ kot

$$\frac{Y(s)}{U(s)} = \frac{\omega_n^2}{(s + \zeta\omega_n + j\omega_d)(s + \zeta\omega_n - j\omega_d)} \quad (8.52)$$

pri čemer je

$$\omega_d = \omega_n \sqrt{1 - \zeta^2} \quad (8.53)$$

lastna frekvenca dušenega nihanja. Ustrezno lego polov prikazuje slika 8.13.



Slika 8.13: Lega polov pri sistemu drugega reda pri $0 < \zeta < 1$

časovni odziv sistema pri enotini stopnici ($U(s) = \frac{1}{s}$) podaja enačba

$$\begin{aligned} y(t) &= 1 - e^{-\zeta\omega_n t} \left(\cos \omega_d t + \frac{\zeta}{\sqrt{1 - \zeta^2}} \sin \omega_d t \right) = \\ &= 1 - \frac{e^{-\zeta\omega_n t}}{\sqrt{1 - \zeta^2}} \sin \left(\omega_d t + \arctg \frac{\sqrt{1 - \zeta^2}}{\zeta} \right) \quad t \geq 0 \quad (8.54) \end{aligned}$$

V ustaljenem stanju gre odziv proti 1, torej v primeru regulacijskega sistema (primer 8.1) ni nikakršnega pogreška.

b) Nedušeno nihanje ($\zeta = 0$)

V primeru, če je dušilni koeficient $\zeta = 0$, se pola nahajata na imaginarni osi in dobimo prenosno funkcijo

$$\frac{Y(s)}{U(s)} = \frac{\omega_n^2}{s^2 + \omega_n^2} \quad (8.55)$$

in časovni odziv

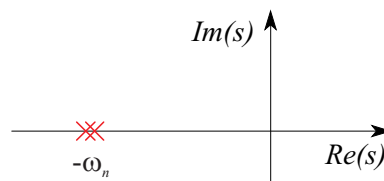
$$y(t) = 1 - \cos \omega_n t \quad t \geq 0 \quad (8.56)$$

Iz te enačbe nazorno vidimo, da je frekvenca ω_n tista frekvenca, s katero bi sistem nihalo, če ne bi imel dušenja. Ker pa ima vsak realni sistem določeno dušenje, dušeno niha s frekvenco $\omega_d = \omega_n \sqrt{1 - \zeta^2}$, ki je nekoliko nižja od frekvence ω_n . V praksi lahko merimo le lastno frekvenco dušenega nihanja ω_d .

c) Meja aperiodičnosti (kritično dušenje $\zeta = 1$)

Če sta dva pola prenosne funkcije $Y(s)/U(s)$ približno enaka (slika 8.14), potem lahko sistem aproksimiramo s kritično dušenim sistemom

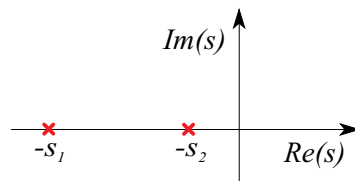
$$\frac{Y(s)}{U(s)} = \frac{\omega_n^2}{(s + \omega_n)^2} \quad (8.57)$$



Slika 8.14: Lega polov pri $\zeta = 1$

Odziv pri vzbujanju z enotino stopnico pa se glasi

$$y(t) = 1 - e^{-\omega_n t} (1 + \omega_n t) \quad t \geq 0 \quad (8.58)$$

Slika 8.15: Lega polov pri $\zeta > 1$ d) **Aperiodični odziv** (nadkritično dušenje $\zeta > 1$)

V tem primeru ležijo poli, kot prikazuje slika 8.15. Regulacijski sistem ima dva neenaka pola na negativni realni osi. Prenosna funkcija je

$$\frac{Y(s)}{U(s)} = \frac{\omega_n^2}{(s + \zeta\omega_n + \omega_n\sqrt{\zeta^2 - 1})(s + \zeta\omega_n - \omega_n\sqrt{\zeta^2 - 1})} \quad (8.59)$$

oziroma odziv

$$y(t) = 1 + \frac{\omega_n}{2\sqrt{\zeta^2 - 1}} \left(\frac{e^{-s_1 t}}{s_1} - \frac{e^{-s_2 t}}{s_2} \right) \quad t \geq 0 \quad (8.60)$$

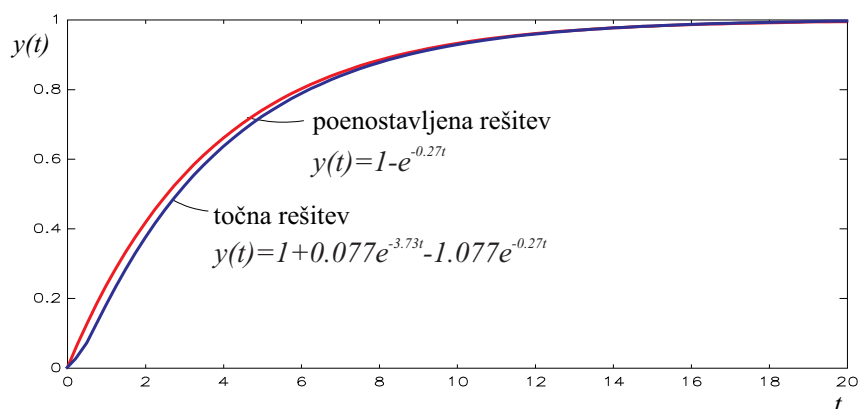
Pri tem je $s_1 = [\zeta + \sqrt{\zeta^2 - 1}]\omega_n$ in $s_2 = [\zeta - \sqrt{\zeta^2 - 1}]\omega_n$. Torej vsebuje odziv dva eksponentno upadajoča člena. Če je $\zeta \gg 1$, en eksponentni člen upada mnogo hitreje, kot drugi, tako da ga lahko zanemarimo (člen z manjšo časovno konstanto ima tudi precej manjšo utež). V obravnavanem primeru je pri $\zeta \gg 1$ pol $-s_2$ precej bližje koordinatnemu izhodišču kot pol $-s_1$. Zato lahko prenosno funkcijo (8.59) poenostavimo, tako da pola $-s_1$ ne upoštevamo, ojačenje preostalega dela pa spremenimo, da je enako vrednosti pred poenostavitvijo. Ojačenje sistema dobimo tako, da v prenosno funkcijo za s vstavimo 0. Ker je pri splošno definiranim sistemu drugega reda, iz katerega izhajamo, ojačenje $Y(0)/U(0) = 1$, je prenosna funkcija poenostavljenega sistema

$$\frac{Y(s)}{U(s)} = \frac{s_2}{s + s_2} \quad Y(s) = \frac{1}{s} - \frac{1}{s + s_2} \quad (8.61)$$

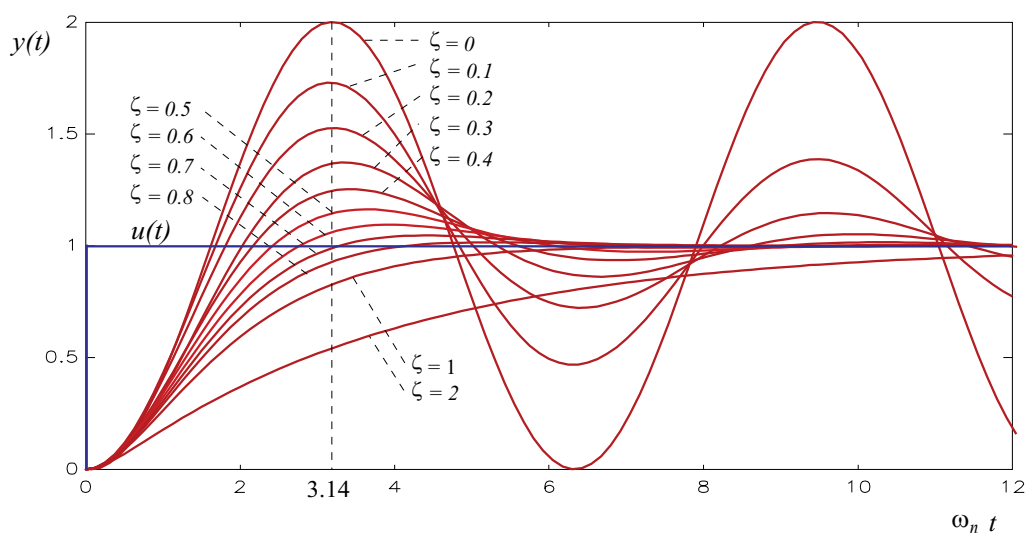
in ustrezen časovni odziv pri vzbujanju z enotino stopnico

$$y(t) = 1 - e^{-s_2 t} = 1 - e^{-(\zeta - \sqrt{\zeta^2 - 1})\omega_n t} \quad t \geq 0 \quad (8.62)$$

Slika 8.16 prikazuje točno in poenostavljeno rešitev pri $\zeta = 2$ in $\omega_n = 1$.



Slika 8.16: Točna in poenostavljena rešitev sistema drugega reda pri vzbujanju z enotno stopnico ($\zeta = 2$, $\omega_n = 1$)



Slika 8.17: Odziv sistema drugega reda v odvisnosti od dušilnega koeficienta

Družino krivulj $y(t)$ za različne ζ prikazuje slika 8.17. Pri tem je na abscisi brezdimenzijska veličina $\omega_n t$. Odziv je v takem koordinatnem sistemu torej odvisen le od dušilnega koeficienta. Opazimo, da sistemi z $0.5 < \zeta < 0.8$ najhitreje dosežejo bližino referenčne vrednosti in v bližini tudi ostanejo. Med aperiodičnimi odzivi pa je najhitrejši odziv pri kritičnem dušenju. Sicer pa aperiodično dušeni sistemi relativno počasi reagirajo na katerekoli vhodne signale.

8.5 Integrirni sistemi

Do sedaj obravnavani sistemi so imeli to skupno lastnost, da se je izhodna veličina približala neki konstantni vrednosti pri stopničastem vzbujujanju. To pa ne velja za integrirne sisteme. Za njih je značilno, da vsebuje prenosna funkcija enega ali več polov v koordinatnem izhodišču, torej je sistem mejno stabilen. Zato integrirni karakter delovanja nikoli ni značilen za vedenje celotnega regulacijskega sistema ampak le za določene komponente, npr. za regulator ali za proces. Najenostavnejši integrirni sistem je t.i. *I0* sistem ali integrirni sistem brez zakasnitve. Zaradi nezakasnjene delovanja srečamo tak sistem predvsem kot del regulatorja. Delovanje sistema določa enačba

$$y(t) = k_I \int u(t) dt \quad (8.63)$$

kjer je $y(t)$ izhodni signal, $u(t)$ vhodni signal, k_I pa ojačenje integrirnega sistema. Prenosna funkcija ima obliko

$$G(s) = \frac{Y(s)}{U(s)} = \frac{k_I}{s} \quad (8.64)$$

Odziv na enotino stopnico pa je

$$y(t) = k_I t \quad (8.65)$$

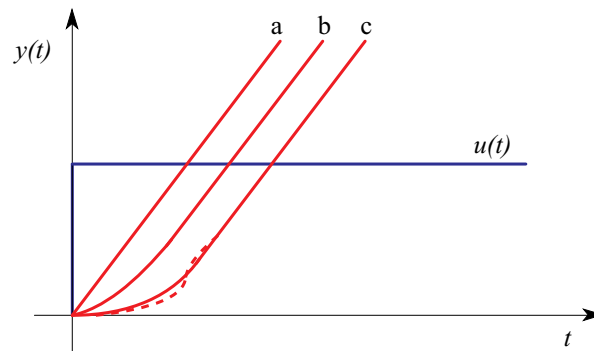
Kadar pa imajo integrirni značaj procesi, se zaradi shranjevalnikov energije pojavi zakasnjeno delovanje (dodatna zakasnitev prvega, drugega, ... višjega reda). Tipičen primer predstavlja prenosna funkcija enosmernega motorja, ki jo podaja enačba (8.37) oz. bločni diagram 8.11b

$$G(s) = \frac{Y(s)}{U(s)} = \frac{K}{s(Js + B)} \quad (8.66)$$

To je integrirni sistem z zakasnitvijo prvega reda ali *I1* sistem. Odziv motorja (zasuk) pri stopničastem vzbujujanju v rotorskem tokokrogu pa podaja enačba

$$y(t) = \frac{KJ}{B^2} \left(\frac{B}{J}t - 1 + e^{-\frac{B}{J}t} \right) \quad (8.67)$$

Za vse integrirne sisteme, ki imajo en pol v koordinatnem izhodišču (sistem prve vrste), je torej značilno, da v "ustaljenem stanju" pri stopničastem vzbujanju odziv linearano narašča. Ta lastnost je razvidna tudi iz slike 8.18, ki prikazuje odziv $I0$, $I1$ in $I2$ sistemov pri vzbujanju s stopničastim signalom.



Slika 8.18: Odzivi različnih integrirnih sistemov prve vrste na stopnico:

- a) $I0$ sistem
- b) $I1$ sistem
- c) $I2$ sistem (nihajoči in aperiodični odziv)

Za $I2$ sistem je značilno, da v prehodnem pojavu deluje tudi sistem drugega reda, ki ima lahko nihajoč ali pa aperiodičen značaj.

Analizirali smo le sisteme prve vrste. Ustrezno tej analizi bi lahko analizirali tudi sisteme višje vrste.

8.6 Diferencirni sistemi

Medtem ko predstavnike integrirnih sistemov najdemo tako med regulatorji kot med procesi, pa se diferencirni člen zaradi hitrega odziva običajno uporablja le kot komponenta regulatorja ali krmilnega sistema. Značilnost takega sistema je v tem, da je izhodna veličina odvisna od odvoda vhodne veličine. Zato ima prenosna funkcija eno ali več ničel v koordinatnem izhodišču. Najenostavnejši diferencirni sistem je idealni diferenciator ali $D0$ sistem, ki ga opisuje diferencialna enačba

$$y(t) = k_D \frac{du(t)}{dt} \quad (8.68)$$

oz. prenosna funkcija

$$G(s) = \frac{Y(s)}{U(s)} = k_D s \quad (8.69)$$

kjer je $y(t)$ izhodna spremenljivka, $u(t)$ vhodna spremenljivka in k_D ojačenje diferencirnega sistema. Odziv $D0$ sistema na enotino stopnico pa ima v časovnem prostoru obliko Dirac-ovega impulza

$$y(t) = k_D \delta(t) \quad (8.70)$$

Idealnega diferenciatorja v praksi ni možno realizirati. Sistem, ki ga je možno realizirati, ima dodatno zakasnitev (zakasnitev prvega reda) in ga imenujemo $D1$ sistem. Le-ta je opisan z diferencialno enačbo

$$T \frac{dy(t)}{dt} + y(t) = k_D \frac{du(t)}{dt} \quad (8.71)$$

oz. s prenosno funkcijo

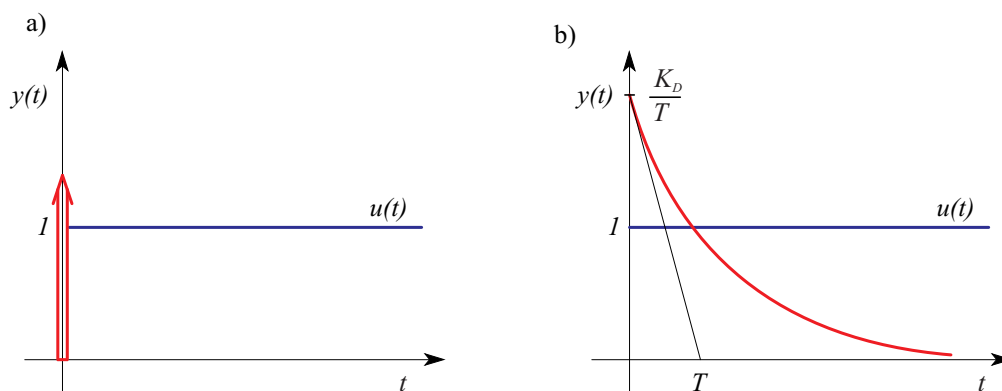
$$G(s) = \frac{Y(s)}{U(s)} = \frac{k_D s}{Ts + 1} \quad (8.72)$$

Časovni odziv sistema $D1$ pri vzburjanju z enotino stopnico podaja izraz

$$y(t) = \frac{k_D}{T} e^{-\frac{t}{T}} \quad (8.73)$$

Slika 8.19 prikazuje odziv idealnega $D0$ in realnega diferencirnega sistema $D1$ pri vzburjanju s stopničastim signalom.

Čim bolj zmanjšamo časovno konstanto T , tem bolj se člen $D1$ približuje členu $D0$ oz. idealnemu diferenciatorju. Vendar moramo pri tem upoštevati, da so signali v praksi omejeni (npr. omejenost izvršnih členov). Slabost diferenciranja je tudi v tem, da ojačuje visokofrekvenčne motnje.

Slika 8.19: Odziv sistema $D0$ (a) in $D1$ (b) na stopnico

8.7 Sistemi z mrtvim časom

Obstajajo tudi sistemi, pri katerih preteče določen čas, preden se pokaže vpliv vhodne veličine na izhodno. Najpogosteje nastopajo mrtvi časi kot transportne zakasnitve (tekoči trakovi, elevatorji, polži, cevovodi,...). Sistem, ki vsebuje samo mrtvi čas, podaja enačba

$$y(t) = u(t - T_m) \quad (8.74)$$

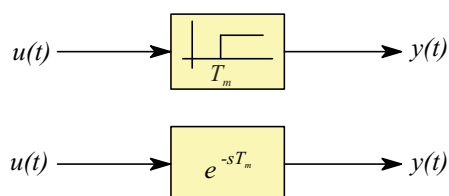
kjer je $y(t)$ izhodna veličina, $u(t)$ vhodna veličina in T_m mrtvi čas. Prenosno funkcijo takega sistema dobimo z upoštevanjem teorema premika in ima obliko

$$G(s) = \frac{Y(s)}{U(s)} = \frac{U(s)e^{-sT_m}}{U(s)} = e^{-sT_m} \quad (8.75)$$

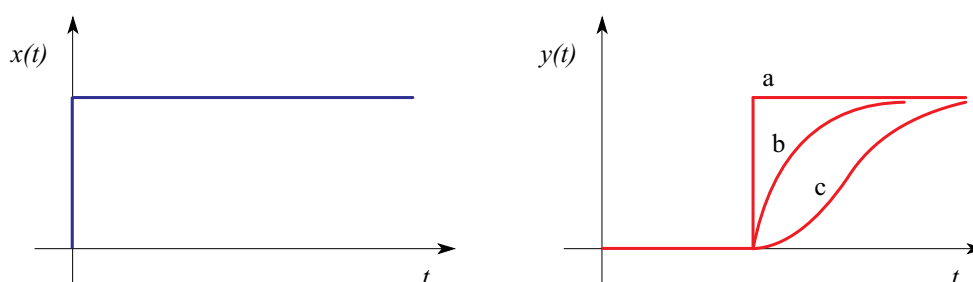
V bločnem diagramu ga predstavimo z eno od ikon, ki sta prikazani na sliki 8.20.

Razen samo mrtvega časa lahko sistemi dodatno (zaporedno) vsebujejo komponente vseh vrst do sedaj obravnavanih sistemov. Slika 8.21 prikazuje odzive proporcionalnih sistemov z mrtvim časom pri stopničastem vzbujanju.

Analitično reševanje sistemov z mrtvim časom je bistveno zahtevnejše od reševanja do sedaj obravnavanih sistemov.



Slika 8.20: Ikoni za sistem z idealnim mrtvim časom



Slika 8.21: Vhodni in izhodni signali sistemov z mrtvim časom:

- a) samo mrtvi čas
- b) mrtvi čas in $P1$ sistem
- c) mrtvi čas in $P2$ sistem

8.8 Stabilnost sistemov

Pri načrtovanju kakršnega koli dinamičnega, zlasti pa regulacijskega sistema je najpomembnejše, da je le-ta stabilen. Zato je preučevanje stabilnosti osrednjega pomena tudi pri analizi dinamičnih sistemov.

8.8.1 Pogoji za stabilnost linearnih, časovno nespremenljivih sistemov

V tem poglavju bomo obravnavali stabilnost linearnih časovno nespremenljivih sistemov, ki so vzbujani z vhodnimi signali. Analiza stabilnosti bo temeljila na legi polov prenosne funkcije dinamičnega sistema

$$G(s) = \frac{Y(s)}{U(s)} = \frac{B(s)}{A(s)} \quad (8.76)$$

oz. na legi korenov karakteristične enačbe

$$A(s) = 0 \quad (8.77)$$

V tem poglavju se bomo omejili na eno izmed osnovnih definicij stabilnosti. Sistem je stabilen, če je izhodni signal omejen pri kakršnem koli omejenem vhodnem signalu (BIBO stabilnost; bounded input, bounded output). Označimo s $y(t)$ izhodno veličino in z $u(t)$ vhodno veličino. Če velja

$$|u(t)| \leq N < \infty \quad \text{za } t \geq t_0 \quad (8.78)$$

potem mora za stabilen sistem veljati

$$|y(t)| \leq M < \infty \quad \text{za } t \geq t_0 \quad (8.79)$$

Pri tem je t_0 čas začetka opazovanja, t pa poljubni čas, N in M pa sta realni števili. Da se dokazati, da BIBO stabilnost vodi do zahtev, ki jih že poznamo v zvezi s stabilnostjo sistemov, t.j. da koreni karakteristične enačbe v primeru stabilnega sistema ležijo v levem delu ravnine s .

Pogoj, da morajo poli ležati v levi polravnini, velja splošno za linearne časovno nespremenljive sisteme. Stabilnost je pri linearnih sistemih lastnost sistema in je neodvisna od oblike, velikosti in mesta nastopa vhodnega ali več vhodnih signalov.

Najbolj naravno torej proučujemo stabilnost tako, da določimo lego polov oz. korenov karakteristične enačbe. Z elementarno matematiko znamo to izračunati za največ sisteme 2. reda. V okolju Matlab pa lahko določimo korene za poljubno visok red sistema. Če nimamo ustrezne računalniške podpore, lahko uporabimo nekatere druge stabilnostne kriterije. Splošno uporaben kriterij je Routhov stabilnostni kriterij.

8.8.2 Routhov stabilnostni kriterij

Routhov stabilnostni kriterij omogoča določiti število polov prenosne funkcije, ki ležijo na desni strani ravnine s , ne da bi določili lego teh polov. Postopek je

naslednji:

1. Napišemo karakteristično enačbo $A(s) = 0$ v obliki

$$a_0 s^n + a_1 s^{n-1} + a_2 s^{n-2} + \dots + a_n = 0 \quad (8.80)$$

2. Potreben, ne pa tudi zadosten pogoj za stabilnost je ta, da so vsi koeficienti enačbe (8.80) pozitivni (lahko so tudi vsi negativni, saj v tem primeru z množenjem polinoma z -1 vsi postanejo pozitivni). Pogoj je razumljiv, če si predstavljamo polinom (8.80) v faktorizirani obliki, v kateri nastopajo členi prvega reda v obliki $s + a$ in členi drugega reda v obliki $s^2 + bs + c$. Da ležijo koreni v levem delu ravnine s , morajo biti a , b in c pozitivni, z množenjem poljubnih kombinacij takih členov pa dobimo tudi karakteristični polinom s pozitivnimi koeficienti. Če nas zanima le stabilnost sistema, lahko v primeru, če niso vsi koeficienti enakega predznaka, postopek prekinemo.
3. V primeru, da nadaljujemo postopek, uredimo koeficiente v Routhovo shemo, kot prikazuje naslednji vzorec:

$$\begin{array}{rcccccc}
 s^n & a_0 & a_2 & a_4 & a_6 & \cdots \\
 s^{n-1} & a_1 & a_3 & a_5 & a_7 & \cdots \\
 s^{n-2} & b_1 & b_2 & b_3 & b_4 & \cdots \\
 s^{n-3} & c_1 & c_2 & c_3 & c_4 & \cdots \\
 s^{n-4} & d_1 & d_2 & d_3 & d_4 & \cdots \\
 \vdots & \vdots & \vdots & & & \\
 s^3 & e_1 & e_2 & e_3 & & \\
 s^2 & f_1 & f_2 & & & \\
 s^1 & g_1 & & & & \\
 s^0 & h_1 & & & &
 \end{array}$$

Koeficiente b_1 , b_2 in b_3 izračunamo s pomočjo enačb

$$\begin{aligned}
 b_1 &= \frac{a_1 a_2 - a_0 a_3}{a_1} \\
 b_2 &= \frac{a_1 a_4 - a_0 a_5}{a_1} \\
 b_3 &= \frac{a_1 a_6 - a_0 a_7}{a_1} \\
 &\vdots
 \end{aligned} \quad (8.81)$$

Koeficiente b_i računamo, dokler ne upoštevamo vseh a_i iz predhodnih vrstic. Enak postopek se uporablja za izračun ostalih koeficientov:

$$\begin{aligned}
 c_1 &= \frac{b_1 a_3 - a_1 b_2}{b_1} & d_1 &= \frac{c_1 b_2 - b_1 c_2}{c_1} \\
 c_2 &= \frac{b_1 a_5 - a_1 b_3}{b_1} & d_2 &= \frac{c_1 b_3 - b_1 c_3}{c_1} \\
 c_3 &= \frac{b_1 a_7 - a_1 b_4}{b_1} & & \vdots \\
 & \vdots & & \vdots
 \end{aligned} \tag{8.82}$$

Postopek nadaljujemo tako dolgo, da dobimo $n + 1$ vrstic. Tako dobimo Routhovo shemo trikotne oblike. Vse elemente v eni vrstici lahko zaradi enostavnejšega nadaljnega računanja množimo ali delimo s poljubnim pozitivnim številom.

Routhov stabilnostni kriterij pravi, da je število korenov karakteristične enačbe (8.80) s pozitivnimi realnimi deli enako številu menjav predznaka v prvi koloni trikotne sheme. Zato je potreben in zadosten pogoj za stabilnost sistema, da so vsi koeficienti karakterističnega polinoma pozitivni in da so vsi elementi v prvi koloni trikotne sheme pozitivni.

Primer 8.4 Karakteristična enačba sistema ima obliko

$$s^4 + 2s^3 + 3s^2 + 4s + 5 = 0 \tag{8.83}$$

Ker so vsi koeficienti pozitivni, je izpolnjen potreben pogoj za stabilnost sistema. Izračunana Routh-ova shema pa ima obliko

$$\begin{array}{cccc}
 s^4 & 1 & 3 & 5 \\
 s^3 & 2 & 4 & 0 \\
 & 1 & 2 & 0 \\
 s^2 & 1 & 5 & \\
 s^1 & -3 & & \\
 s^0 & 5 & &
 \end{array} \quad / \cdot \frac{1}{2}$$

Ker se predznak v prvi koloni dvakrat zamenja (upoštevati moramo prehod iz + v - in nato iz - v +), ni izpolnjen zadostni pogoj za stabilnost, sistem ima dva pola v desnem delu ravnine s . Primer tudi prikazuje, kako smo drugo vrstico zaradi lažjega nadaljnjega računanja pomnožili z $\frac{1}{2}$. \square

Posebni primeri

Če je element prve kolone katerekoli vrstice v Routhovi shemi enak nič, nadomestimo element s poljubno majhno konstanto ε in nadaljujemo postopek. Na koncu limitiramo ε proti 0.

Primer 8.5 Če imamo polinom

$$s^4 + 5s^3 + 7s^2 + 5s + 6 = 0 \quad (8.84)$$

je Routh-ova shema

$$\begin{array}{r} s^4 \\ s^3 \\ s^2 \\ s^1 \\ s^0 \end{array} \begin{array}{ccc} 1 & 7 & 6 \\ 5 & 5 & \\ 6 & 6 & \\ \lim_{\varepsilon \rightarrow 0} \varepsilon & & \\ 6 & & \end{array}$$

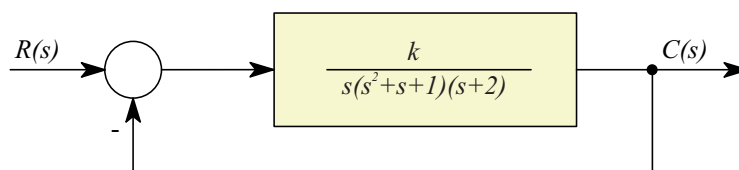
Če je $\varepsilon > 0$, ni spremembe predznaka v prvi koloni, sistem je stabilen. Če je $\varepsilon < 0$, imamo dve spremembi znaka, torej dva korena v desnem delu ravnine s . Ker pa je $\lim_{\varepsilon \rightarrow 0} \varepsilon = 0$, to pomeni, da sta dva korena na imaginarni osi (konjugirano kompleksni par). \square

Uporaba Routhovega stabilnostnega kriterija pri analizi regulacijskih sistemov

Slabost Routhovega stabilnostnega kriterija pri analizi regulacijskih sistemov je v tem, da daje le informacijo o absolutni stabilnosti, ne daje pa informacije o

relativni stabilnosti (koliko je sistem stabilen oz. nestabilen) in ne pove, kako sistem stabilizirati. Možno pa je proučevati vpliv enega ali dveh parametrov regulacijskega sistema, tako da s pomočjo kriterija ugotovimo tiste vrednosti parametrov, za katere je sistem stabilen.

Primer 8.6 Izračunajmo vrednosti ojačenja k , za katere je regulacijski sistem na sliki 8.22 stabilen.



Slika 8.22: Regulacijski sistem

Karakteristično enačbo zaprtozančnega sistema dobimo iz izraza

$$\frac{C(s)}{R(s)} = \frac{k}{s(s^2 + s + 1)(s + 2) + k} \quad (8.85)$$

v obliki

$$s^4 + 3s^3 + 3s^2 + 2s + k = 0 \quad (8.86)$$

Routh-ova shema pa je

s^4	1	3	k
s^3	3	2	0
s^2	$\frac{7}{3}$	k	
s^1	$2 - \frac{9}{7}k$		
s^0	k		

Da je regulacijski sistem stabilen, morajo biti vsi koeficienti karakterističnega polinoma pozitivni. Torej mora veljati

$$k > 0 \quad (8.87)$$

Pozitivni pa morajo biti tudi vsi elementi v prvi koloni Routh-ove sheme

$$2 - \frac{9}{7}k > 0 \implies k < \frac{14}{9} \quad (8.88)$$

Celotna rešitev za parameter k je

$$0 < k < \frac{14}{9} \quad (8.89)$$

Pri ojačenju $\frac{14}{9}$ sistem nedušeno niha.

□

Literatura

Aburdene, M.F. (1988), *Digital Continuous System Simulation*, Wm.C. Brown Publishers, Dubuque, Iowa, USA.

Bertalanffy, L.V. (1972), *The History and Starts of General System Theory v: Trends in General Systems Theory*, (G.J.Klir, Ed.), John Wiley & Sons, Inc., New York, USA.

Cellier, F.E. (1991), *Continuous System Modeling*, Springer - Verlag, New York, USA.

Cha P. D., Molinder J.I.(2006), *Fundamentals of Signals and Systems: A Building Block Approach*, Cambridge University Press, UK.

Dabney J.B., Harman T.L. (2004), *Mastering SIMULINK*, Prentice Hall, Upper Saddle River, N.J., USA.

Dorf, R.C, R.H. Bishop (2008): *Modern Control Systems*, Eleventh edition, Prentice Hall, USA

Dymola (2008), Multi-engineering modeling and simulation, Users manual, ver 7.0. Dessault System, Dynasim AB, Sweden, Lund.

Elmqvist, H. (1978), *A structured model language for large continuous systems*, Ph. D. diss. Rep. CODEN: LUTFD2/(TFRF-1015), Departement of Automatic Control, Lund Institute of Technology, Sweden.

Franklin, G.F., J.D. Powell, A.E. Naeini (2009): *Feedback Control of Dynamic Systems*, Six edition, Prentice Hall, USA

Karba, R. (1994), *Gradniki sistemov vodenja*, Fakulteta za elektrotehniko in računalništvo, Univerza v Ljubljani, SLO

Karba, R. (1994), *Modeliranje procesov*, Založba Fakultete za elektrotehniko in Fakultete za računalništvo in informatiko, Univerza v Ljubljani, SLO

Korn, G.A. and J.V. Wait (1978), *Digital Continuous System Simulation*, Prentice Hall, Englewood Clifs, N.J., USA.

Kuo, B.C., F. Golnaraghi (2010), *Automatic Control Systems*, Eighth edition, John Wiley & Sons, USA

- Lauber, R. (1989), *Prozessautomatisierung - 1. part.* Springer-Verlag, Berlin, Germany
- Matko, D., B. Zupančič, R. Karba (1992), *Simulation and Modelling of Continuous Systems - A Case Study Approach.* Prentice Hall.
- Mihelič, F. (2006), *Signali*, Založba Fakultete za elektrotehniko in Fakultete za računalništvo in informatiko, Univerza v Ljubljani, SLO
- Mlakar, J. (2002), *Linearna vezja in signali*, Založba Fakultete za elektrotehniko in Fakultete za računalništvo in informatiko, Univerza v Ljubljani, SLO
- Modelica (2007), A Unified Object-Oriented Language for Physical Systems Modeling, Language Specification, Version 3.0*, Modelica Association, <http://www.modelica.org/documents/ModelicaSpec30.pdf>
- Neelamkavil, F. (1987), *Computer Simulation and Modelling*, John Willey, New York, USA.
- Oblak, S., Škrjanc, I. (2008), *Matlab s Simulinkom, priročnik za laboratorijske vaje*, Univerza v Ljubljani, Založba FE in FRI, Slovenija.
- Ogata K. (2010), *Modern Control Engineering*, Fifth edition, Prentice Hall, USA.
- Polke M., Ed. (1994). *Process Control Engineering*, VCH Verlagsgesellschaft mbH, Weinheim.
- Schmidt, B. (1987), "What does simulation do? Simulation's place in the scientific method." *Syst. Anal. Model. Simul.* (Benelux journal), 4 (1987) 3, 193-211.
- Simulink, Dynamic System Simulation Software (2009), Users manual, R2009b, Math Works, Inc., Natick, MA, ZDA.
- Smith, D.J.M (1995), *Continuous System Simulation*, Chapman & Hall, London, UK.
- Sodja, A, B. Zupančič (2009), *Modelling thermal processes in buildings using an object-oriented approach and Modelica.* Simulation Modelling Practice and Theory, vol. 17, no. 6, str. 1143-1159.
- Strmčnik, S., R.Hanus, Đ. Juričić, R. Karba, Z. Marinšek, D.Murray-Smith, H. Verbruggen, B. Zupančič (1998): *Celostni pristop k računalniškemu vodenju procesov*, Univerza v Ljubljani, Fakulteta za elektrotehniko, SLO

Šega, P. (1988), *Regulacije in sistemi*, Fakulteta za elektrotehniko, Univerza v Ljubljani, SLO

Zupančič B. (1992), *SIMCOS- jezik za simulacijo zveznih in diskretnih dinamičnih sistemov*, Fakulteta za elektrotehniko in računalništvo, Univerza v Ljubljani, Slovenija.

Zupančič, B. (2010a): *Simulacija dinamičnih sistemov*, Založba Fakultete za elektrotehniko in Fakultete za računalništvo in informatiko, Univerza v Ljubljani, SLO

Zupančič, B. (2010b), *Zvezni regulacijski sistemi, I.del*, Založba Fakultete za elektrotehniko in Fakultete za računalništvo in informatiko, Univerza v Ljubljani, SLO

Zupančič, B. (2010c), *Zvezni regulacijski sistemi, II.del*, Založba Fakultete za elektrotehniko in Fakultete za računalništvo in informatiko, Univerza v Ljubljani, SLO