

Univerza
v Ljubljani Fakulteta
za elektrotehniko



AVTOMATSKO VODENJE SISTEMOV

*Založba
FE in FRI*

BORUT ZUPANČIČ

Kazalo

1	Uvod v vodenje sistemov	1
1.1	Kaj je vodenje?	1
1.2	Vrste vodenja	4
1.3	Učinki vodenja	5
1.4	Razvoj področja	9
1.5	Celostni pristop k računalniškemu vodenju	10
1.6	Struktura podjetja in vodenja	13
1.7	Struktura računalniškega sistema za vodenje v podjetju	15
1.8	Gradniki sistemov avtomatskega vodenja	18
1.8.1	Opazovanje	18
1.8.2	Ukrepanje	21
1.8.3	Odločanje	22
1.8.4	Naprednejši gradniki sistemov vodenja	32
1.9	Sistemi pristop pri načrtovanju vodenja (regulacije)	34
2	Sistemi in signali	39
2.1	Sistemi in sistemska teorija	39
2.2	Različni primeri sistemov	41

2.3	Procesi	43
2.4	Bistvo sistemske teorije v povezavi z modeliranjem	45
2.5	Signali	48
2.5.1	Osnovni pojmi	48
2.5.2	Vrste signalov	49
2.5.3	Moč in energija signala	50
2.5.4	Obdelava signalov	51
2.5.5	Temeljni signali	53
2.5.6	Uvod v spektralno analizo	61
3	Modeliranje procesov	65
3.1	Cilji in pomembnejše zakonitosti pri modeliranju	65
3.2	Vrste modelov	67
3.3	Načini modeliranja	73
3.3.1	Teoretično modeliranje	74
3.3.2	Eksperimentalno modeliranje	76
3.3.3	Kombinirano modeliranje	77
3.4	Modeliranje in simulacija kot enovit ciklični postopek	78
3.5	Primeri	81
4	Zapisi matematičnih modelov	95
4.1	Zapis z diferencialno enačbo	95

4.2	Laplace-ova transformacija	96
4.3	Reševanje diferencialne enačbe z Laplace-ovo transformacijo . . .	99
4.4	Prenosne funkcije	102
4.5	Bločni diagrami	108
4.5.1	Splošne značilnosti bločnih diagramov	108
4.5.2	Postopek za risanje bločnih diagramov	112
4.5.3	Poenostavljanje bločnih diagramov	116
5	Računalniško podprta analiza in simulacija sistemov	121
5.1	Vnos linearnih dinamičnih sistemov	121
5.2	Osnovne zmožnosti dela s sistemi	123
5.3	Simulacija s pomočjo Matlabovih funkcij	124
6	Osnovne metode pri reševanju problemov s simulacijo	131
6.1	Simulacijska shema	131
6.2	Indirektna metoda	133
6.3	Simulacija prenosnih funkcij	139
6.4	Princip modularnosti	141
7	Orodja za simulacijo dinamičnih sistemov	143
7.1	Osnovne lastnosti simulacijskih sistemov	143
7.2	Simulacijsko okolje Matlab-Simulink	144

7.3	Primeri v okolju Matlab - Simulink	159
8	Analiza sistemov v časovnem prostoru	167
8.1	Osnovne lastnosti časovnega odziva	167
8.2	Vpliv polov in ničel na časovni odziv	169
8.3	Razvrstitev sistemov	172
8.4	Proporcionalni sistemi	175
8.4.1	Sistem prvega reda	175
8.4.2	Sistem drugega reda	180
8.5	Integrirni sistemi	189
8.6	Diferencirni sistemi	190
8.7	Sistemi z mrtvim časom	192
8.8	Stabilnost sistemov	193
8.8.1	Pogoj za stabilnost linearnih, časovno nespremenljivih sistemov	193
8.8.2	Routhov stabilnostni kriterij	194
9	Vodenje sistemov	201
9.1	Odprtozančni sistem vodenja	203
9.2	Zaprtozančni sistem vodenja	204
9.3	Sledilno delovanje regulacijskega sistema	205
9.3.1	Pokazatelji kvalitete pri sledilni regulaciji	205

9.4	Regulacijsko delovanje regulacijskega sistema	208
9.5	Obravnava regulacijskega sistema v delovni točki	210
9.6	Ustaljeno stanje regulacijskih sistemov	212
9.7	Stabilnost regulacijskih sistemov	217
9.8	Nekateri preostali učinki povratne zanke	218
9.9	Nekateri dodatni pojmi, ki jih srečujemo v regulaciji	220
9.10	Primeri sistemov vodenja	221
10	Industrijski regulacijski algoritmi	227
10.1	Razdelitev industrijskih regulatorjev	229
10.2	Zvezno delujoči regulatorji	234
10.2.1	Proporcionalni regulatorji	234
10.2.2	Integrirni regulatorji	242
10.2.3	Proporcionalno – integrirni regulatorji	244
10.2.4	Regulatorji z diferencirnim dodatkom	247
10.2.5	Modificirani PID regulatorji	253
10.3	Načrtovanje zveznih PID regulatorjev	254
10.3.1	Analitični postopki	256
10.3.2	Nastavitvena pravila	258
10.3.3	Optimizacija s pomočjo integralskih cenilk	265
10.4	Izvedbe PID regulatorjev	270

Literatura**277**

1.

Uvod v vodenje sistemov

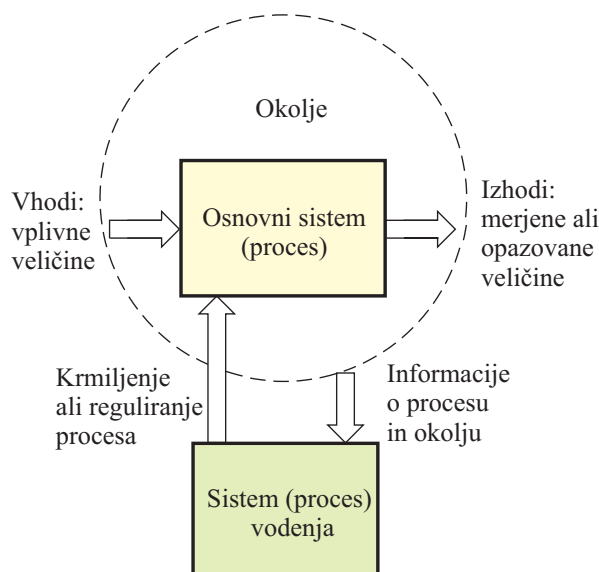
1.1 Kaj je vodenje?

V vsakdanjem življenju nas obdajajo številna realna dogajanja. Le ta imajo veliko skupnega, zato se je razvilo pomembno področje sistemske teorije, ki ta dogajanja obravnava enovito in sistematično. Sistemska teorija uči, da nas obdajajo sistemi, dogajanja znotraj sistemov pa se imenujejo procesi. Primer: znotraj sistema generator se odvija proces proizvodnje električne energije. Ena skupnih lastnosti sistemov oziroma dogajanj je ta, da morajo biti želeni ali čim bolj optimalni. In ker vsaj v začetku delovanja dogajanja niso optimalna, potrebujemo sistem vodenja. Sistem vodenja torej potrebujemo, ker realno dogajanje nekega sistema odstopa od želenega. Lahko postavimo naslednjo definicijo:

Vodenje je proces, s katerim vplivamo na delovanje (obnašanje) sistema z namenom, da dosežemo zastavljen cilj oz. želeno delovanje.

Tako kot smo realna dogajanja sistematizirali v sisteme in procese, tudi vodenje izvedemo s sistemom (npr. računalniškim), znotraj katerega potekajo procesi (npr. algoritmi vodenja). Vendar pa se v vsakdanjem izražanju besedi sistem in proces redko dosledno uporabljata.

Vodene sisteme sistematično obravnavamo tako, da skušamo, če je le možno, razdvojiti mehanizme delovanja realnega sistema in sistema vodenja, kot prikazuje slika 1.1. Osnovni sistem se vedno nahaja v okolju. Okolje deluje na sistem, sistem deluje na okolje. Govorimo o vplivnih ali vhodnih veličinah in o merjenih

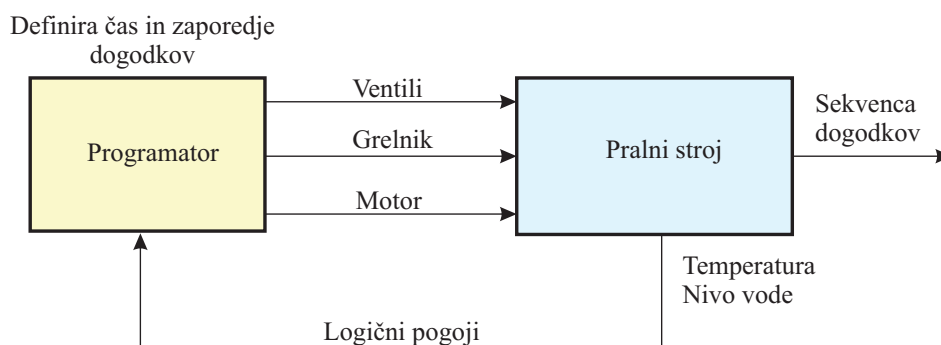


Slika 1.1: Razdvojitve vodenega sistema na osnovni sistem in sistem vodenja

ali izhodnih veličinah. Sistem vodenja dobiva informacije o sistemu in okolju in iz teh informacij določa krmilne ali regulirne signale (npr. meri temperaturo v prostoru in določa moč grelnika).

Primer 1.1 Vodenje pralnega stroja

Vzemimo enostavni pralni stroj. V njem poteka proces, ki iz umazanega naredi čisto perilo. Sistem vodenja pralnega stroja mora zagotoviti, da bo perilo določene vrste v določenem času oprano tako, da bo uporabnik zadovoljen. Bistvo vodenja procesa pranja perila je, da proces že vnaprej razdelimo na določene faze (polnjenje z vodo, segrevanje, dodajanje praška, vrtenje bobna, praznjenje vode, itd.) ter že vnaprej določimo, kakšno naj bo zaporedje faz in koliko časa naj določena faza traja, da bo perilo dobro oprano. To rešitev vgradimo v programator, ki potem ustrezno odpira in zapira ventile, vklaplja in izklaplja grelnik ter vklaplja in izklaplja motor in to vedno na enak način, ne glede na to, kaj se v stroju v resnici dogaja. Seveda pa je dejanski pralni stroj nekoliko bolj zapleten, saj je potrebno upoštevati, da je količina perila, ki ga damo v pranje enkrat lahko velika, drugič majhna. Temu ustrezno je treba prilagajati količino vode, količini vode pa čas segrevanja. Da ta problem rešimo, določenih faz ne pričnemo izvajati, dokler ni dosežen določen nivo vode v bobnu, ali dokler ni dosežena določena temperatura, itd. (glej sliko 1.2).



Slika 1.2: Vodenje pralnega stroja

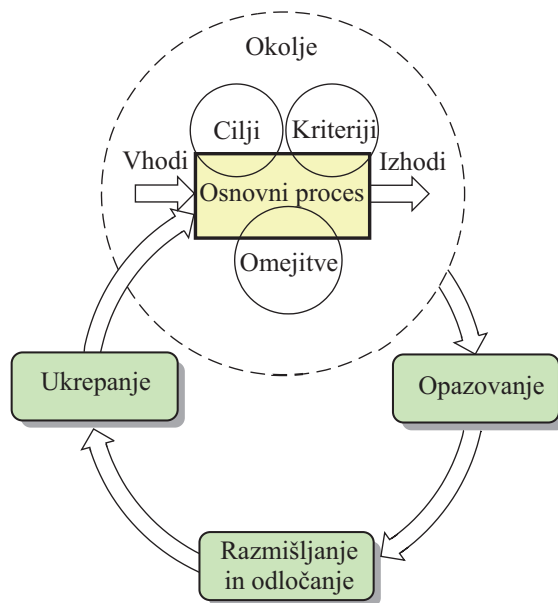
Vidimo, da smo za vodenje pralnega stroja uporabili informacijo o želenem cilju (čisto perilo), informacijo o poti kako priti do tega cilja (do katere smo prišli že prej npr. s preizkušanjem) in informacijo o logičnih pogojih (nivo vode, temperatura), ki so pogoj za nadaljevanje izvajanja predpisanih faz. □

Če torej zadevo nekoliko posplošimo potem lahko zaključimo naslednje: Eden od bistvenih delov procesa vodenja je pridobivanje različnih tipov informacij, kot so:

- informacija o **ciljih**, želenih vrednostih,
- informacija o **kriterijih**, ki opredeljuje načine doseganja ciljev (natančnost, hitrost, cena, varnost, itd.),
- informacija o sistemu, ki ga vodimo oz. o vplivu **vhodov** (kako reagira na spremembe vhodnih-krmilnih spremenljivk in kako nanj vpliva okolica),
- informacija o **izhodih** sistema (kaj se s sistemom trenutno dogaja),
- informacija o **omejitvah** (ki izhajajo iz sistema in/ali okolice).

Ta segment aktivnosti lahko imenujemo **opazovanje** procesa in njegovega okolja v najširšem smislu. S to aktivnostjo dobimo ustrezno bazo znanja, s pomočjo katere pripravimo ukrepe, ki naj bi sistem pripeljali proti želenemu cilju. Uvedli smo torej naslednjo aktivnost- **razmišljanje in odločanje**. Priprava in izbira ukrepov je seveda srce vodenja. Kvaliteta vodenja je nedvomno bistveno odvisna od kvalitete ukrepa pa tudi od njegove pravočasnosti. Oboje pa zavisi od uporabljenega principa in metode vodenja. Da bi odločitve lahko v resnici izpeljali, potrebujemo še tretjo komponento v sistemu vodenja - **ukrepanje**. Na voljo

moramo imeti načine in sredstva, ki vplivajo na procese v sistemu, ki ga vodimo, tako da se ti procesi spreminjajo. Še tako dobra in pravočasna odločitev namreč nima smisla, če je ne moremo izpeljati. Če sedaj vse tri našteje elemente upoštevamo, dobimo razširjeno strukturo procesa vodenja, ki ga prikazuje slika 1.3.



Slika 1.3: Univerzalni mehanizem vodenja

1.2 Vrste vodenja

O **ročnem vodenju** govorimo, če se opazovanje, odločanje in ukrepanje (vse faze ali le nekatere) vršijo pod vplivom človeka.

Če te faze izvaja tehniški sistem, potem govorimo o **avtomatskem vodenju**. Avtomatsko vodenje je temeljna vsebina področja *Avtomatika*, vede, ki obravnava analizo in načrtovanje samodejno delujočih naprav oz. sistemov.

Če faze izvaja računalniški sistem, govorimo o **računalniškem vodenju**.

Danes seveda prevladujejo izvedbe s pomočjo računalniških sistemov. Precej so v rabi tudi elektonske izvedbe, mehanske, pnevmatske in hidravlične izvedbe

pa le izjemoma (npr. pnevmatski regulatorji v petrokemijski industriji zaradi varnostnih zahtev).

1.3 Učinki vodenja

Avtomatsko vodenje tehnološko vse bolj zahtevnih in zapletenih proizvodnih procesov omogoča

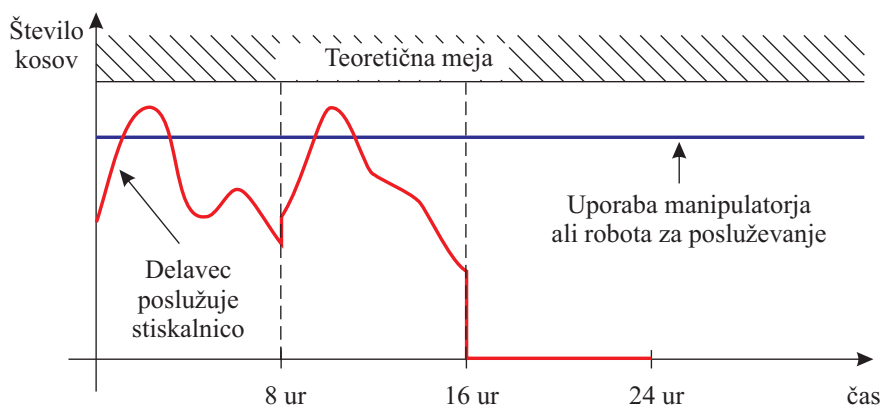
- večjo produktivnost,
- boljšo kvaliteto izdelkov in večjo ponovljivost proizvodnje,
- varčnejšo porabo energije,
- ekološke izboljšave,
- varčno rabo surovin in nenazadnje tudi
- sociološke učinke.

Uporaba sodobnih metod vodenja dinamičnih sistemov je nepogrešljiva v domala vseh proizvodnih procesih in predstavlja eno izmed osnov uspešne proizvodnje z veliko dodano vrednostjo.

Primer 1.2 Ilustracija nekaterih najpomembnejših učinkov sistemov za avtomatsko vodenje

Večja produktivnost

Večjo produktivnost je mogoče doseči z boljšo izkoriščenostjo obstoječih proizvodnih kapacitet. Te kapacitete je mogoče bolje izkoristiti, če za planiranje in razporejanje dela ali pa za posluževanje posameznih strojev uporabljamo računalniško podprte sisteme za vodenje. Na sliki 1.4 je podan primer obratovanja stiskalnice. Če jo poslužuje človek, je njena izkoriščenost v povprečju manjša od tiste, ki jo doseže računalniško vodeni sistem za posluževanje (robot ali manipulator).

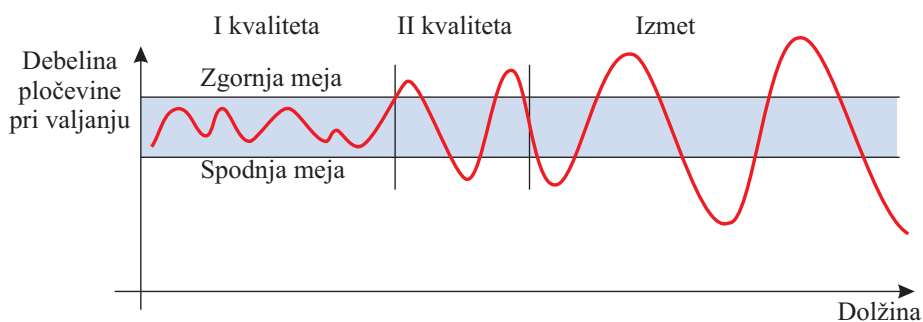


Slika 1.4: Primer izkoriščenosti stiskalnice

S prerazporeditvijo delavca na manj nevarno in manj utrujajoče delovno mesto bomo po eni strani povečali produkcijo, po drugi strani pa povečali varnost pri delu.

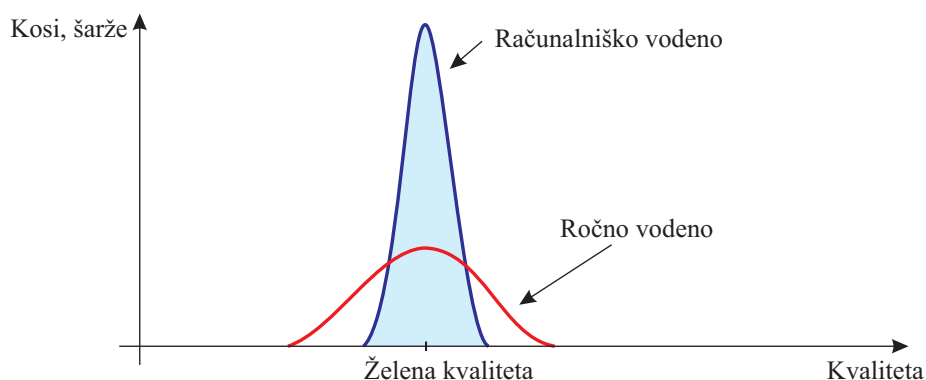
Izboljšanje kvalitete in boljša ponovljivost

Boljšo kvaliteto lahko dosežemo z večjo natančnostjo izdelave določenih proizvodov. To pomeni, da se bolj držimo predpisanih toleranc. Na sliki 1.5 vidimo primer valjanja pločevine. Z uporabo dobrega sistema za vodenje bo nihanje debeline po dolžini oziroma času (čas je zaradi konstantne hitrosti sorazmeren z dolžino) manjše. Zato bo pločevine II. kvalitete ali celo tiste za izmet znatno manj, ali pa je sploh ne bo.



Slika 1.5: Primer valjanja pločevine

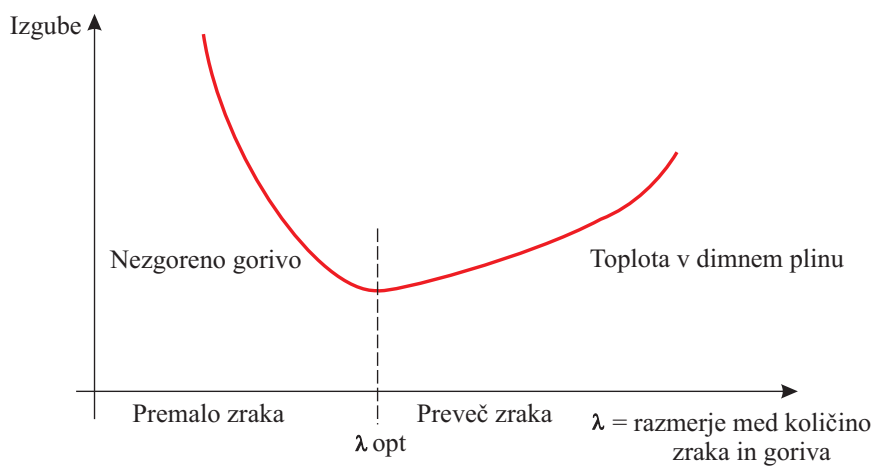
Večjo kvaliteto dosežemo tudi z boljšo ponovljivostjo, kar je posebej važno pri kosovni oziroma šaržni proizvodnji. V splošnem velja, da je stresanje po kvaliteti pri računalniško vodeni proizvodnji manjše kot pri ročno vodeni (slika 1.6).



Slika 1.6: Primer ponovljivosti v kosovni ali šaržni proizvodnji

Varčnejša poraba energije

Z računalniško podprtim sistemom za vodenje lahko pri kotlih in industrijskih pečeh stalno iščemo optimalno točko zgorevanja, to pomeni optimalno razmerje med dovedenim gorivom in zrakom. Če je namreč zraka preveč, izgubljammo toploto z dimnimi plini, če ga je premalo, pa prihaja do nepopolnega zgorevanja. Ilustracijo tega primera lahko vidimo na sliki 1.7.

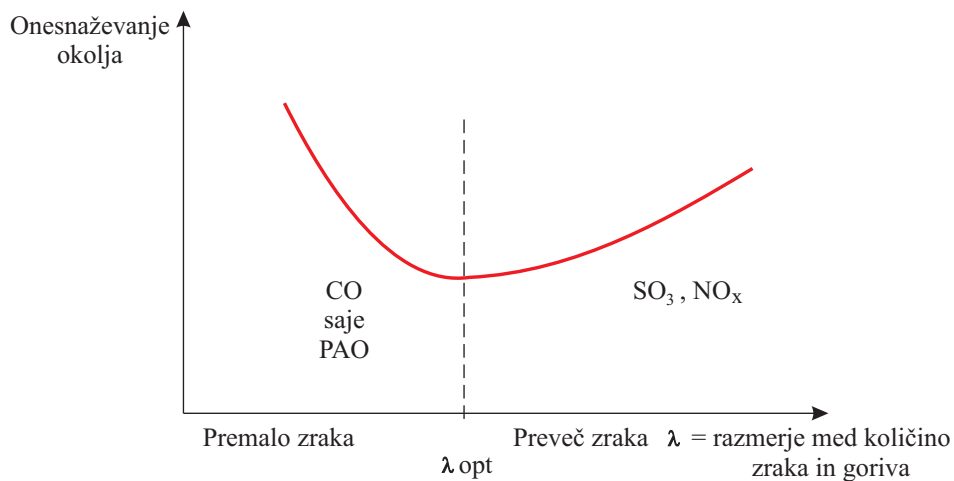


Slika 1.7: Zmanjšanje izgub pri optimalnem zgorevanju

Na ta način lahko prihranimo nekaj odstotkov goriva, kar je dolgoročno gledano lahko zelo pomembno.

Ekološke izboljšave

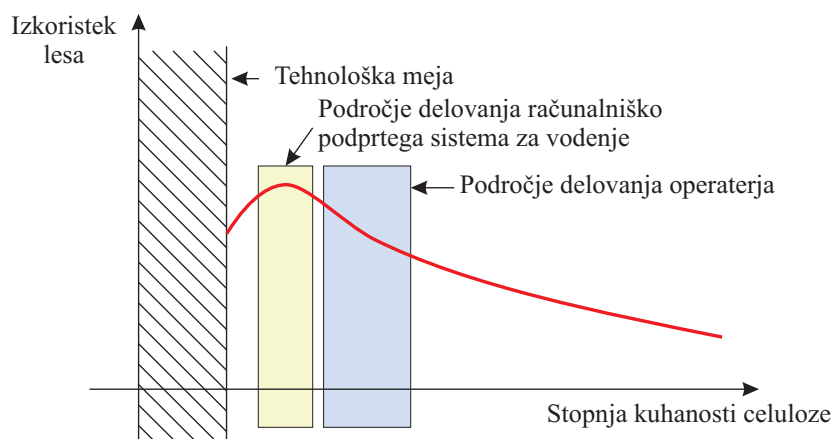
Pri regulaciji zgorevanja v kotlih in industrijskih pečeh dosežemo pri razmerju goriva z zrakom, ki je optimalno s stališča minimizacije izgub tudi minimalno onesnaževanje okolja. Če je zraka preveč, nastajajo pri zgorevanju žveplov trioksid in dušikovi oksidi, če ga je premalo, pa pride do tvorbe ogljikovega monoksida, saj in poliaromatskih ogljikovodikov. Primer ilustrira slika 1.8.



Slika 1.8: Zmanjšanje onesnaževanja okolja pri optimalnem zgorevanju

Varčna raba surovin

Manjšo porabo surovin lahko dosežemo s povečanjem izkoristkov pri njihovi predelavi. Na sliki 1.9 je podan diagram, ki kaže, kako je z računalniško podprtim vodenjem mogoče bolje izkoristiti les pri pridelavi celuloze.



Slika 1.9: Primer optimiranja izkoristka pri kuhanju celuloze

Krivulja na sliki podaja odvisnost izkoristka lesa od stopnje kuhanosti celuloze. Če je celuloza premalo kuhana, potem je v suspenziji, ki jo dobimo, še polno ostankov lesa in grč. To poleg slabega izkoristka povzroči zamašitev filtrov in resne zastoje v proizvodnji. Če pa je kuhana preveč, pa je izkoristek zopet slab, ker se tudi vlakna celuloze razkuhajo. Človek (operater) zaradi varnosti teži v desno od optimuma. Z računalniško podprtim vodenjem pa je mogoče sproti iskati optimalno točko. □

1.4 Razvoj področja

Vodenje tehniških sistemov je staro toliko, kot so stari dinamični tehniški sistemi. Vlogo sistema za vodenje je v začetku seveda opravljal človek (npr. krmar na ladji), mehanizmi, ki so zagotavljali avtomatsko vodenje, pa so se začeli pojavljati kasneje. Čeprav se že v antičnih časih srečamo z napravami, ki so bile sposobne regulirati pretok ali nivo tekočin, veljata za pomembna mejnika v razvoju področja avtomatski sistem za regulacijo temperature v valilnici kokošjih jajc, ki jo je načrtoval Drebbel okrog leta 1620 ter Wattov centrifugalni regulator za hitrost parnega stroja leta 1788.

Avtomatizacija proizvodnih procesov v pravem pomenu te besede se je začela intenzivneje razvijati v tridesetih letih prejšnjega stoletja, zelo važno prelomnico v razvoju konceptov in uporabe avtomatskih sistemov za vodenje pa je predstavljala druga svetovna vojna. Po vojni se po eni strani srečamo z razcvetom teorije vodenja (predvsem v 60. in 70. letih), po drugi strani pa, ob pospešenem razvoju elektronike in računalniške tehnologije, z zelo široko uporabo sistemov za vodenje na najrazličnejših področjih človekovega delovanja.

Računalniško vodenje procesov je zato zelo mlada disciplina. Predstavlja povezavo med disciplino vodenja (tehniških) sistemov in področjem računalništva oziroma informacijske tehnologije. Za začetek računalniškega vodenja procesov lahko štejemo uporabo procesnega računalnika v petrokemični tovarni Texaco v Port Arthurju (ZDA) v letu 1959. V šestdesetih letih se je potem razvil koncept direktnega digitalnega vodenja (DDC), ki je temeljil na zamenjavi analognih regulatorjev s centralnim digitalnim računalnikom. Pravi razcvet pa je računalniško vodenje procesov doživelo v sedemdesetih letih s pojavom mikroprocesorjev in mikroračunalnikov, katerih uporaba je omogočila skokovito povečanje funkcionalnosti in učinkovitosti. Razen izjemne vloge pri vodenju vesoljskih ladij, izstrelkov v vojni industriji, pri izvedbi avtopilotskih sistemov v letalski industriji, pri vo-

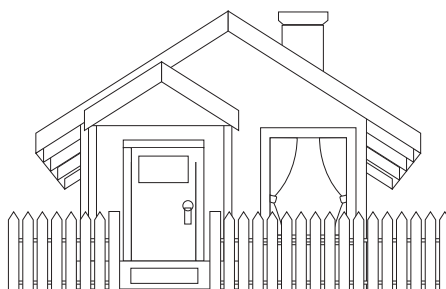
denju robotskih in mobilnih sistemov, je vodenje oz. avtomatska regulacija kot nekoliko ožje področje vodenja sistemov postala ključni in integralni del sodobnih industrijskih procesov. To velja predvsem v procesni industriji za regulacijo nivoja, tlaka, temperature, vlažnosti, viskoznosti, pretoka, PH itd. Vodenje je postalo ključno za učinkovito rabo energije, v sodobnih zgradbah, itd. Uporabnost pa se je razširila tudi na druga področja kot npr. na ekonomijo, biologijo, biomedicinske sisteme, na urbanizacijo in ekologijo.

1.5 Celostni pristop k računalniškemu vodenju

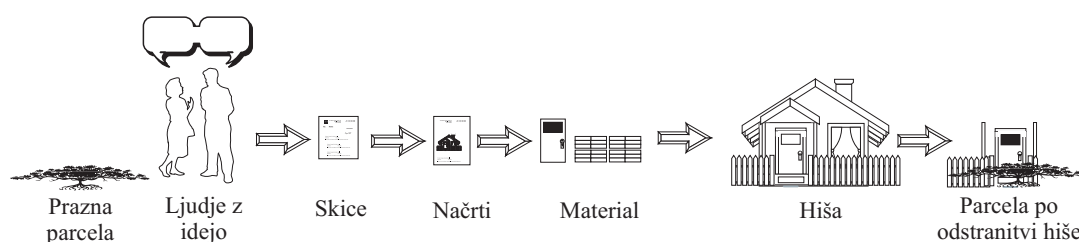
Prav kompleksnost računalniških sistemov za vodenje, ki se kaže v prepletanju procesa in njegove tehnologije s teorijo in tehnologijo vodenja, računalniško tehnologijo, tehnologijo razvoja programske opreme, vlogo in mestom človeka, itd., je osnovni problem, s katerim se na tem področju srečujemo. Ljudje smo seveda navajeni reševati zelo različne in tudi kompleksne probleme. Vprašanje pa je, kako povečati učinkovitost tega reševanja. Izkušnje iz številnih področij kažejo, da se učinkovitost reševanja in kvaliteta rešitev kompleksnih problemov izboljšata s sistematičnim in celostnim pristopom. Parcialni načini reševanja ne prinesejo rezultatov. Zato bomo na tem mestu predstavili strukturiran celostni pristop k izgradnji in uporabi računalniških sistemov za vodenje, ki upošteva najpomembnejše dimenzije problema. Zaradi večje jasnosti bomo za predstavitev različnih dimenzij uporabili analogijo z gradbeništvom, ki ima mnogo daljšo tradicijo, in kjer so rezultati bolj oprijemljivi in vsakomur poznani.

Problem izgradnje hiše lahko gledamo vsaj skozi tri dimenzije (glej tudi sliko 1.13). Prva dimenzija je hiša kot objekt oziroma **proizvod** (slika 1.10). V tem okviru je pomembno, kakšen je namen hiše, kakšna je funkcija posameznih njenih delov, kako hiša kot objekt in posamezni segmenti v hiši delujejo ter kakšna je oblika in struktura hiše.

Druga pomembna dimenzija je dinamični proces nastajanja, uporabe in odstranjevanja te hiše. Ta proces imenujemo tudi življenjski cikel hiše. Pri tem je pomembno, da se hiša v fazah svojega življenjskega cikla pojavlja v različnih pojavnih oblikah (slika 1.11): od zamisli v glavah npr. moža in žene, preko zapisa želja na papirju, skice, ki jo naredita skupaj z arhitektom, podrobnega načrta, ki ga naredi gradbeni inženir, pa do posameznih faz gradnje, njenega dokončanja, uporabe hiše in njenega vzdrževanja, ter njenega porušjenja in odstranitve, ko je dotrajala in mora dati prostor novim in sodobnejšim zgradbam.



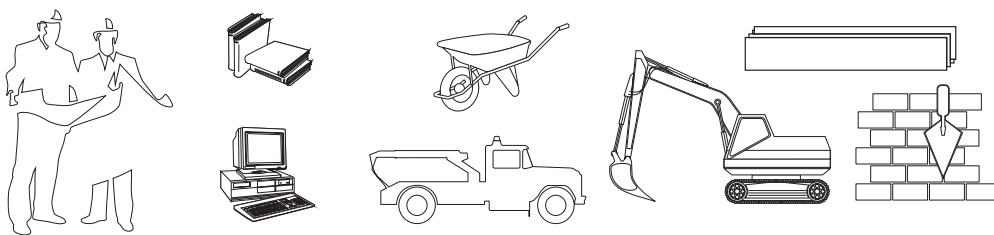
Slika 1.10: Hiša kot objekt



Slika 1.11: Različne pojavne oblike hiše

Dimenzijo **življenjskega cikla** torej sestavljajo posamezne faze. Najbolj običajne faze so definicija zahtev, opredelitev specifikacij, načrtovanje, izvedba (izgradnja), zagon in uvažanje, obratovanje (delovanje) in upokožitev.

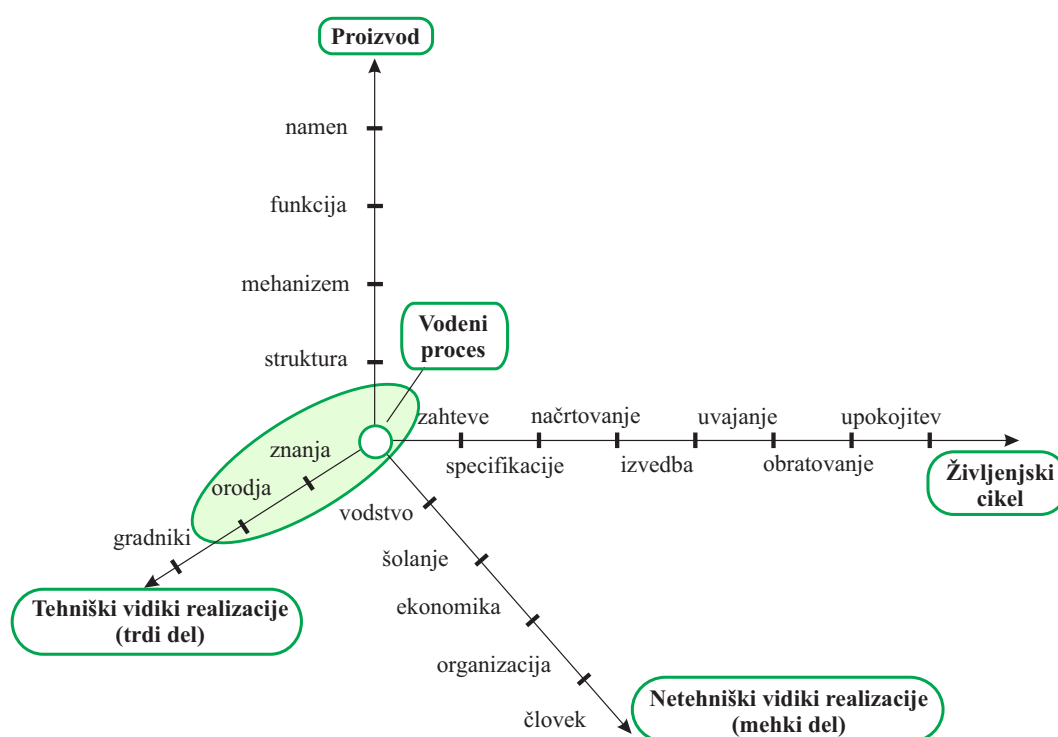
Tretjo dimenzijo predstavlja **tehnologija** v najširšem smislu besede. To je tehnologija, ki omogoča, da iz zamisli v glavah pridemo do načrta hiše in tehnologija, ki omogoči, da iz surovin in gradnikov zgradimo načrtovano hišo. Gre torej za skupek znanj (postopkov), orodij in gradnikov (slika 1.12), ki omogočajo transformacijo pojavne oblike hiše iz ene v drugo fazo življenjskega cikla.



Slika 1.12: Znanja, orodja in gradniki za gradnjo hiše

Tak večdimenzionalni (celostni) pogled (slika 1.13) na izgradnjo sistemov je seveda dovolj splošen, da ga lahko uporabimo tudi za razlago načrtovanja, izgrad-

nje in uporabljanja sistemov za vodenje, še zlasti, če dodamo še četrto dimenzijo, ki poudarja **netehniške vidike** izvedbe sistemov za vodenje. Le-ti imajo pomembno vlogo v vseh projektih vodenja. Ta dimenzija upošteva naslednje vidike: predanost vodstva (strategija, vizija), šolanje in izobraževanje, ekonomika, organizacija ter človeški, kulturni in družbeni vidik. Ob upoštevanju te dodatne dimenzije, ki seveda zopet ni posebnost samo za sisteme vodenja, dobimo štiridimenzionalno strukturo, ki je prikazana na sliki 1.13. Vidimo, da smo tehnologijo razdelili na tehniške vidike izvedbe (trdi del) in na netehniške vidike izvedbe (mehki del).



Slika 1.13: Štiridimenzionalni pogled na načrtovanje, izgradnjo in uporabo sistemov za vodenje

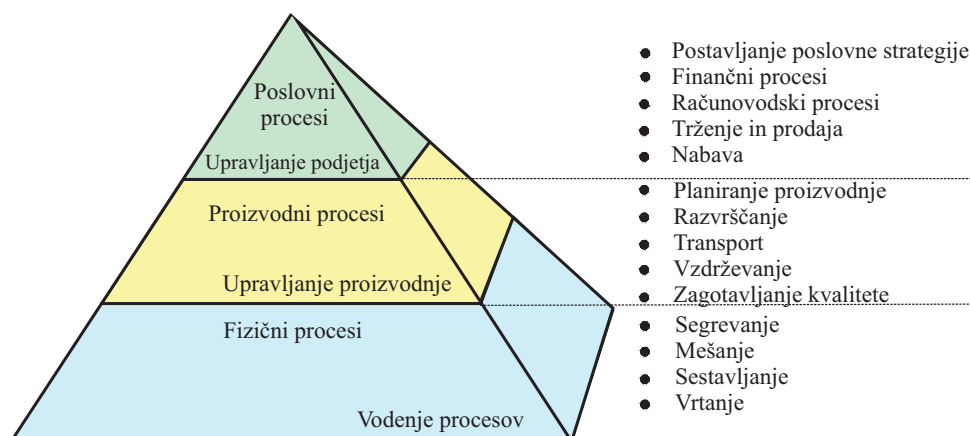
Tako dobljeno štiridimenzionalno strukturo je potrebno še nekoliko dopolniti. Prva dopolnitev je posebnost sistemov vodenja. O sistemih vodenja namreč ne moremo govoriti brez objekta, ali bolje rečeno **procesa**, ki ga vodimo. Vodenje procesa je namreč problem, ki ga rešujemo. Proces, ki ga vodimo, je zato izhodišče, okrog katerega se vrtili vse ostalo. Zato je smiselno, da ga v naši shemi postavimo v koordinatno izhodišče.

Na tem mestu smo torej opisali celostni pristop k računalniškemu vodenju. V

nadaljevanju se bomo bolj posvetili trem temam iz tega celovitega pristopa: **procesu** predvsem v smislu razumevanja vplivov, ciljev, modeliranja, **metodam** (znanjem) za modeliranje, simulacijo in vodenje in ustreznim računalniškim **orodjem**.

1.6 Struktura podjetja in vodenja

Podjetje lahko definiramo kot kompleksno povezavo različnih procesov. Zato je primerno, da na njem predstavimo splošno strukturo sistema vodenja. Različne vrste procesov strukturiramo v različne nivoje. Eno od pogosto uporabljenih struktur imenujemo *fizični model podjetja* ali tudi *piramida podjetja* in jo prikazuje slika 1.14.



Slika 1.14: Piramida podjetja

Ta struktura odraža predvsem razliko med tipi procesov in temu ustrezno tudi tipi izvajalcev teh procesov. Zato govorimo tudi o disciplinarno orientirani hierarhiji oz. o delitvi na poslovne, proizvodne in fizične procese. Slika 1.14 tudi konkretizira možne procese na posameznih nivojih podjetja.

Vsi ti procesi seveda zahtevajo svoje vodenje, ki je prilagojeno specifični posameznega procesa. Smiselno je, da struktura funkcij vodenja ustreza strukturi procesov, ki jih vodimo. Zato je tudi za vsak nivo vodenja na voljo obsežna in specifična materialna in programska oprema. Vse večji pa je tudi poudarek na povezovanju in prepletanju funkcij vodenja posameznih nivojev. Konkreten nabor funkcij in njihova razporeditev je seveda mogoča samo v povezavi s konkretnim podjetjem

oz. procesom, za katerega razvijamo sistem vodenja. Klasični hierarhični zgradbi podjetja lahko priredimo ustrezne hierarhične nivoje vodenja. Na ta način dobimo znano *CIM* (Computer Integrated Manufacturing) ali *CIP* (Computer Integrated Processing) piramido, ki je razdeljena na različno število nivojev. V skladu s fizičnim modelom podjetja, ki ga prikazuje slika 1.14, smo piramido vodenja razdelili na tri nivoje (Polke, 1994):

Vodenje procesov (Process control)

Na najnižjem nivoju, ki zahteva pretežno tehnična znanja, je vodenje procesov. Predvsem temu nivoju je posvečeno to delo pa tudi večina del, ki so povezana z avtomatiko.

Na tem nivoju vodenja prevladujejo funkcije, ki so neposredno vezane na fizična postrojenja, naprave in stroje. Po tipu vodenja jih lahko razdelimo v več skupin (ANSI/ISA-S88.01, 1995):

- **Varnostne blokade (Safety Interlocks)**

Varnostne blokade predstavljajo najpomembnejši sklop funkcij na procesnem nivoju. Gre za preprečevanje akcij, ki bi lahko ogrozile človeka, okolje ali opremo. Ta del funkcij mora biti po definiciji ločen od ostalih funkcij vodenja in je hierarhično na najnižjem nivoju vodenja. To pomeni, da se nobena druga aktivnost vodenja ne more vmešati med varnostno blokado in elektronsko opremo na napravah, ki izvajajo njene ukaze.

- **Osnovno vodenje (Basic Control)**

Osnovno vodenje je namenjeno vzpostavljanju in vzdrževanju določenega stanja procesa in naprav. Vsebuje krmiljenje ali regulacijo posameznih spremenljivk, diskretno/logično vodenje (npr. izbira med vnaprej določenimi stanji odprtosti ventila - ODPRT/ZAPRT), vodenje ponavljajočih se sekvenc (npr. zagon črpalke), procesne blokade (preprečevanje akcij, ki bi lahko ogrozile kvaliteto proizvoda), obravnava izrednih dogodkov (exception handling, off-normal handling) (npr. okvara določene opreme), obravnava posegov operaterja, itd.

- **Vodenje postopkov (Procedure Control)**

Ta funkcija zagotavlja, da z določenim zaporedjem akcij, ki so vezane na določeno opremo, izpeljemo nek postopek. Pri kontinuirnih procesih so taki postopki npr. zagon ali zaustavitev procesa, pri šaržnih procesih pa je to izvedba posamezne šarže. Vodenje postopkov je zato ključna funkcija za šaržne procese.

Upravljanje proizvodnje (Production Management)

Najpogostejše funkcije na nivoju upravljanja proizvodnje so tiste, ki se ukvarjajo z obvladovanjem logističnih procesov. Gre predvsem za funkcije planiranja in razvrščanja. Planiranje proizvodnje mora odgovoriti na vprašanja, kot so: kaj se bo proizvajalo, koliko se bo proizvajalo, kje se bo proizvajalo, kdaj mora biti narejeno, itd. Razvrščanje proizvodnje pa tipično vključuje opredelitve: v kako velikih šaržah oz. številu proizvodov se bo proizvajalo, na katerih postrojenjih, napravah in strojih se bo delalo, kdaj se bo delalo, kakšne so omejitve, itd. (Sawyer, 1993)). V kemični in procesni industriji je za ta nivo tipična funkcija upravljanja z recepti (kjer gre za opredeljevanje, shranjevanje in vzdrževanje (spreminjanje) receptov) ter upravljanje s šaržami (kjer gre za spremljanje in nadzor šarž v skladu z urnikom opredeljenim v razvrščanju). Poleg teh osrednjih funkcij srečamo na tem nivoju še funkcije kot so npr.: optimiranje proizvodnje, optimiranje kapacitet in pa funkcije, ki zagotavljajo delovanje spremljevalnih procesov kot so npr. vodenje skladišč, upravljanje z zalogami, vodenje transporta, vodenje vzdrževanja, zagotavljanje kvalitete in podobno.

Upravljanje podjetja (Corporate Management)

Upravljanje podjetja zajema informacijsko podporo in podporo pri odločanju pri ključnih poslovnih procesih, kot so: postavljanje poslovne strategije, definiranje ciljev posameznim enotam v podjetju, investicijska politika, raziskovalno-razvojna politika, finančna politika, kadrovska politika, tržna in prodajna politika, itd. Vsi navedeni procesi tvorijo okvir za proizvodno politiko. Bistvo proizvodne politike je prilagajanje proizvodnje tržnim razmeram, možnostim nabave surovin, polproizvodov in opreme ter usklajevanje z notranjimi (materialnimi in kadrovskimi) omejitvami. Osrednji del proizvodne politike je zato tudi dolgoročno planiranje proizvodnje.

1.7 Struktura računalniškega sistema za vodenje v podjetju

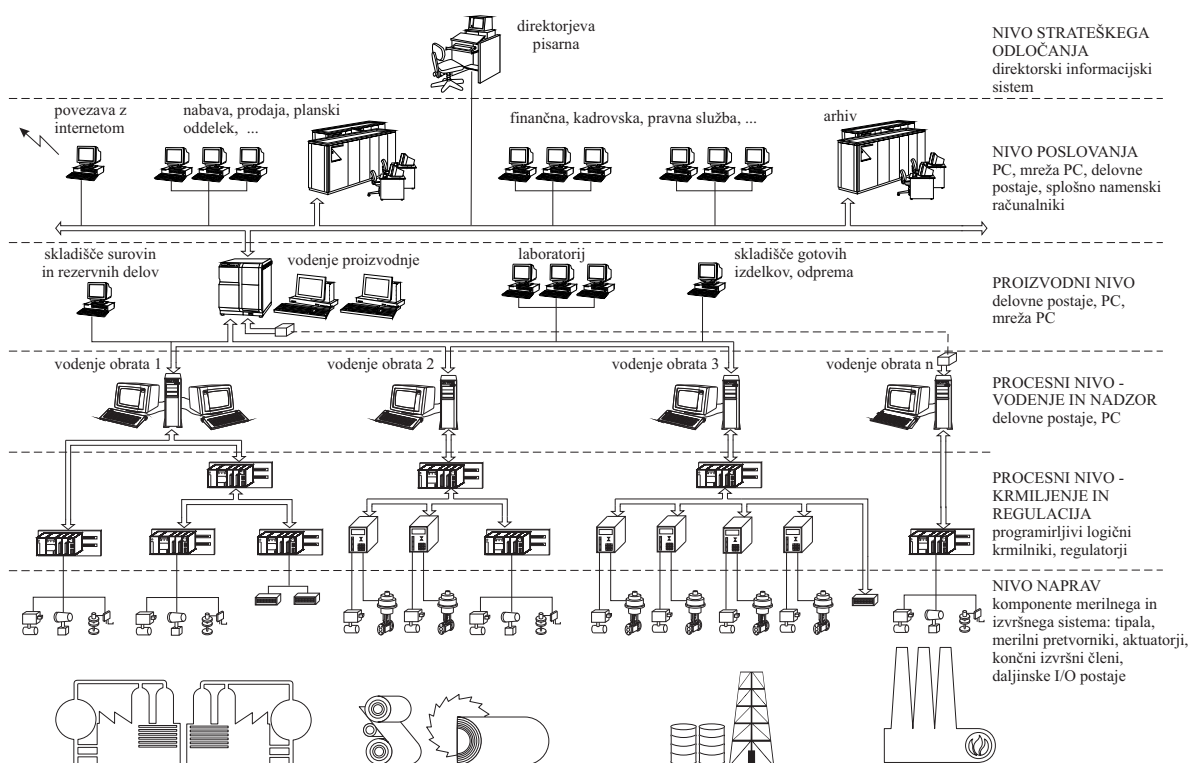
CIM koncept vodenja podjetja namenja sprotni in pravilni informacijski podpori bistveno vlogo. Ažurnost in pravilnost prenosa podatkov lahko zagotovimo le z ustrezno računalniško podporo, zato je potrebno posvetiti posebno pozornost zasnovi računalniškega sistema. Upoštevati je potrebno predvsem:

- zahteve po ažurnosti in pravilnosti informacij;
- zahteve po veliki zanesljivosti delovanja;
- možnost rekonfiguracije oziroma dograjevanja sistema;
- zahteve po zasilnem delovanju ob eventuelnih okvarah ali dograjevanju sistema;
- zahteve po integraciji različnih računalniških sistemov z različno namembnostjo;
- možnost avtonomnosti, selekcije in filtriranja informacij;
- možnost ustrezne zaščite in varovanja podatkov;
- možnost arhiviranja velike količine podatkov.

Sočasno je potrebno ugotoviti, da v celovitem sistemu vodenja proizvodnega podjetja obstaja močan pretok informacij:

- znotraj posameznega nivoja vodenja;
- med posameznimi nivoji vodenja in
- med celotnim sistemom vodenja podjetja in okolico.

Hitrost vzorčenja in količina informacij, ki se prenašajo med posameznimi nivoji vodenja znotraj podjetja se načelno manjša od osnovnih (procesnih) nivojev vodenja proti višjim (poslovnim) nivojem. Količina informacij, ki prihaja v podjetje od zunaj ali jih podjetje pošilja v okolico, se glede na nivoje vodenja povečuje od procesnega nivoja proti poslovnim nivojem. Iz navedenih dejstev in zahtev izhaja, da se pri zasnovi računalniškega sistema za podporo celovitem vodenju podjetja odločimo za porazdeljeni računalniški sistem, katerega dejanska arhitektura pa je odvisna od konkretnega tipa proizvodnje, zahtevane stopnje celovitosti informacijske podpore, velikosti podjetja in razpoložljivih finančnih sredstev. Poenostavljeno splošno shemo porazdeljenega sistema za celovito vodenje proizvodnega podjetja prikazuje slika 1.15. Na njej lahko vidimo tipično razdelitev na različne nivoje vodenja, običajno organizacijske enote v podjetju, tipične naprave, ki jih srečamo na različnih nivojih ter njihovo povezavo v integrirani sistem.

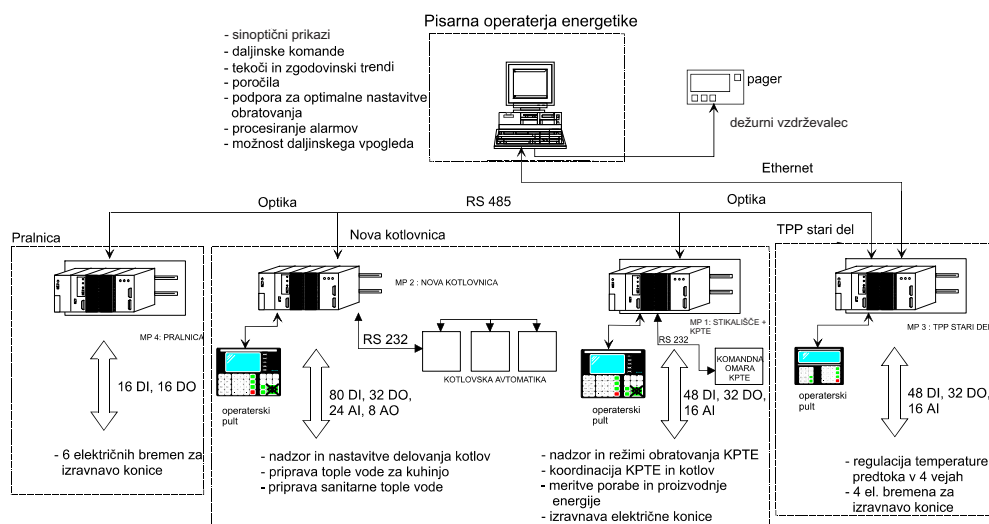


Slika 1.15: Porazdeljeni računalniški sistem kot informacijska podpora vodenju in odločanju v proizvodnem podjetju

Spodnji trije nivoji računalniškega sistema na sliki 1.15 pripadajo procesnemu nivoju piramide podjetja: nivo naprav vsebuje komponente merilnega in izvršnega sistema (merilniki, pretvorniki, ventili, motorji, kontaktorji, ...), nato sledi nivo, ki vsebuje krmilnike in regulatorje, ki na tem spodnjem nivoju vzdržujejo temperature, tlake, zaporedje opravil itd. Najvišji nivo procesnega nivoja je običajno nivo nadzora. Ta nivo lahko skrbi le za uporabniško prijazen prikaz procesnih spremenljivk, lahko pa opravlja tudi kakšne zahtevnejše naloge odkrivanja napak ali optimiranja. Proizvodni nivo je v tej shemi opisan z enim nivojem, poslovni nivo pa iz dveh nivojev: nivo poslovanja in nivo strateškega odločanja.

Primer 1.3 Porazdeljeni sistem za vodenje energetike

Slika 1.16 prikazuje arhitekturo porazdeljenega sistema za vodenje energetike. Iz sheme je razvidna smiselna porazdelitev funkcij sistema, ki zagotavlja vodenje in popoln pregled objektov energetike v podjetju in delno funkcionalnost sistema tudi v primeru izpada posamezne računalniške enote.



Slika 1.16: Primer dvo nivojske arhitekture sistema za vodenje energetike

□

1.8 Gradniki sistemov avtomatskega vodenja

Ker se avtomatiki ukvarjajo predvsem s tehničnimi vidiki izvedbe (glej sliko 1.13), se bomo v tem delu največ ukvarjali s procesi, znanji in orodji. Zadnja stvar na osi tehniških vidikov pa so gradniki. Gradnike bomo vpeljali skladno s sliko 1.3, ki sistem vodenja razdeli na tri faze: opazovanje, odločanje in ukrepanje. Vsaka faza zahteva določene gradnike.

1.8.1 Opazovanje

Bolj tehnično govorimo o merilnem sistemu. La ta običajno sestoji iz tipala (senzorja) in merilnega pretvornika. Tipalo pretvori neko fizikalno veličino v električno veličino (npr. velikostnega reda [mV]). Zato potrebujemo še merilni pretvornik, ki ojača merjene signale v signale industrijskih standardov (največkrat v standardne analogne signale 4-20 mA). Razen zveznih (analognih) signalov, se pojavljajo tudi digitalni signali (0/1, vklop/izklop, ...). Najbolj tipični

so merilniki naslednji veličin:

Premik, hitrost, pospešek (merilni potenciometri, enkoderji - dajalniki pulzov, ...) - slika 1.17



Slika 1.17: Merjenje pomika, hitrosti, pospeška

Merilniki temperature (termočleni, uporovni merilniki, poplprevodniški merilniki, ...), slika 1.18



Slika 1.18: Merjenje temperature

Merilniki tlaka (absolutni, diferencialni, manometri, merjenje z uporavnimi lističi, piezoelektrični, ...), slika 1.19



Slika 1.19: Merjenje tlaka

Merilniki pretoka, nivoja, slika 1.20



Slika 1.20: Merjenje pretoka, nivoja

1.8.2 Ukrepanje

V tehniškem izrazoslovju bomo govorili o izvršnem sistemu, ker preko tega sistema sistem vodenja deluje na realni sistem. V splošnem izvršni sistem sestoji iz aktuatorja ('pogona' izvršnega sistema) in končnega izvršnega člena (npr. motor in ventil). Tipični primeri:

Pogonski motorji in ustrezni krmilniki, slika 1.21



Slika 1.21: Pogonski motorji, krmilniki

Ventili z različnimi vrstami pogona, slika 1.22



Slika 1.22: Ventili

Pnevmatski cilindri in elektro-pnevmatski ventili, slika 1.23



Slika 1.23: Pnevmatiski cilindri in elektropnevmatski ventili

1.8.3 Odločanje

V tem sklopu se nahajajo najbolj inteligentni gradniki sistemov vodenja oz. 'možgani' sistemov vodenja. Le-ti so sposobni razmišljati in odločati tako, da se realni sistem vede, kot se zahteva. Nekaj tipičnih gradnikov prikazuje slika 1.24.



Slika 1.24: Krmilniki, regulatorji in procesni računalniki

Omenjeni gradniki vsebujejo algoritme vodenja. Glede na kompleksnost nalog, ki jih mora sistem vodenja opraviti, se lahko na tem mestu nahaja preprost in cenen mikrokrmilnik. Lahko je to programirljivi logični krmilnik, ki je bolj specializiran za t.i. diskretno in/ali sekvenčno vodenje, ko prevladujejo binarni signali (vklop, izklop, ...). Industrijski regulator pa je bolj specializiran za t.i. zvezno vodenje, ko prevladujejo zvezni signali (npr. regulator nastavlja odprtje mešalnega ventila za doseg ustrežne temperature). Za posebej kompleksne in zahtevne sisteme uporabljamo večje računalniške sisteme, ki jim pravimo procesni računalniki.

Programirljivi logični krmilniki - PLK-ji

Programirljivi logični krmilnik (PLK) (angleško: Programmable Logic Controller - PLC) je osnovno orodje, ki je danes na razpolago za avtomatizacijo industrijskih procesov in naprav. Njegov razvoj je bil pogojen z vedno večjimi potrebami po avtomatizaciji proizvodnje po eni strani, in potrebami po spremembah oz. fleksibilnosti te avtomatizacije po drugi strani (Stenerson, 1993). Pred uporabo PLK-jev, so bile za krmiljenje procesov potrebne velike električne omare s fiksno ožičeno logiko in množico elektomehanskih relejev. Že pri zasnovi takšne logike je bilo precej napak, še več pa pri izdelavi povezav. Toda najhujši problem je nastal, ko je bilo potrebno spremeniti logiko delovanja takšnega krmilja. Tudi pri majhni spremembi (npr. zaporedje operacij) je bilo potrebno izvesti spremembo ožičenja. To je bil strokovno in časovno zelo zahteven proces. Pri tem je trpela tudi proizvodnja, saj jo je bilo potrebno pri tem zaustaviti za daljši čas. Zelo pogoste so bile tudi okvare relejev zaradi mehanskih enot z gibljivimi deli in kontaktov. Zaradi potrebnih sprememb je bilo problematično tudi vzdrževanje ažurne dokumentacije. General Motors je med prvimi spoznal, da bi lahko uporabili računalnik namesto relejev za realizacijo logike. V primeru spremembe načina delovanja se tako menja samo program, povezave pa lahko ostanejo iste. To zahteva bistveno krajši čas zaustavitve proizvodnje in daje veliko večjo fleksibilnost. Problem pa je bil, kako usposobiti ljudi za delo z računalniki.

PLK bi lahko definirali na naslednji način: *PLK je na mikroprocesorju zasnovana krmilna naprava, ki je bila razvita kot nadomestek relejne logike. V osnovi je namenjena za realizacijo sekvenčnega vodenja, omogoča pa tudi reševanje enostavnejših regulacijskih nalog. Strojna in programska oprema je prilagojena za delo v industrijskem okolju in za uporabo "neračunalnikarjev".*

Zaradi množične proizvodnje in uporabe PLK-jev v industriji so ti postali:

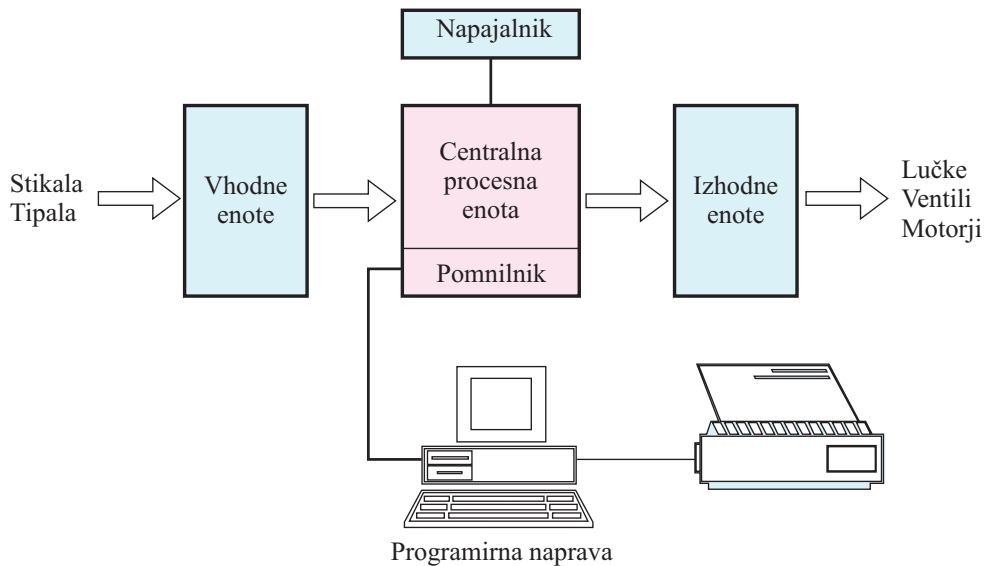
- relativno poceni,
- zelo zanesljivi v delovanju in
- enostavni za uporabo (programiranje, montažo in servisiranje).

PLK-je uporabljamo predvsem za:

- logične operacije,
- krmiljenje,
- zbiranje podatkov za potrebe nadzora in
- kot vmesnik za izvajanje daljinskih komand.

Glavni sestavni deli PLK-ja

Slika 1.25 prikazuje funkcionalne dele PLK-ja.



Slika 1.25: Zgradba PLK-ja

Centralna procesna enota (CPU): To so "možgani" krmilnika. Vsebuje enega ali več mikroprocesorjev, ki krmilijo delovanje PLK-ja in izvajajo uporabniški program. CPU tudi krmili komunikacije in povezave z drugimi enotami sistema.

Sistemiški program je shranjen v ROM pomnilniku, spremenljivke v RAM pomnilniku, uporabniški program pa v RAM ali EEPROM pomnilniku. Razvoj uporabniškega programa poteka pogosto v lestvičnem diagramu (angl. ladder diagram). To je grafična tehnika, ki ima korenine v elekto-mehanskih krmiljih.

Napajalnik: PLK je običajno napajan z 220 V AC ali 24 V DC napetostjo. Napajalnik skrbi za potrebne nižje DC napetosti za delovanje CPU in vhodno-izhodnih enot. Za napajanje zunanjih enot (relejev, kontaktorjev, sond) običajno potrebujemo ločen (zunanji) napajalnik.

Vhodne enote: Skrbijo za sprejem zunanjih signalov (digitalnih in analognih), pretvorbo signalov v nivoje potrebne za nadaljnjo obdelavo in za zaščito CPU pred motnjami iz okolja (optična izolacija, filtriranje, ...). Stanje digitalnih vhodov se običajno prikazuje z LED diodami - z namenom diagnostike delovanja PLK-ja.

Izhodne enote: Skrbijo za prenos rezultatov izvedbe lestvičnega diagrama na izhodne signale (digitalne in analogne) in za zaščito CPU pred motnjami iz okolja (običajno optična izolacija). Stanje digitalnih izhodov se običajno prikazuje z LED diodami.

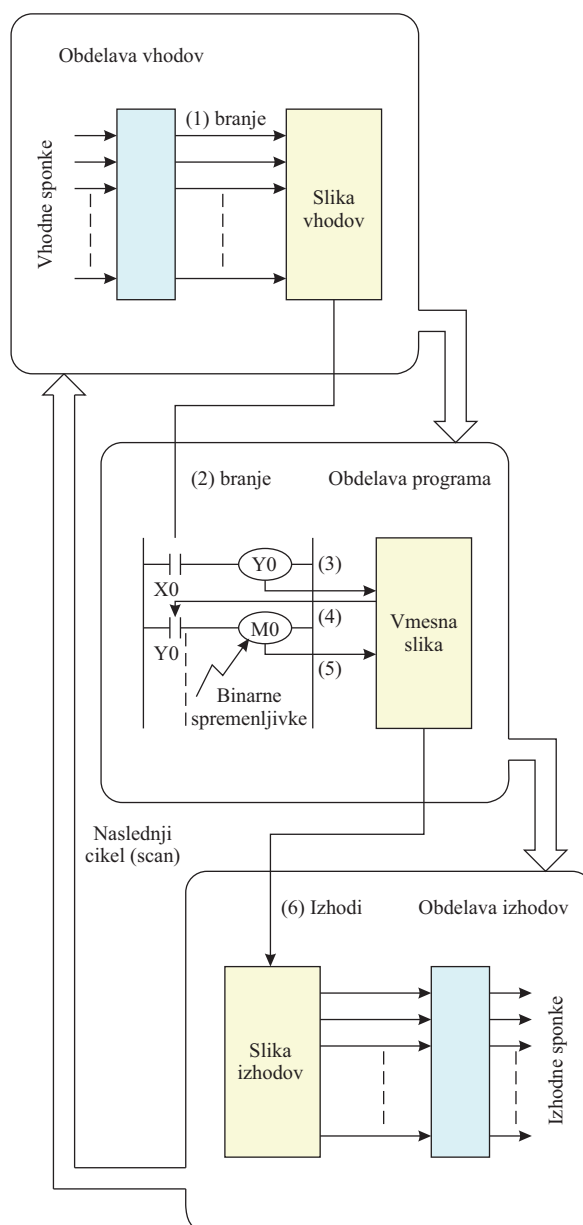
Delovanje PLK-ja

Princip delovanja PLK-ja se precej razlikuje od delovanja programov v "običajnih" računalnikih. Upoštevati moramo, da je osnovna funkcija PLK-ja krmiljenje izhodov na osnovi stanja vhodov. Potek delovanja je naslednji (glej sliko 1.26):

1. odbiranje digitalnih signalov iz vhodnih modulov in preslikava v pomnilnik (v sliko vhodov),
2. izvedba oz. vrednotenje celotnega uporabniškega programa,
3. preslikava iz pomnilnika (iz slike izhodov) na digitalne izhodne module.

Pri tem je pomembno, da v fazi vrednotenja programa ni odbiranja dejanskega stanja vhodov tako, da je celotno vrednotenje z istim stanjem vhodov. To zagotavlja konsistentnost in zmanjša možnost napak pri programiranju.

Rezultati izhodov se med vrednotenjem shranjujejo v sliko izhodov, in le-ti se uporabljajo v naslednjih vrsticah programa za nadaljnja vrednotenja. Dejanska preslikava na izhode pa se izvrši šele po vrednotenju celotnega programa. Celotno



Slika 1.26: Princip delovanja PLKja

zaporedje operacij se izvaja zelo hitro. Ta čas imenujemo čas cikla (čas pregledovanja - scanning time) in zavisi od:

- hitrosti delovanja mikroprocesorja,
- izvedbe strojne opreme, npr. način odbiranja vhodov,

- kvalitete izvedbe interpreterja instrukcij programa,
- dolžine uporabniškega programa,
- vrste uporabljenih instrukcij; enostavne vhodno/izhodne instrukcije se izvedejo zelo hitro, medtem ko kompleksne aritmetične in druge podobne instrukcije porabijo veliko več časa.

Zelo dolgi uporabniški programi se običajno izvedejo v manj kot 20 ms. Proizvajalci PLK-jev podajajo okvirni čas, potreben za vrednotenje tisoč (1K) enostavnih instrukcij uporabniškega programa. To omogoča grobo oceno hitrosti delovanja različnih PLK-jev.

Vrste PLK-jev

Na trgu je veliko vrst PLK-jev različnih proizvajalcev, npr. Allen Bradley, Hitachi, Mitsubishi, Omron, Siemens, Telemecanique, itd. Skoraj vsi imajo v proizvodnem programu širok nabor različnih vrst PLK-jev.

Krmilnike lahko razdelimo po namenu in zgradbi v majhne, srednje in velike. Ločijo se po kapaciteti pomnilnika, po številu vhodno-izhodnih priključkov, pa seveda tudi po ceni (od nekaj sto evrov do več deset tisoč evrov).

Pomožna oprema PLK-jev

Na krmilnik pogosto priključimo tudi operatorski pult. Preko pulta lahko pregledno (v uporabniškem jeziku) prikazujemo stanje procesa, vnašamo parametre in vršimo ukaze kot so npr. zagon stroja. Z njimi lahko bistveno zmanjšamo potrebno število vhodov (potrebnih za operatorske tipke) in izhodov PLK-ja (potrebnih za signalizacijo) ter izvedemo pregledno javljanje poteka delovanja in alarmiranje. Pulti so običajno vezani preko serijskega vodila direktno na PLK. Lahko pa so vezani tudi neposredno na komunikacijsko mrežo, ki povezuje med seboj večje število PLK-jev. Na nekatere pulte lahko priključimo tudi pisalnik za izpis poročil. Prikazovalniki so lahko alfanumerični ali grafični. Pulti so namenjeni za vgradnjo na vrata elektro omare.

Regulatorji

V sodobnih sistemih za vodenje industrijskih procesov se pojavlja veliko regulacijskih zank, ki pretežno vsebujejo proporcionalno-integrirno-diferencirne (PID)

regulacijske algoritme in so realizirane v okviru procesnih računalnikov, v primeru majhnega števila zank pa tudi v programirljivih logičnih krmilnikih (PLK). Tradicionalno pa so se za izvajanje regulacij uporabljali samostojni regulatorji. Ti se v veliko primerih uporabljajo še danes, in sicer predvsem takrat, kadar se zahteva višja stopnja zanesljivosti, hiter dostop operaterja in je prisotnih manj ne preveč zahtevnih regulacijskih zank.

Vrste regulatorjev

V širšem smislu lahko regulatorje glede na izvedbo in princip delovanja razdelimo v naslednje skupine:

- mehanski regulatorji,
- pnevmatski regulatorji,
- analogni elektronski regulatorji,
- mikroračunalniški regulatorji.

Mehanski regulatorji se uporabljajo samo za posebne namene in v primerih, ko ni zahtevana velika natančnost. Prav tako se še vedno uporabljajo pnevmatski regulatorji, in sicer v tistih kemijskih procesih, kjer je zahtevana eksplozijska varnost ali pa je glavna skrb korozijska odpornost. V nasprotju s tem pa klasični analogni elektronski regulatorji, razen izjemoma, niso več primerni za vgradnjo v nove sisteme za vodenje, saj imajo v primerjavi z mikroračunalniškimi regulatorji vrsto slabosti, kot je npr. neudobna in netočna nastavitve PID parametrov in komplicirano konfiguriranje regulatorja (s pomočjo mehanskih stikal). Vendar pa se analogni elektronski regulatorji še uporabljajo v primerih, kadar je v ospredje postavljena hitrost delovanja, kot je to npr. pri servo-pozicioniranju. Mikroračunalniški regulatorji se odlikujejo s široko uporabnostjo, nizko ceno in enostavnim načinom konfiguriranja. Opremljeni so s čelno ploščo, ki vsebuje tipkovnico in ustrezne prikazovalnike. Aparaturna oprema mikroračunalniških regulatorjev je bolj ali manj fiksna. Prikazovalniki prikazujejo regulirano, želeno in regulirno vrednost, tipke pa so namenjene ročnemu načinu vodenja, preklopu ročno/avtomatsko in nastavitvi želene vrednosti. Vse to omogoča operaterju, da lahko ustrezno in hitro reagira, kadar se pojavijo nepravilnosti v delovanju regulacijske zanke.

Funkcije mikroračunalniških regulatorjev

Osnovna naloga samostojnega mikroračunalniškega regulatorja je izvedba PID algoritma v regulacijski zanki. Poleg osnovne regulacijske funkcije pa obstaja v okviru mikroračunalniških regulatorjev še niz dodatnih funkcij, ki se izkazujejo kot zelo uporabne za izvedbo sodobnih sistemov za vodenje. Te funkcije so:

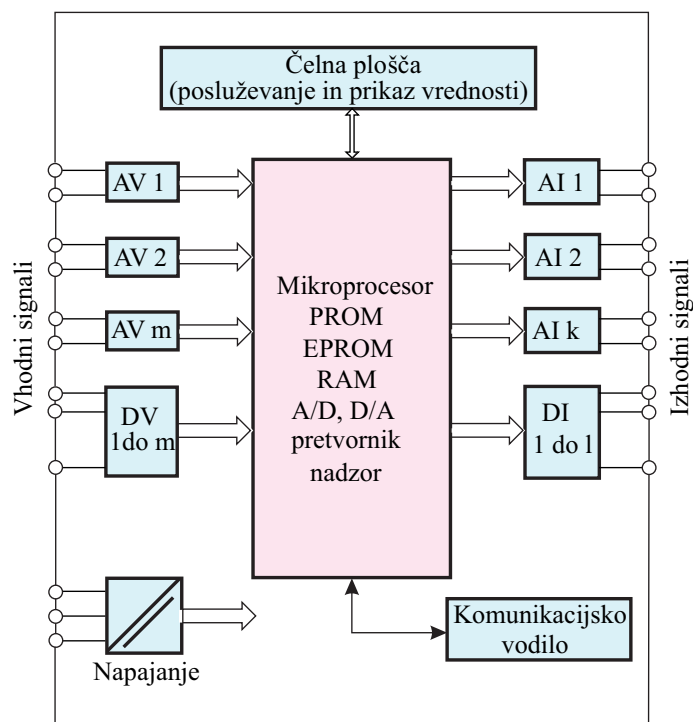
- prikazovanje in izpisovanje podatkov,
- alarmiranje,
- izbira med lokalno nastavljivo želeno vrednostjo in zunanjo želeno vrednostjo, ki se podaja preko signala iz nekega nadzornega mesta,
- kaskadna regulacija (dva serijsko povezana PID regulatorja), ki se izvede znotraj mikroračunalniškega regulatorja,
- regulacija razmerja procesnih veličin,
- izvajanje različnih logičnih in krmilnih funkcij itd.

Zgradba in princip delovanja tipičnega mikroračunalniškega regulatorja

Zgradba tipičnega mikroračunalniškega regulatorja je močno poenostavljeno prikazana na sliki 1.27.

Osrednja enota vsakega mikroračunalniškega regulatorja je mikroprocesor skupaj s pripadajočimi spominskimi enotami (RAM, ROM, EPROM). Mikroprocesor omogoča izvajanje uporabniškega programa regulatorja, to je izvajanje ene ali več regulacijskih zank ter ostalih aritmetičnih in logičnih funkcij, katerih izvedbo omogoča programska oprema regulatorja. Spominske enote služijo shranjevanju programske opreme regulatorja, uporabniškega programa in vrednosti signalov ter parametrov. Pri izvajanju uporabniškega programa se zajamejo analogni vhodni električni signali (AV) in digitalni (DV) signali. Izmerjene - regulirane vrednosti procesnih veličin se s pomočjo A/D pretvornika pretvorijo iz analogne v digitalno obliko. Le-ta je primerna za nadaljnjo obdelavo s pomočjo algoritma v mikroprocesorju. Rezultat obdelave so ustrezna števila (regulirni signal v digitalni obliki), ki se s pomočjo D/A pretvornika pretvorijo v ustrezne analogne izhodne električne signale (AI) - regulirne veličine. Na mikroprocesor je posredno priključena tudi čelna plošča, ki služi za prikaz vrednosti posameznih signalov ter za nastavitve vrednosti različnih parametrov.

Funkcije čelne plošče



Slika 1.27: Zgradba mikroračunalniškega regulatorja

Tipičen mikroračunalniški regulator ima čelno ploščo, ki ostane vidna tudi po vgraditvi regulatorja v komandni pult. Elementi čelne plošče so v večini primerov naslednji:

- tipke (služijo za poseganje operaterja v proces, nastavljanje vrednosti procesnih veličin, ipd.),
- analogni prikazovalniki (v analogni grafični obliki prikazujejo želeno in dejansko vrednost procesne veličine, regulirno veličino, oziroma ostale analogne veličine),
- digitalni (alfa-numerični) prikazovalniki (v obliki števila prikazujejo želeno in dejansko vrednost procesne veličine, regulirno veličino, oziroma ostale analogne veličine),
- digitalni svetlobni indikatorji - LED diode (prikazujejo stanja digitalnih on/off signalov).

Slika 1.28 prikazuje čelne plošče Siemensovih regulatorjev serije DR.



Slika 1.28: Serija Siemensovih regulatorjev DR

Programiranje regulatorja

Glede na programsko opremo ločimo dva tipa mikroračunalniških regulatorjev, in sicer tovarniško pred-programirane regulatorje ter prosto programirljive regulatorje.

V prvem primeru ima mikroračunalniški regulator tovarniško vstavljen program, ali nekaj programov, ki omogočajo reševanje nekaterih tipičnih regulacijskih problemov. Tovarniški programi vsebujejo poleg regulacijskega PID algoritma v glavnem še preproste aritmetične operacije in pa prikaz reguliranih (merilnih) in regulirnih veličin. Uporabnik-programer regulatorja lahko v tem primeru le minimalno vpliva na delovanje programa, tako da izbere, kateri od tovarniško vstavljenih programov naj se izvaja oziroma spreminja parametre, ki vplivajo na delovanje izbranega programa.

Drugi primer obsega naprednejše mikroračunalniške regulatorje, ki so namenjeni izvajanju kompleksnejših in nestandardnih regulacijskih nalog in so temu primerno opremljeni s kompleksnejšo programsko opremo. Uporabnik-programer ne uporabi tovarniško vgrajenega programa, pač pa program izdelava sam in ga tako do podrobnosti prilagodi konkretnemu problemu. Pri izdelavi programa se uporabljajo in povezujejo tovarniško pred-programirani programski bloki oziroma podprogrami, ki so namenjeni za reševanje različnih funkcij. Pri programiranju regulatorjev obstajata dve možnosti:

- programiranje preko tipk in prikazovalnikov čelne plošče predstavlja sicer časovno potratno in utrudljivo možnost, ki pa je primerna v prvi vrsti za manjše popravke obstoječih programov, manj pa za kompleten razvoj novih programov,
- programiranje preko razvojnega sistema (npr. SIPROM za Siemensov regulator DR 24), ki je običajno kar osebni računalnik z ustrezno razvojno programsko opremo. Razvojni sistem omogoča razvoj uporabniških programov, njihovo dokumentiranje ter prenos v regulator ter tako predstavlja boljšo in hitrejšo možnost programiranja v primerjavi s programiranjem preko čelne plošče.

1.8.4 Naprednejši gradniki sistemov vodenja

Zgoraj predstavljeni gradniki predstavljajo osnovne gradnike računalniško vodenih sistemov. Z razvojem tehnologije tudi merilniki in aktuatorji pridobivajo več inteligence:

- Včasih se dve ali celo več komponent v fazah merjenja, odločanja in ukrepanja združuje v eno komponento.
- Meja med krmilniki in regulatorji je vse bolj zabrisana, ker imajo krmilniki vedno več sposobnosti zveznega vodenja, regulatorji pa omogočajo tudi logične in pomnilniške operacije, ki so potrebne za diskretno in sekvenčno vodenje.
- Zlasti se inteligenca povečuje v krmilnikih, regulatorjih in procesnih računalnikih. Konvencionalnim algoritmom, ki predvsem temeljijo na proporcionalno-integrirno-diferencirni regulaciji, se dodajajo pristopi, ki so značilni zlasti za področje umetne inteligence:
 - mehka logika (fuzzy logic),
 - umetne nevronske mreže (artificial neural nets) in
 - ekspertni sistemi.
- Lep primer večje inteligence v merilnem sistemu pa je uporaba računalniškega vida. Način je zlasti primeren v mehanskih (npr. robotskih) sistemih, ko je možno s pomočjo kamere razpoznavati pomike, hitrosti in pospeške. Zlasti pa je računalniški vid pomemben za kontrolo kvalitete, pri odkrivanju napak itd.

- V računalniških sistemih vodenja imajo vse večji pomen komunikacijska omrežja. Medtem ko so bile komponente v konvencionalnih sistemih povezane predvsem s standardnimi tokovnimi signali 4-20 [mA], pa se vse bolj uporabljajo digitalne (tudi brezžične) komunikacije, ki omogočajo enostavnejše prenose na oddaljene lokacije, signali pa so tako manj podvrženi motnjam. Uveljavljeni so t.i. asinhronski načini komunikacije, ki so primerni za doseg manjših hitrosti prenosa (PROFIBUS, običajni serijski prenosi RS232, ...) in sinhronski načini prenosa za večje hitrosti (npr. ETHERNET, Bitbus, IBM-ov sistem SDLC, ...).

Sodobna avtomatizacija tako danes temelji na vgrajevanju kompleksnejših sklopov, med katerimi so zlasti sodobni robotski manipulatorji, mobilni sistemi in sklopi z računalniškim vidom in /ali govorom. Slika 1.29 prikazuje nekaj robotskih manipulatorjev, slika 1.30 pa nekaj mobilnih sistemov.

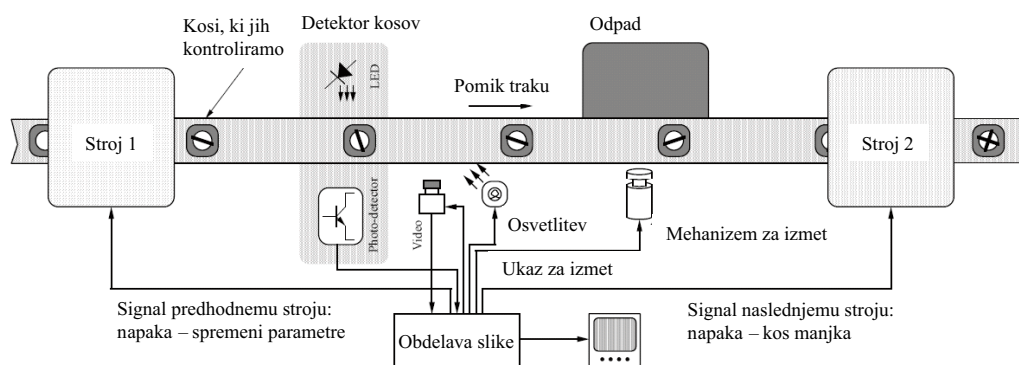


Slika 1.29: Robotski sistemi

Slika 1.31 pa prikazuje računalniški sistem, ki uporablja računalniški vid. Izdelke na tekočem traku spremlja kamera in na osnovi analize slike se računalniški sistem odloči, ali je izdelek ustrezen ali ne. V primeru, da izdelek ni ustrezen, ga mehanizem za izmet odstrani.



Slika 1.30: Mobilni sistemi



Slika 1.31: Sistem vodenja z računalniškim vidom

1.9 Sistemski pristop pri načrtovanju vodenja (regulacije)

Slika 1.13 prikazuje štiridimenzionalni pogled na načrtovanje, izgradnjo in uporabo sistemov za vodenje. Tretjo dimenzijo smo poimenovali dimenzijo *življenjskega cikla*, sestavljajo pa jo posamezne faze kot npr: definicija zahtev, opredelitev specifikacij, načrtovanje, izvedba (izgradnja), zagon in uvajanje, obratovanje (delovanje) in upokojitve.

V okviru tega cikla bomo podrobneje opisali tiste faze, ki jih bomo v tem delu pa tudi pri predmetih v naslednjih letnikih zlasti na smeri Avtomatika bolj podrobno obravnavali. Opisali bomo, kako se na dovolj sistematičen način lotimo načrtovanja vodenja (predvsem regulacije) s pomočjo modeliranja. Faze zagona, obratovanja in upokojitve bomo na tem mestu izpustili.

Načrtovanje vodenja sistemov ima značilnosti koračnih kot tudi zančnih (cikličnih) postopkov. V vsakem koraku, katerega namen je približevanje h končni rešitvi problema, se namreč iterativno približujemo končni rešitvi. Seveda iteracije niso omejene le na določen korak, temveč so značilne za celotni postopek.

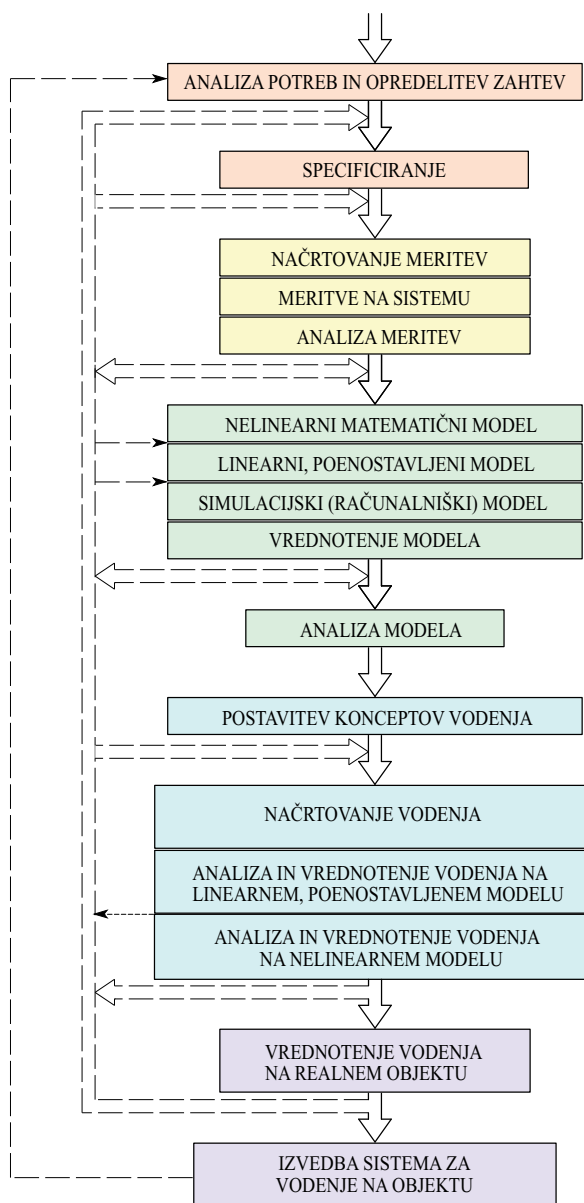
Potek načrtovanja (od nastopa problema vodenja do izpolnitve zastavljenega cilja) najlažje ponazorimo s shemo. Slika 1.32 prikazuje običajne korake postopka in iterativne zanke med njimi. Pri konkretnem načrtovanju je potrebno izvesti le nekatere izmed njih, največkrat pa kar vse.

Analiza potreb in opredelitev zahtev je prvi korak. Tu je nujno potrebno sodelovanje med inženirji v podjetju, vzdrževalci, tehnologi in ostalimi, ki dobro poznajo proces, in avtomatiki, ki poznajo metode, orodja in gradnike vodenja. Natančno identificiramo cilje in probleme, ki jih želimo reševati. Iz tega nastanejo osnovne **specifikacije in koncepti**.

Za načrtovanje vodenja kompleksnejših sistemov je potreben **matematični model sistema**. Model za načrtovanje vodenja koristno uporabimo na različne načine: z njim lahko bolje spoznamo mehanizme delovanja v realnem sistemu, lahko ga uporabljamo zgolj za preizkušanje algoritmov vodenja, model je včasih potreben v načrtovalnem postopku, včasih pa je del sheme vodenja. Pri modeliranju moramo jasno izraziti namen, cilje, omejitve in natančnost. Model lahko izpeljemo iz fizikalnih zakonov procesa (teoretično modeliranje) ali pa ga določimo z eno izmed identifikacijskih metod (eksperimentalno modeliranje). V obeh primerih je možno določiti ustrezne vrednosti parametrov modela le na osnovi določenih eksperimentov (meritev) realnega sistema.

Meritve so torej nujno potrebne (če proces že obstaja) in so ponavadi prvi korak postopka. Model nikoli ne more biti boljši od meritev, zato slabe meritve vodijo do slabega modela. Priprava meritev in načrtovanje eksperimentov sta zelo pomembna elementa načrtovalnega postopka. Planiranje meritev temelji na izkušnjah, intuiciji in splošno znanih fizikalnih zakonih, ki jih uporabimo pri obravnavi konkretnega realnega objekta in njegovih komponent. Rezultati planiranja meritev so:

- seznam veličin, ki jih je potrebno meriti, in njihova narava glede na problem vodenja (regulirne, regulirane, motilne in spremljajoče veličine),
- seznam eksperimentov, ki jih je potrebno opraviti (načini vzbujanja sistema pri meritvah),



Slika 1.32: Iterativni postopek načrtovanja in implementacije sistema za vodenje (regulacijo)

- pridružene definicije, kot so izbor in način priključitve merilne opreme, časi vzorčenja, preračunski faktorji ter nastavitev filtrov, s katerimi izločamo motnje procesa in kompenziramo vpliv diskretnega odjemanja vzorcev.

Običajno sta potrebni dve vrsti meritev:

- statične in
- dinamične.

Statične meritve so osnova za določevanje parametrov posameznih komponent procesa ali procesa v celoti. Meritve izvedemo tako, da pri konstantnih vrednostih vplivnih veličin izmerimo (v ustaljenem stanju) vrednosti odvisnih veličin. Tako dobimo t.i. statične karakteristike komponent. Pri dinamičnih meritvah, katerih rezultati so časovni odzivi sistema v izbranih delovnih točkah, vzbujamo sistem s preizkusnimi, časovno spremenljivimi poteki vplivnih veličin. Izmerjeni časovni poteki vhodov in izhodov so osnova za razvoj matematičnega modela.

Sledi analiza meritev. Z njo si izpopolnimo začetna znanja o sistemu, izločimo očitne merilne napake in obdelamo meritve (npr. normiranje, kalibriranje, glajenje, filtriranje).

Naslednji korak predstavlja **gradnja dinamičnega modela**. V postopku modeliranja običajno najprej dobimo nelinearni matematični model, v nadaljnjih korakih pa ta model lineariziramo in poenostavljamo v skladu z nameni modela. Model moramo prevesti tudi v obliko, ki je primerna za eksperimentiranje, t.j. v numerični (simulacijski) model-v računalniški simulacijski program. Ena od najpomembnejših faz v modeliranju je vrednotenje modela ali validacija. S tem preverimo, ali se model za dane cilje in ob določenih omejitvah vede tako kot realni sistem. Za vrednotenje modela največkrat uporabljamo simulacijo, pa tudi nekatere druge metode analize v časovnem ali frekvenčnem prostoru (npr. Bodejev diagram, analiza lege polov in ničel, ...).

Sledi **analiza modela**. Tu skušamo z analitičnimi metodami bolje spoznati sistem, oziroma dobiti občutek o lastnostih sistema. Možne metode so simulacija, transformacije modela v druge možne oblike zapisa, metode analize v časovnem in frekvenčnem prostoru ter v prostoru stanj.

Nato sledi **načrtovanje vodenja** v ožjem smislu (sinteza vodenja). Tu so na voljo številne metode. Vendar je postavitve koncepta sistema za vodenje odvisna predvsem od znanja in izkušenj načrtovalca. Najpomembnejši cilji, ki jih želimo doseči, so stabilno delovanje, želeni prevzpon ter primerna hitrost reagiranja pri spremembi zelenih vrednosti (sledilno delovanje), hitro odpravljanje motenj v regulacijski zanki (regulacijsko delovanje), robustno delovanje pri predvidljivih in nepredvidljivih spremembah parametrov obratovanja, itd. Za doseg ciljev se uporabljajo naslednje metode:

- simulacija in optimizacija,
- metode za doseg želenih zaprtozančnih polov (diagram lege korenov, metode za premikanje polov),
- frekvenčne metode (Bodejev diagram, polarni diagram in Nyquistov diagram, Nyquistov stabilnostni kriterij, ojačevalni in fazni razločki, inverzni Nyquistov diagram, metoda karakterističnih mest),
- metode sinteze v časovnem prostoru (optimalni regulator stanj), itd.

Po načrtovanju je nujno vrednotenje, ki je podobno vrednotenju modela. Celotni sistem je namreč podan z modelom, ki združuje model realnega procesa ter načrtane elemente za vodenje. Vrednotenje običajno izvedemo najprej s pomočjo poenostavljenega, lineariziranega modela, nato pa še s pomočjo nelinearnega modela. Pri tem je potrebno posvetiti posebno pozornost zahtevam, ki smo jih opredelili na začetku.

Zadnji korak načrtovanja je **izvedba sistema vodenja** na realnem objektu, kar v nobenem primeru ni preprosto. Vrednotenje na realnem objektu vključuje vse dejavnike končne izvedbe z uporabo razvojnega računalnika. Končna rešitev pa vključuje izvedbo načrtanih algoritmov na namenski materialni opremi (npr. procesni računalnik, industrijski regulator, programirljivi logični krmilnik, ...). Pri izvedbi je potrebno vključiti še vrsto zahtev, ki so v načrtovalnem postopku zaradi poenostavljanja namerno izpuščene (npr. možnost preklapljanja ročno - avtomatsko, način merjenja in obdelava merjenih signalov, alarmi, različni grafični prikazi in komunikacije z operaterjem, ...).

Podani opis postopka načrtovanja vodenja sistemov ni popoln. Njegov namen je prikazati le tipične korake, njihov vrstni red in nakazati, kakšne metode in postopke bi morala vključevati sodobna orodja za računalniško podprto načrtovanje (inženirstvo). Uporabimo ga lahko za orientacijo pri pripravi načrtovalnega postopka. Iz njega je razvidna kompleksnost in raznovrstnost opravil, ki jih mora načrtovalec izvesti. Če izvzamemo analizo potreb in opredelitev zahtev ter izvedbo na realnem objektu, t.j. faze, ki jih zelo težko posplošimo zaradi specifičnosti konkretne aplikacije in odvisnosti od programske in strojne opreme, so v načrtovalnem postopku najbolj bistveni postopki eksperimenti (meritve), modeliranje, analiza in načrtovanje sistema vodenja. **Zato načrtovanje sistemov vodenja s pomočjo modeliranja in simulacije predstavlja rdečo nit nadaljnje obravnave. Taka obravnava zahteva tudi ustrezno analizo tako modelov kot sistemov vodenja in nekaj uvodnih znanj iz teorije sistemov in signalov.**

2.

Sistemi in signali

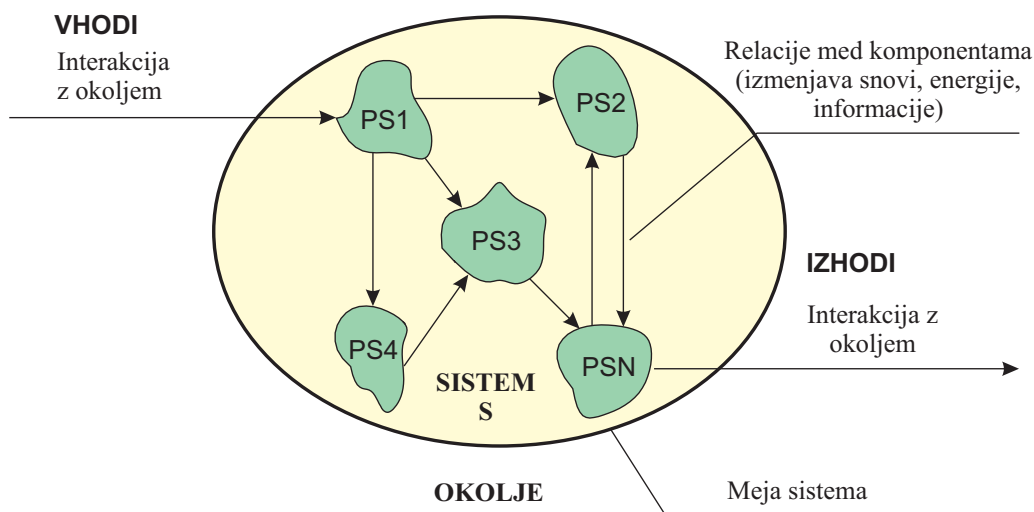
Uspešno reševanje problemov vodenja zahteva po eni strani poglobljeno razumevanje samih sistemov in procesov, ki jih vodimo, njihove strukture, dinamike, medsebojnih relacij, splošnih zakonitosti in podobnosti, ki se pojavljajo na različnih področjih, itd., po drugi strani pa obvladovanje splošnih konceptov 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.

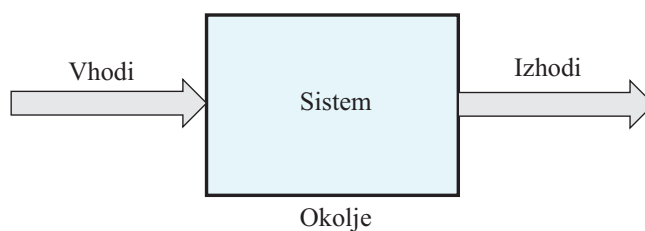
Vidimo, da ima sistem S neko notranjo strukturo, ki je definirana z različnimi komponentami. Vsaka komponenta ima svoje značilnosti in je običajno zopet sistem nižje kompleksnosti (podsistemi PS_1 do PS_N). Komponente so povezane z relacijami. Te relacije običajno pomenijo izmenjavo snovi, energije in/ali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

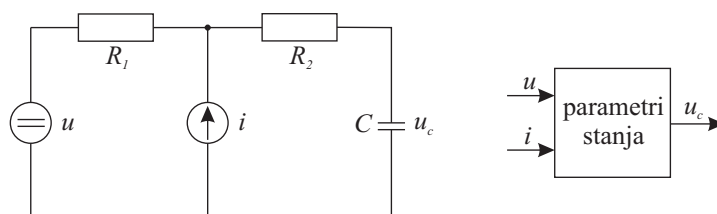
Mnogokrat nas notranja struktura sistema ne zanima, zanima pa nas medsebojna odvisnost med vhodi in izhodi sistema in njegova povezanost z okoljem. Zato često sistem predstavimo tudi v obliki, ki jo prikazuje slika 2.2.



Slika 2.2: Vhodno-izhodni prikaz sistema

Primer: Slika 2.3 prikazuje, kako električno shemo spremenimo v vhodno-izhodni blok. V primeru bloka nas notranjost ne zanima, zanimajo pa nas običajno njegovi 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 ogrođjem, 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.

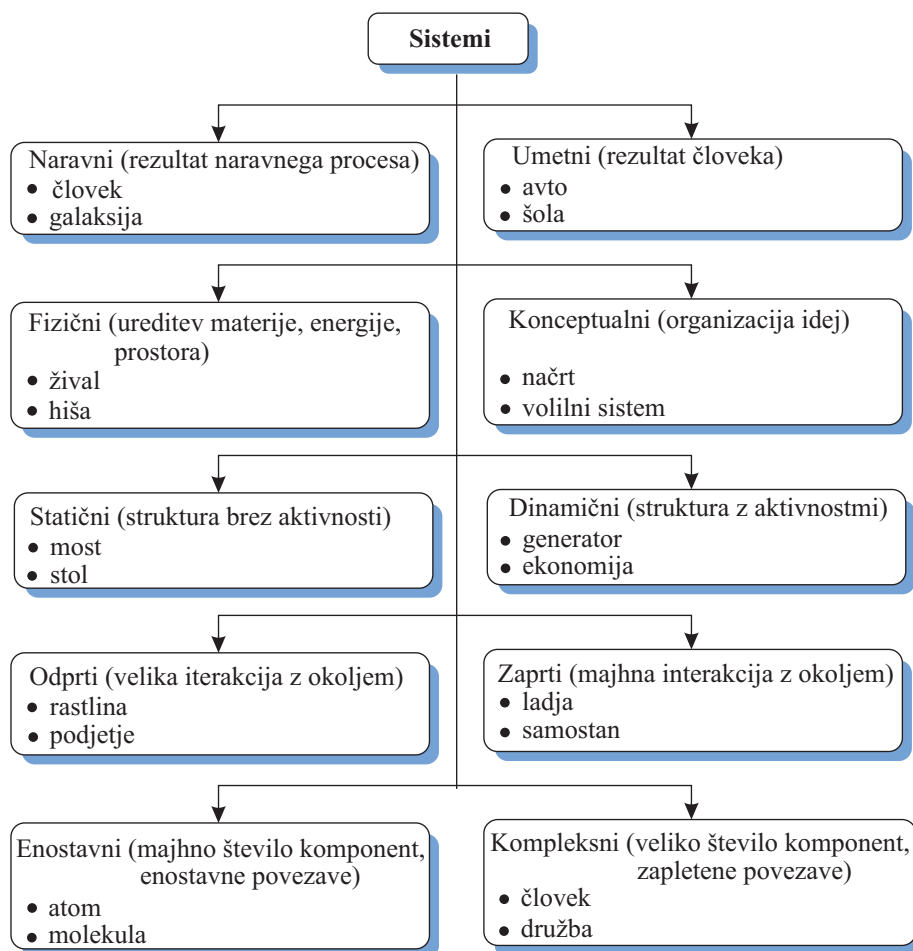
2.2 Različni primeri sistemov

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.

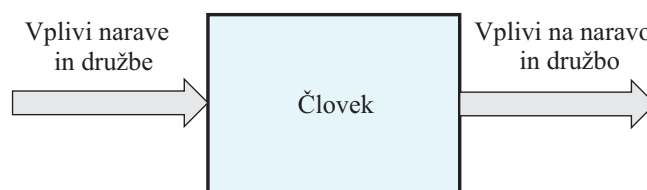
Vidimo, da vhode lahko opredelimo kot vplive narave in družbe na človeka, izhode pa kot človekov vpliv na naravo in družbo. Če želimo človeka kot sistem klasificirati, bi lahko rekli, da gre za naravni, fizični, dinamični, odprti in kompleksni sistem.

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

Vhode v sistem predstavljajo električna energija in televizijski signali v obliki



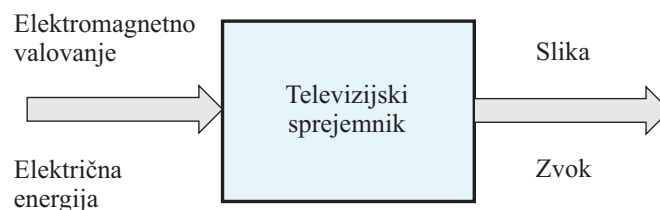
Slika 2.4: Klasifikacija sistemov



Slika 2.5: Človek kot sistem

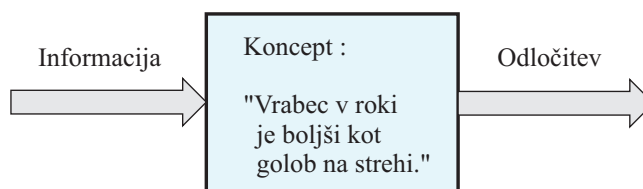
elektromagnetnega valovanja. Izhoda iz sistema sta slika in zvok. Televizijski sprejemnik bi lahko opredelili kot umetni, fizični, dinamični, odprti in srednje enostavni sistem.

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



Slika 2.6: Televizijski sprejemnik kot sistem

"Vrabc 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 je 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.



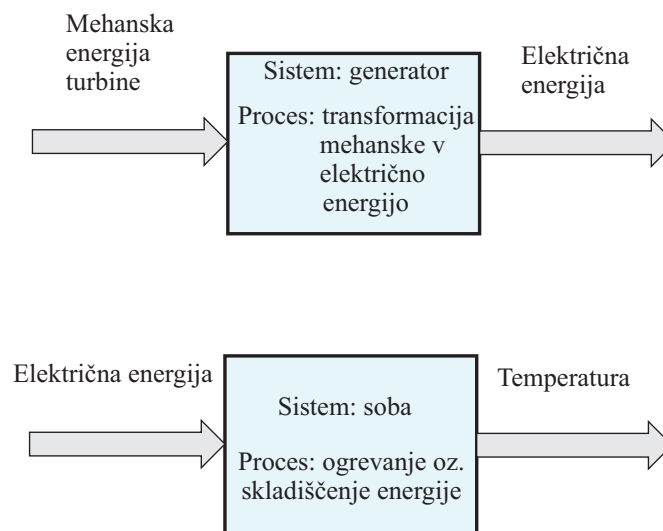
Slika 2.8: Shematski prikaz procesa

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 (n.pr.: v izdelni industriji proizvodi iz metalov, lesa, plastike, stekla itd.). Tako modeliranje kot vodenje temeljita na teoriji diskretnih dogodkov (angl. discrete event systems).



Slika 2.9: Sistema generator in soba s pripadajočima procesoma

Pri zveznih (kontinuirnih) procesih masa in/ali energija prihajata skozi enote procesa v neprekinjenih tokovih (n.pr. v parnih kotlih, toplotnih izmenjevalnikih, pretočnih kemičnih reaktorji itd.). Obravnava temelji na teoriji zveznih dinamičnih sistemov.

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.

2.4 Bistvo systemske teorije v povezavi z modeliranjem

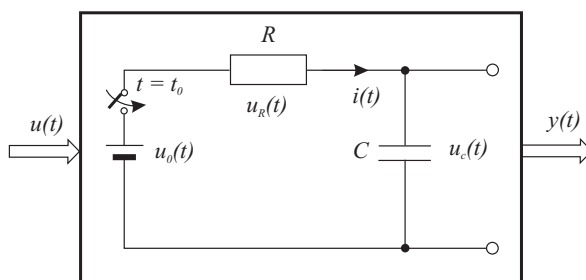
Človek je v celotnem obdobju svoje zgodovine proučeval naravne sisteme ter snoval in gradil umetne sisteme. Rezultat tega razvoja je bilo spoznanje, da pri proučevanju strukture sistemov, njihove dinamike, medsebojnega vpliva, razvoja in vodenja naletimo na pojave, ki niso lastni samo sistemu določene 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

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 2.10.



Slika 2.10: RC vezje

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 zakonov pridemo v nekaj korakih do enačbe 2.2.

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

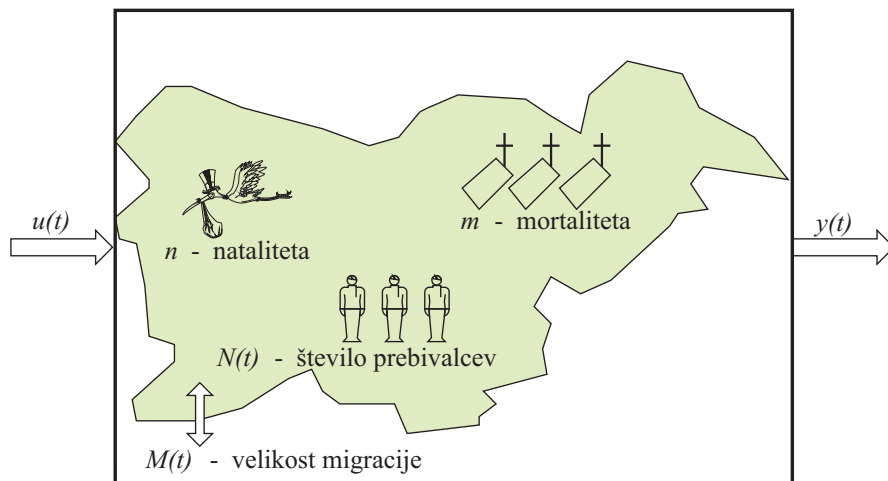
$$y(0) = u_C(0) = y_0$$

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

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

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 migracije $M(t)$ (glej sliko 2.11).



Slika 2.11: Prebivalstvo države

Intuitivno lahko razumemo, da je sprememba številčnosti $\frac{dN(t)}{dt}$ odvisna od razlike med novo rojenimi in umrlimi $((n - m)N(t))$ in številom priseljenih $M(t)$.

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

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.

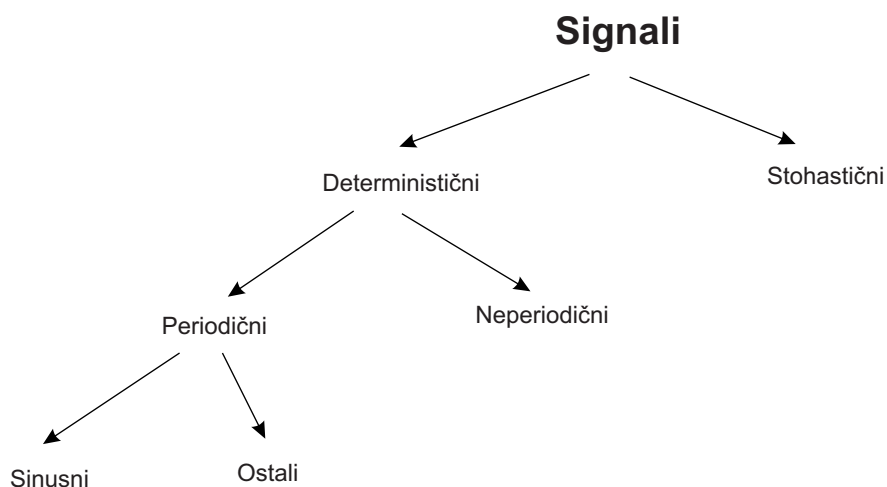
Fizikalno realni signali so signali, ki nas obdajajo v prostoru in času in jih lahko zaznamo z ustreznimi tipali (senzorji). Taki signali imajo omejeno velikost-amplitudo, omejeno energijo in omejen spekter.

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 2.12).

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

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



Slika 2.12: Delitev signalov

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

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

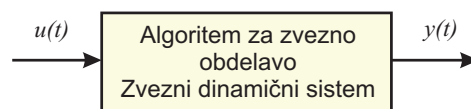
Povprečna moč signala

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 2.13.



Slika 2.13: Zvezna obdelava signalov

Dandanes pa se za obdelavo signalov pa tudi za sprotno vodenje procesov vse bolj uporabljajo digitalni računalniki (slika 2.14, 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 2.14a).

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 2.14a 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 2.14b).

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 2.14b v 2.14c).

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 2.14d).

Postopek digitalne obdelave signalov ob predpostavljenih 3 bitnih pretvornikih z območjem 0-7V prikazuje slika 2.14.

Slika 2.15 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).

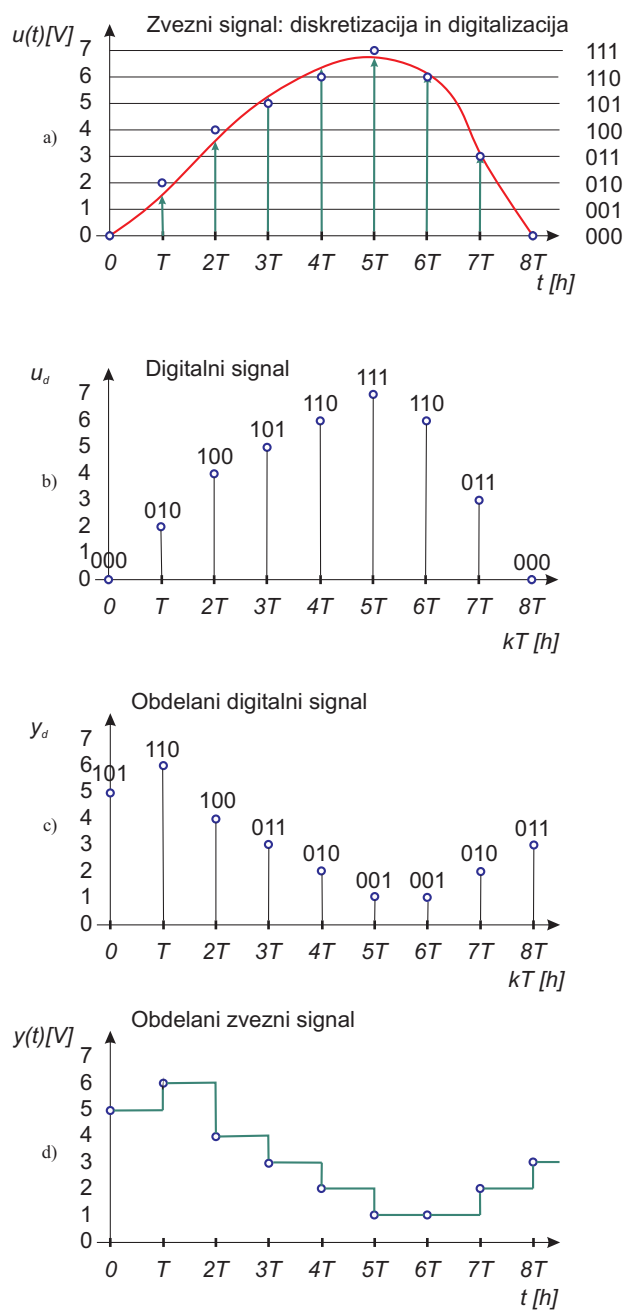
2.5.5 Temeljni signali

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 2.16)

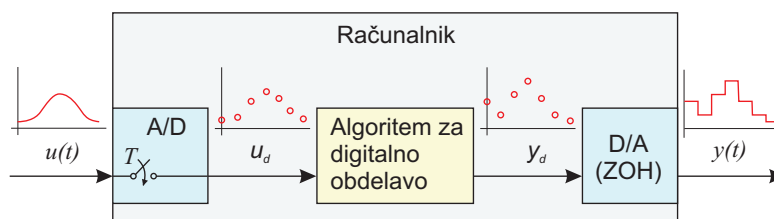
$$f(t) = konst \quad (2.11)$$



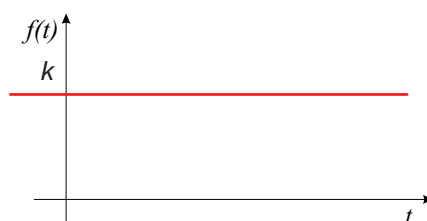
Slika 2.14: Digitalna obdelava signalov

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



Slika 2.15: Računalniški sistem za obdelavo signalov ali za vodenje sistemov



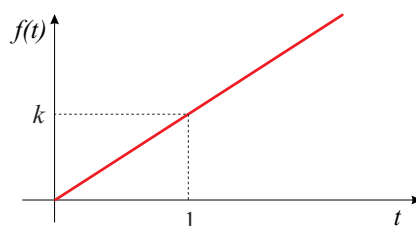
Slika 2.16: Konstanta

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 2.17). Opisuje ga enačba

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



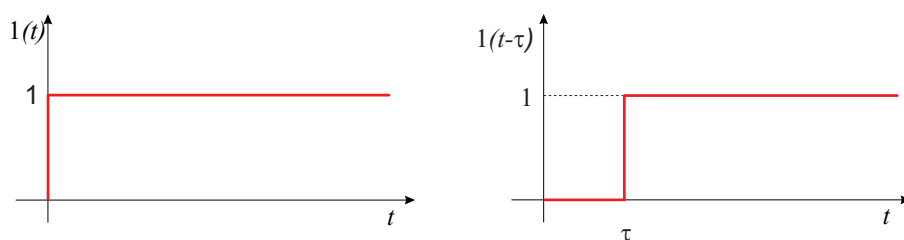
Slika 2.17: Linearno naraščajoči signal

Enotin stopničasti signal

Enotina stopnica se od konstante loči po tem, da je za negativne čase vrednost

enaka nič (glej sliko 2.18).

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



Slika 2.18: Stopnica in zakasnjena stopnica

Slika 2.18 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 preklonov 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.

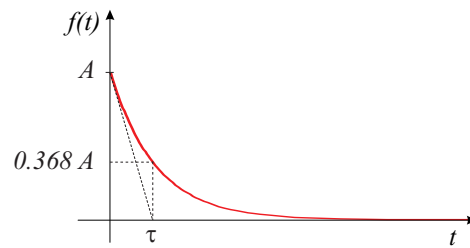
Eksponentni signali

Pri proučevanju električnih vezij se kondenzator preko upora prazni po eksponentnem zakonu (slika 2.19)

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

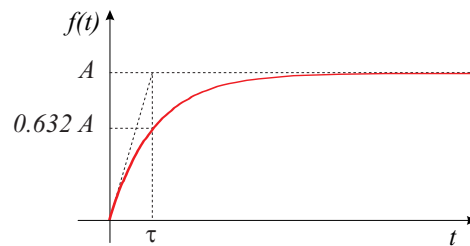
A je začetna vrednost napetosti na kondenzatorju. Končna vrednost napetosti je nič. $\tau = \frac{1}{a}$ je časovna konstanta. V času ene časovne konstante se kondenzator izprazni na 36.8% začetne vrednosti.

Če pa na RC člen v trenutku $t=0$ priklopimo enosmerno napetost A , pa se napetost na kondenzatorju polni po zakonitosti (slika 2.20)



Slika 2.19: Eksponentna funkcija- upadanje

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



Slika 2.20: Eksponentna funkcija- naraščanje

Sinusni signali

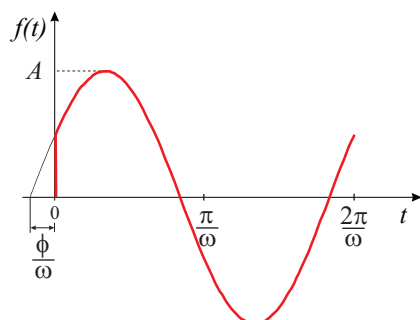
V električnih vezjih so vzbujaalni signali ter signali na elementih v ustaljenem stanju pogosto v obliki sinusnih ali cosinusnih funkcij (sliki 2.21 in 2.22)

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

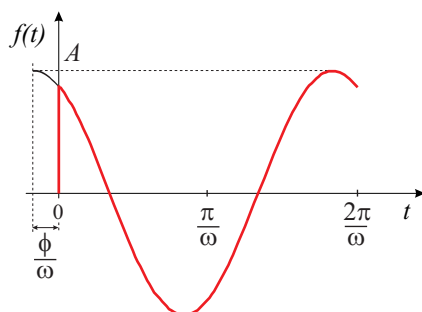
$$f(t) = \begin{cases} A \cos(\omega t + \phi) & ; t \geq 0 \\ 0 & ; t < 0 \end{cases} \quad (2.17)$$

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



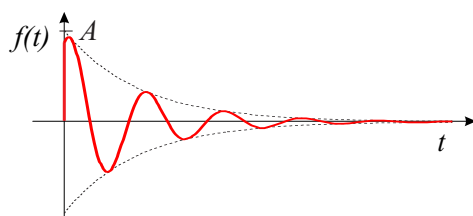
Slika 2.21: Sinusna funkcija



Slika 2.22: Kosinusna funkcija

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 2.23)

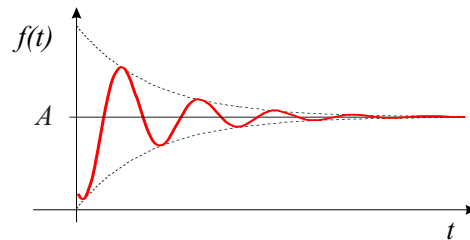
$$f(t) = \begin{cases} Ae^{-at} \cos(\omega t + \phi) & ; t \geq 0 \\ 0 & ; t < 0 \end{cases} \quad (2.18)$$



Slika 2.23: Dušena kosinusna funkcija - upadanje

Pri polnjenju kondenzatorja pa ima napetost obliko (slika 2.24)

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



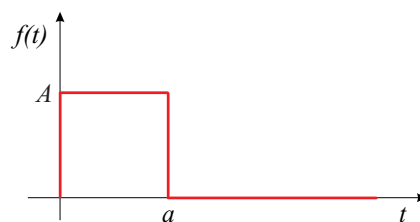
Slika 2.24: Dušena kosinusna funkcija - naraščanje

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. Pri nekem dovolj velikem a nihanja izginejo.

Pravokotni impulz

To je prav tako pogosta funkcija zlasti uporabna pri testiranju vezij (slika 2.25)

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



Slika 2.25: Pravokotni impulz

Enotin pravokotni impulz

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

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

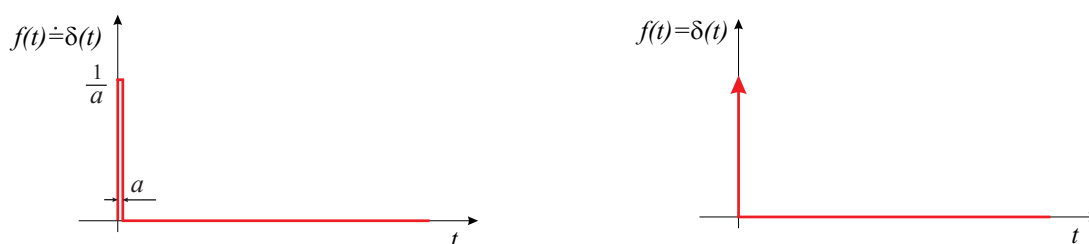
Veljati mora torej odvisnost

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

Dirac-ov impulz

Dirac-ov impulz (delta impulz) nastane iz enotinega pravokotnega impulza, če limitiramo a proti nič (slika 2.26). Takrat gre $A = \frac{1}{a}$ čez vse meje.

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



Slika 2.26: Dirac-ov impulz

Bistveno je torej, da je ploščina Dirac-ovega impulza enaka 1.

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

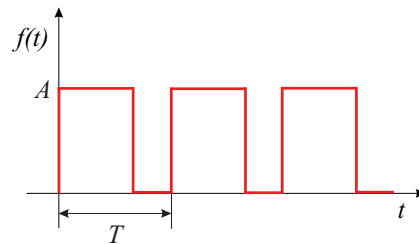
Dirac-ov impulz je povsem teoretska forma, vendar ima zaradi nekaterih ugodnih lastnosti pomembno mesto pri analizi in načrtovanju dinamičnih sistemov. Z nekaj matematike pa lahko pokažemo, da dobimo Dirac-ov impulz tudi z odvajanjem stopničaste funkcije.

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

Vlak impulzov

Iz pravokotnega impulza lahko zgradimo tudi periodično funkcijo, ki jo imenujemo vlak impulzov (slika 2.27). Veljati mora seveda osnovna zahteva

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



Slika 2.27: Vlak impulzov

2.5.6 Uvod v spektralno analizo

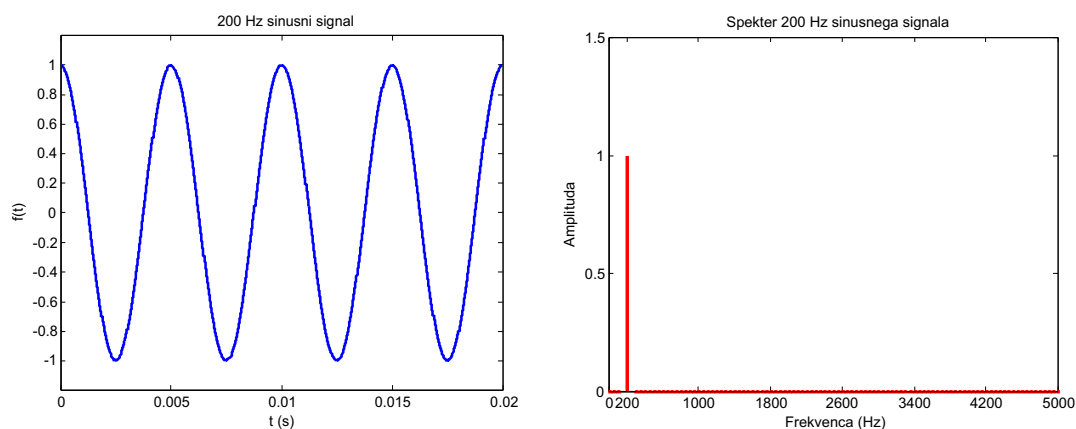
Do sedaj smo signale obravnavali v časovnem prostoru, kar pomeni, da smo jih predstavljali kot časovne funkcije. Spektralna analiza pa pove, iz katerih frekvenc je sestavljen nek signal, torej iz kakšnih trigonometričnih signalov bi nek signal lahko sestavili (Mlakar, 2002, Cha, Molinder, 2006). Spektar je zlasti enostaven pri periodičnih časovnih signalih, saj je diskreten (črtast). Če je frekvenca periodičnega signala ω_0 , potem spektar v splošnem vsebuje enosmerno komponento in frekvence $\omega_0, 2\omega_0, 3\omega_0, \dots$, torej osnovno harmonsko in višje harmonske. Slika 2.28 prikazuje 200 Hz sinusni signal

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

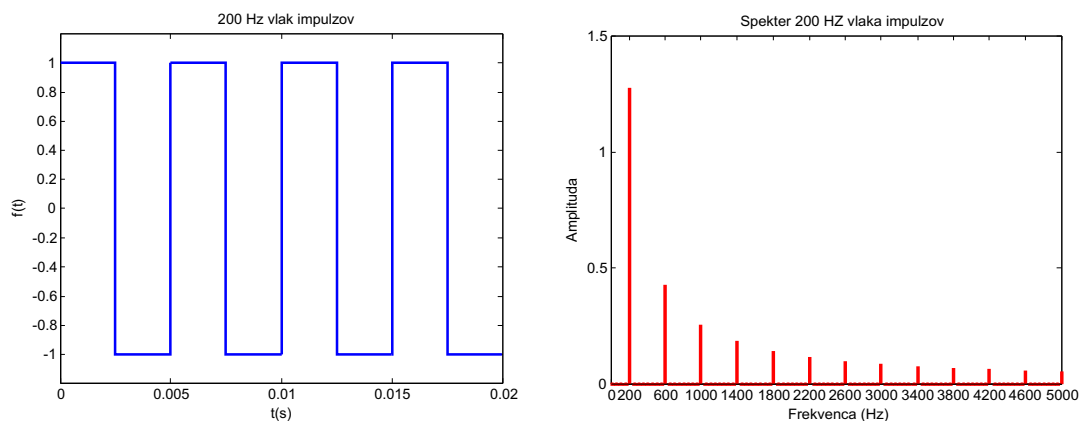
in njegov spektar. Razvidna je prisotnost ene spektralne komponente frekvence 200 Hz in amplitude 1.

Slika 2.29 prikazuje periodični vlak impulzov frekvence 200 Hz in njegov spektar.

Vlak impulzov se spreminja med +1 in -1 s periodo $T = 0.005s$. Spektar vsebuje komponente pri 200 Hz, 600 Hz, 1000 Hz, 1400 Hz, ... Najmočnejša je torej frekvenca 200 Hz, nato 3 harmonska 600 Hz, pa peta harmonska 1000 Hz itd. Spektar torej vsebuje le lihe harmonske komponente.



Slika 2.28: 200Hz signal in njegov spekter



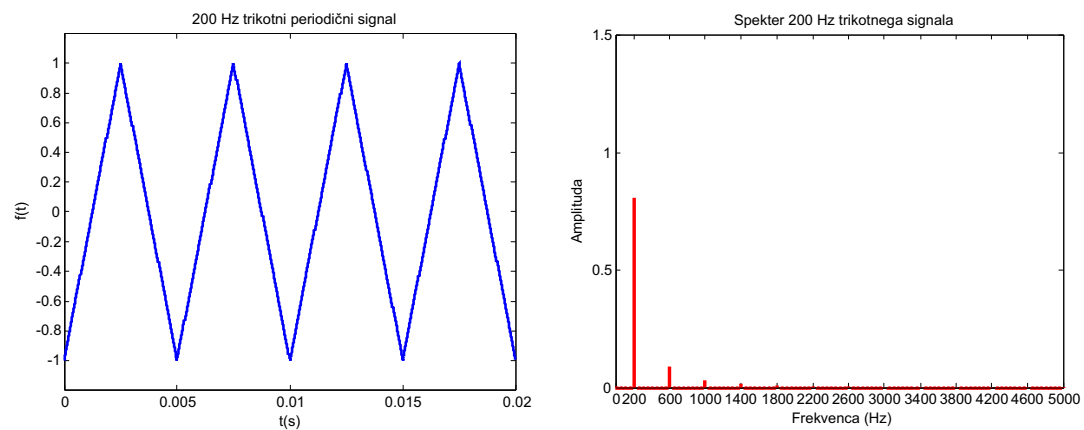
Slika 2.29: 200 Hz vlak impulzov in spekter

Slika 2.30 pa prikazuje trikotni periodični signal frekvence 200Hz in njegov spekter.

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

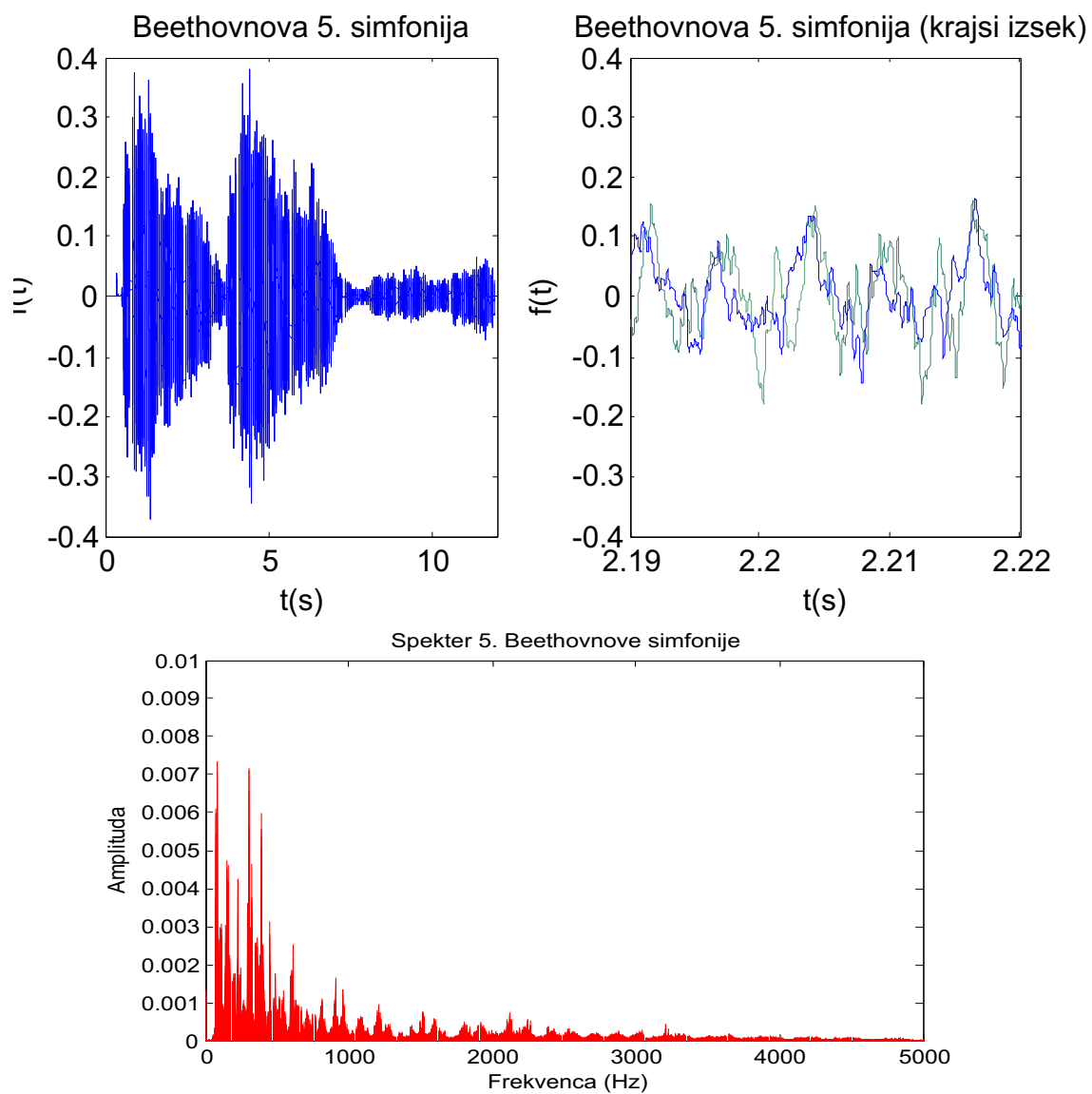
Slika 2.31 pa prikazuje dvo-kanalni audio signal, ki je vzorčen s 44.1 kHz , kar zagotavlja ustrezno CD kvaliteto in njegov frekvenčni spekter. Oko težko razbere, kaj vsebuje časovni signal. Če pa ga pretvorimo v akustični signal, tako da ga pošljemo na zvočnik, pa prepoznamo Beethovnovo 5. simfonijo.

Signale lahko torej predstavljamo in analiziramo na različne načine: v časovnem prostoru, v frekvenčnem prostoru, kot akustične signale, ... Včasih je primerne-



Slika 2.30: 200 Hz trikotni periodični signal in spekter

jša neka predstavitev, včasih pa kaka druga.

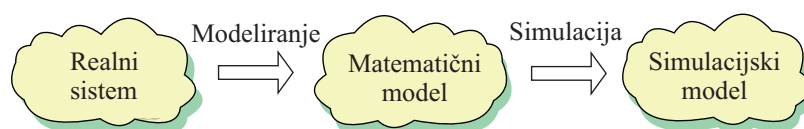


Slika 2.31: Dvokanalni audio signal - 12 s opazovanja in 30 ms izsek, spodaj frekvenčni spekter

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.

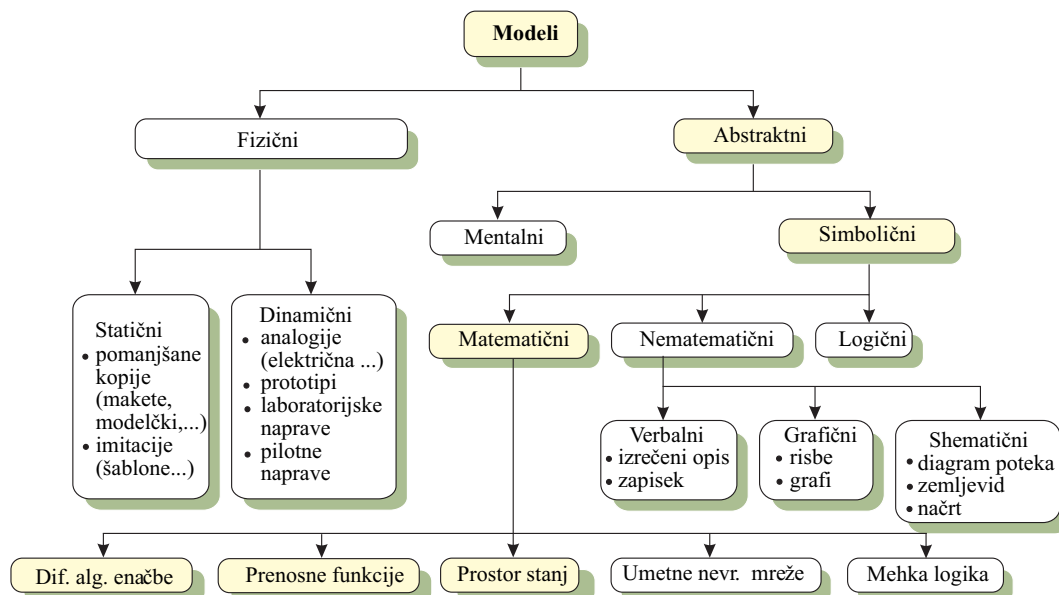
Za uspešno uporabo modeliranja moramo imeti v mislih naslednje najpomembnejše zakonitosti:

- 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

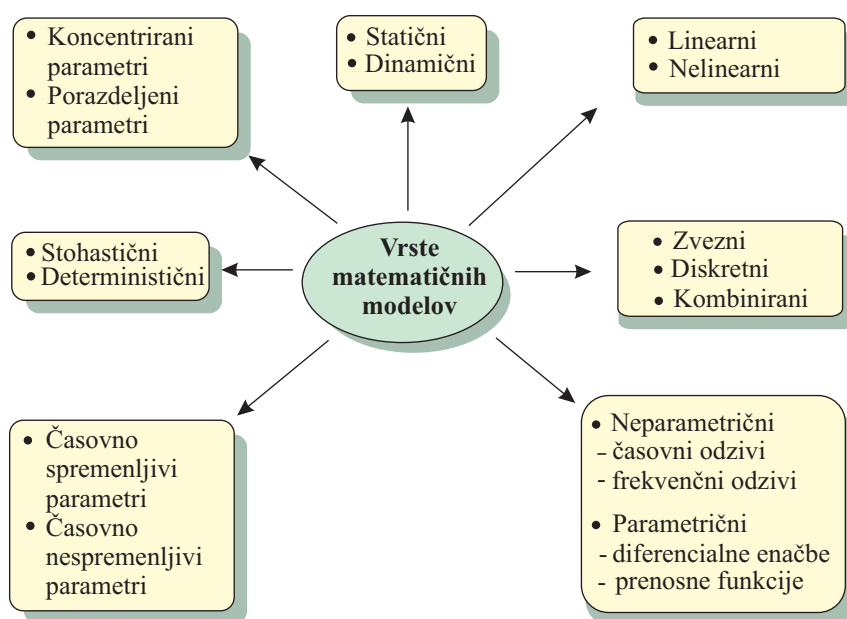
Poznamo seveda veliko različnih vrst modelov. Eno od možnih delitev modelov na različne vrste prikazuje slika 3.2.



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:

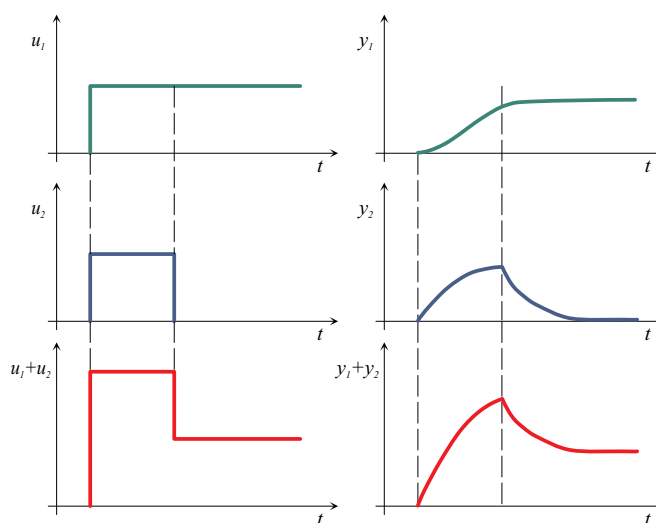
$$\begin{aligned} y(t) &= 2u(t) \\ \dot{y}(t) + 3y(t) &= 2u(t) \end{aligned}$$

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

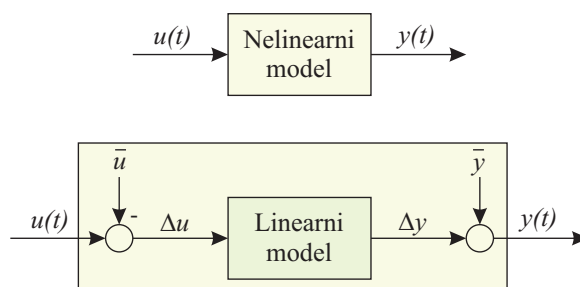
Že pri najenostavnejši nelinearni relaciji $y = u^2$ pa vidimo, da zakon superpozicije ne velja, saj $(u_1 + u_2)^2$ ni enako $u_1^2 + u_2^2$.

Sistem, ki je sestavljen izključno iz linearnih sklopov oz. podsystemov, 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

$$u(t) = \bar{u} + \Delta u(t)$$

izhodni signal pa

$$y(t) = \bar{y} + \Delta y(t)$$

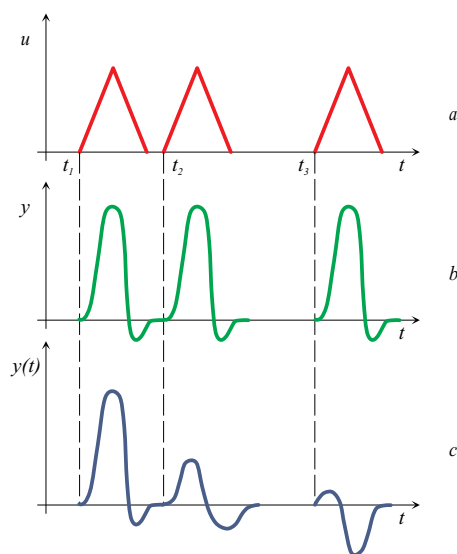
$\bar{u}$ in $\bar{y}$ sta delovni (ustaljeni) vrednosti vhodnega in izhodnega signala, $\Delta u(t)$ in $\Delta y(t)$ pa sta majhni spremembi. Pri uporabi linearnega modela moramo zato od operativnega vhodnega signala $u(t)$ odšteti delovno vrednost $\bar{u}$, da dobimo signal Δu , ki je vhodni signal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

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

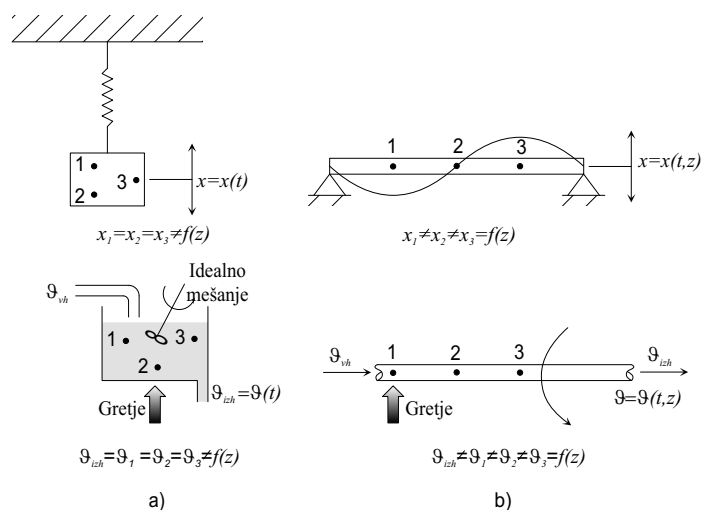
Procese, pri katerih je sprememba procesnih veličin neodvisna od prostorskih koordinat (torej je odvisna največkrat le od časa) imenujemo procese s koncentriranimi parametri. Kjer pa procesne spremenljivke v istem trenutku niso enake po celotnem prostoru, govorimo o procesih s porazdeljenimi parametri.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

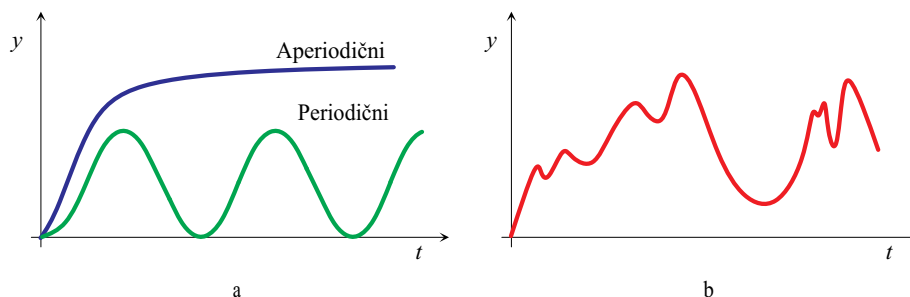
Laplace-ovo transformacijo. Pri diskretnih modelih lahko signali zavzamejo le določene vrednosti iz definicijskega območja (diskretni signali) in/ali iz zaloge vrednosti (digitalni signali). Obravnava temelji na teoriji vzorčenja, diferenčnih enačbah in z-transformaciji. Opisani diskretni modeli so konceptualno 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

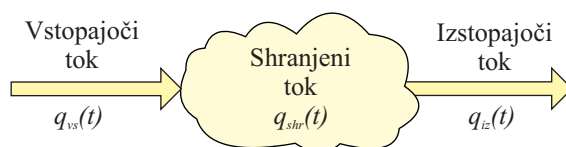
- ohranitvi mase,
- energije in
- gibalne količine.

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

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

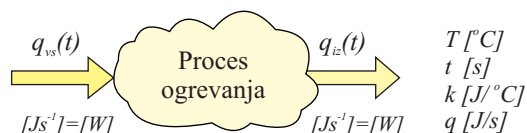


Slika 3.9: Teoretično modeliranje

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 $\frac{dT(t)}{dt}$.

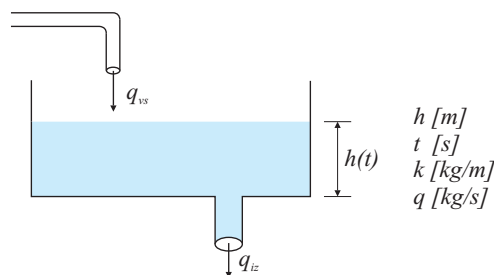
$$k \frac{dT(t)}{dt} = q_{vs}(t) - q_{iz}(t) \quad (3.2)$$

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 $\frac{dh(t)}{dt}$.



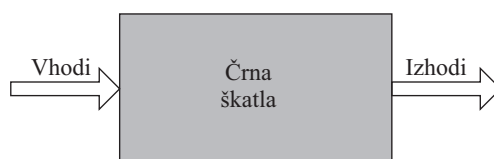
Slika 3.11: Hidravlični sistem

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

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).



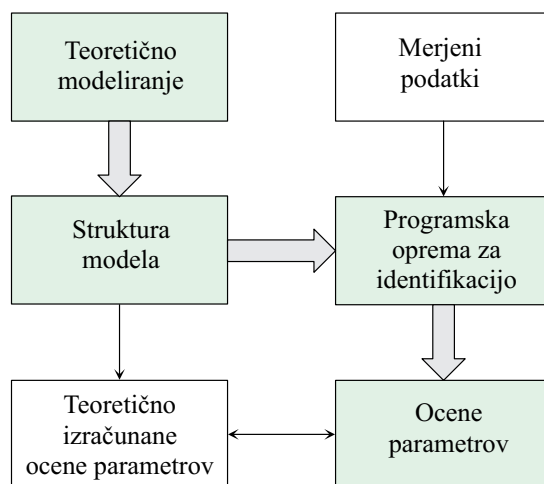
Slika 3.12: Eksperimentalno modeliranje

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 vzbujaajo 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

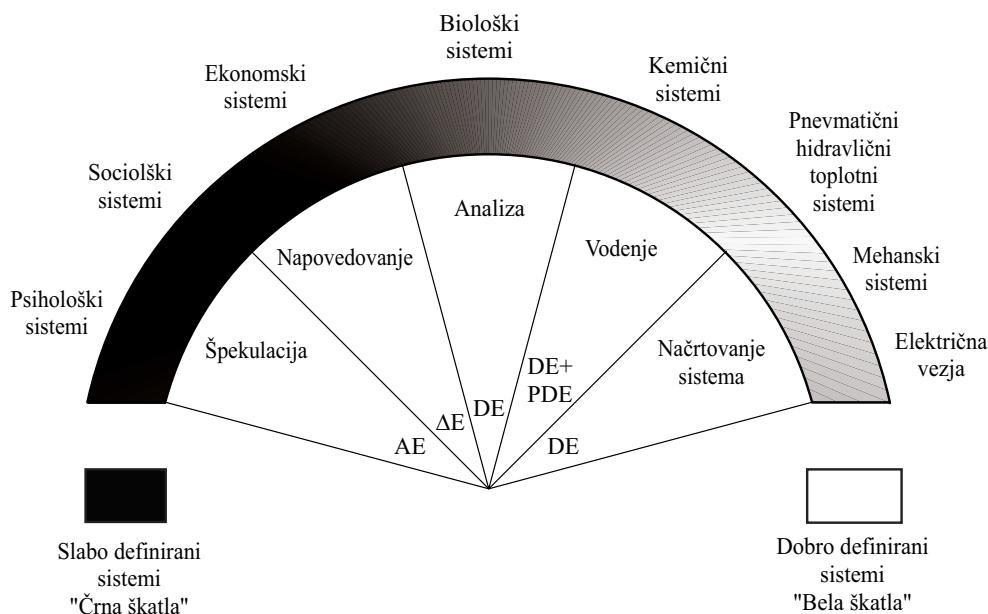
V praksi največkrat uporabljamo kombinirano modeliranje, ki skuša izkoristiti dobre lastnosti omenjenih pristopov (slika 3.13). Pri tem strukturo modela v glavnem definiramo s pomočjo teoretičnega pristopa, medtem ko nekatere parametre lahko ocenimo s pomočjo eksperimentalnega modeliranja. Poglavitna 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 odebeltene 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.

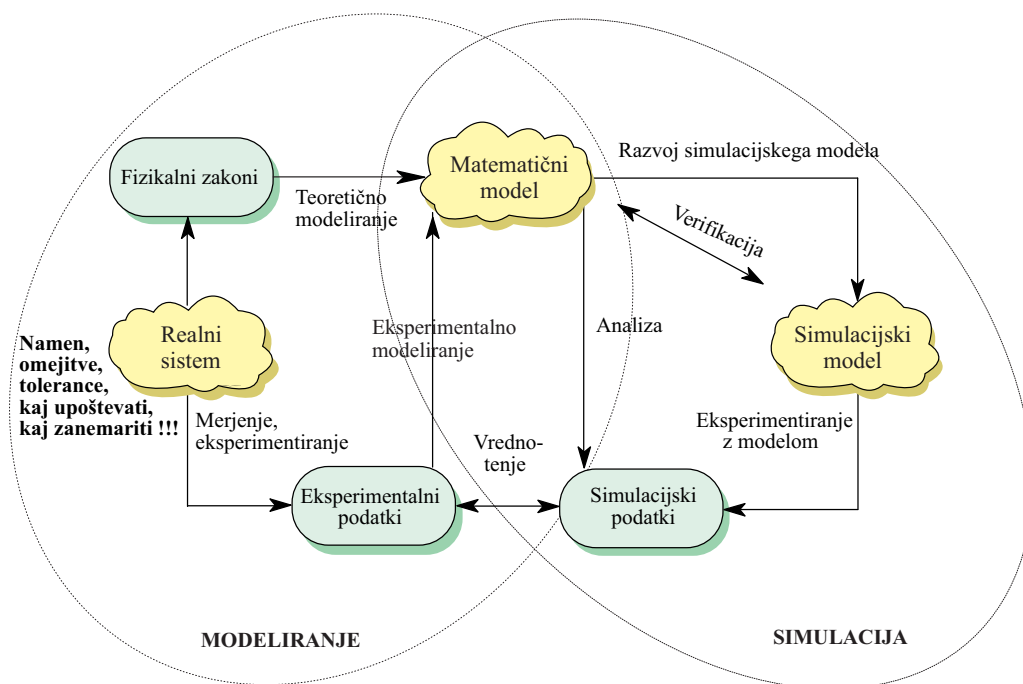


Slika 3.14: Spekter področij, uporabe, namenov in metodologij pri matematičnem modeliranju in simulaciji (AE, ΔE , DE in PDE - algebrajske, diferenčne, diferencialne, parcialne diferencialne enačbe)

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 želenega 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. Prikaže 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 (atribute), 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 si 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).

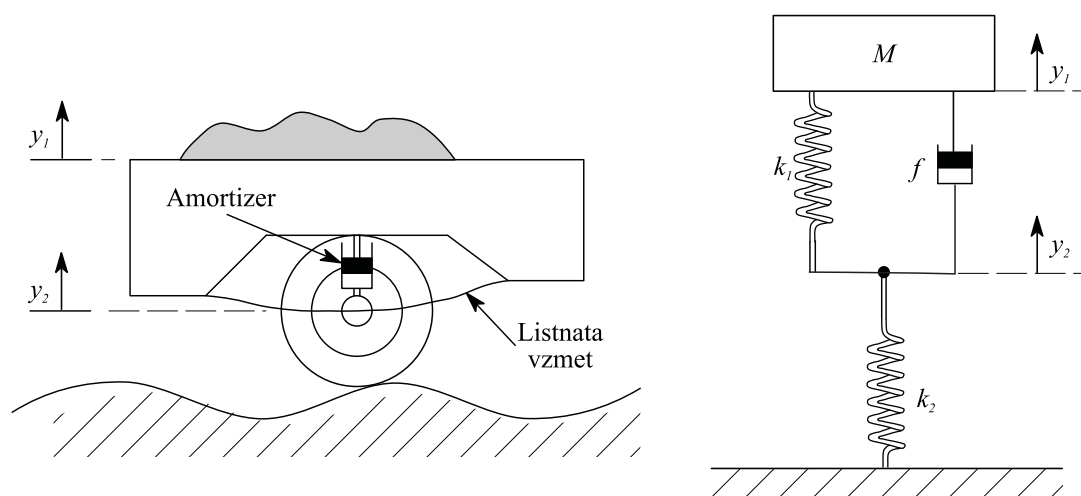
Da bo povezava modeliranja in simulacije dovolj poudarjena, bomo podali tri primere, v katerih bo poudarek najprej na modeliranju, v naslednjih poglavjih pa bomo te modele uporabljali za prikaz simulacijskih metod in orodij.

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 raztegnemo (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}].

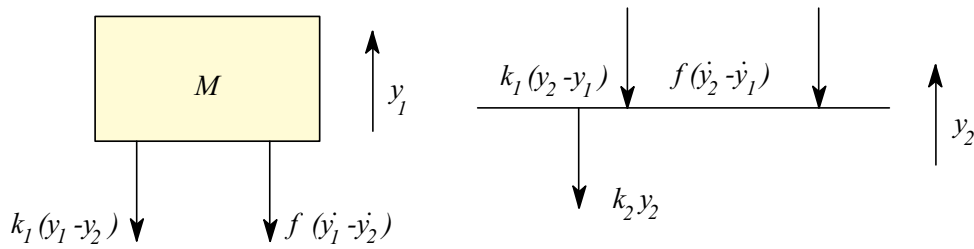
Silo, ki je potrebna, da premaknemo en konec viskoznega dušilnika s hitrostjo v relativno glede na drugi konec, podaja zveza

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

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

Tako se moramo odločiti le še o predznaku sil, kar najlažje naredimo s pomočjo diagrama sil. Pri tem se dogovorimo, da so sile v smeri predpostavljenega 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 vzbujujanju sistema. Kot vemo, je sistem lahko vzbujaan 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:

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

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

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

□

Primer 3.2 Modeliranje regulacije gretja prostora

Oglejmo si še možnost "intuitivnega" pristopa k modeliranju gretja prostora z električnim grelom, ki ga prižiga ali ugaša termostatski regulator. 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:

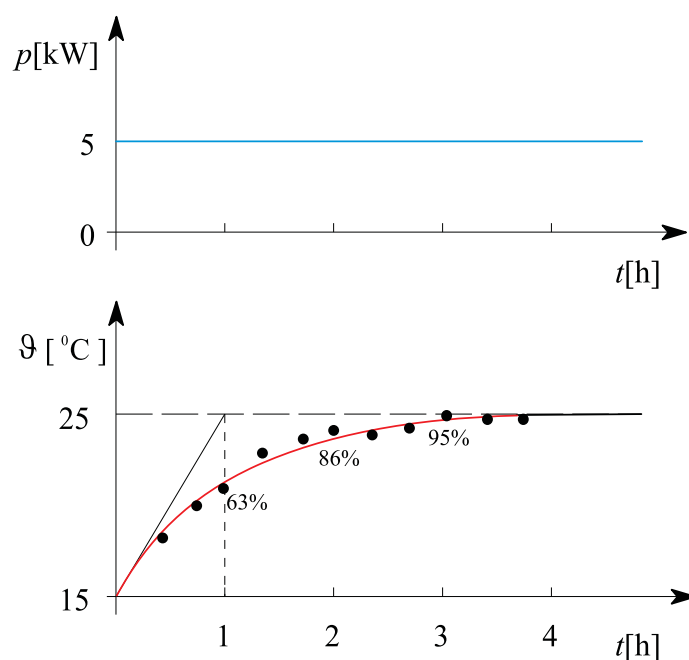
- moči grelnega telesa,
- histereze termostatskega regulatorja in
- zakasnitev, ki so prisotne v procesu.

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 postavitev 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 povratnozančni 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 zaprtozančnem 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 3.18.

Kot vidimo na sliki 3.18, 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 lahko predpostavimo, da je odvod (sprememba) temperature proporcionalen razliki gretja in izgub, kar lahko zapišemo v obliki naslednje diferencialne enačbe:



Slika 3.18: Gretje prostora ob vklopu grela (p ...moč gretja, ϑ ...temperatura v prostoru). Pike pomenijo merjene vrednosti, krivulja pa odziv modela.

$$k_1 \dot{\vartheta} = p - k_2(\vartheta - \vartheta_e) \quad (3.15)$$

Če enačbo (3.2) delimo s konstanto k_1 , namesto konstant k_1 in k_2 pa vpeljemo časovno konstanto T in ojačenje k , dobimo diferencialno enačbo prvega reda

$$\dot{\vartheta} + \frac{1}{T}(\vartheta - \vartheta_e) = \frac{k}{T}p \quad (3.16)$$

kjer je pomen oznak naslednji:

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

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

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

k - ojačenje sistema prvega reda,

T - časovna konstanta sistema prvega reda.

Tako dobljeni linearni model dobro opisuje realni proces za temperaturni interval $15^{\circ}\text{C} \leq \vartheta \leq 25^{\circ}\text{C}$ 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

$$\vartheta_w = \vartheta - \vartheta_e \quad (3.17)$$

kar daje končno obliko modela ($\dot{\vartheta} = \dot{\vartheta}_w$)

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

Model, ki ga podaja enačba (3.18), lahko zapišemo še v obliki prenosne funkcije

$$\frac{\theta_w(s)}{P(s)} = \frac{k}{Ts + 1} \quad (3.19)$$

Postavitev enačbe (3.18) 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 3.18). 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)

$$k = \frac{\Delta\vartheta}{\Delta p} = \frac{25 - 15}{5 - 0} = 2^{\circ}\text{C/kW} \quad (3.20)$$

Če dobljene vrednosti vstavimo v enačbo (3.18), dobimo model ogrevanja prostora

$$\dot{\vartheta}_w + \vartheta_w = 2p \quad (3.21)$$

oz. v obliki prenosne funkcije

$$\frac{\theta_w(s)}{P(s)} = \frac{2}{s+1} \quad (3.22)$$

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 bolje 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 (3.18) spremeniti v zapis z dvema enačbama

$$\begin{aligned} \dot{\vartheta}_w + \frac{1}{T}\vartheta_w &= \frac{k}{T}p_d \\ p_d(t) &= p(t - T_d) \end{aligned} \quad (3.23)$$

kjer je T_d mrtvi čas. Prenosna funkcija takega sistema je

$$\frac{\theta_w(s)}{P(s)} = \frac{k}{Ts+1} e^{-T_d s} \quad (3.24)$$

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.

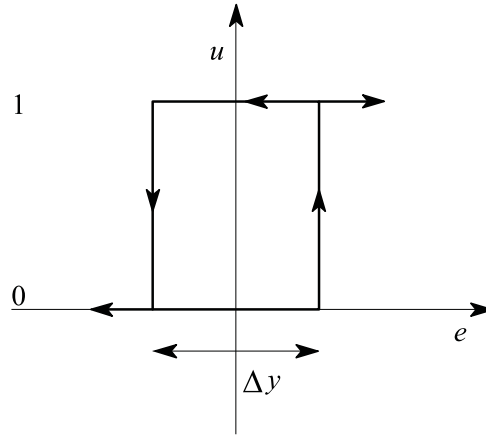
Zgradimo bločni diagram regulacije gretja prostora. Želeno temperaturo prostora (ϑ_r) nastavimo na termostatskem regulatorju. Vhod v slednjega je tudi temperatura v prostoru ϑ . Termostatski regulator ustvari razliko

$$e = \vartheta_r - \vartheta \quad (3.25)$$

iz katere preko histerezne karakteristike

$$u(t) = \begin{cases} 0, & e < -\frac{\Delta y}{2} \\ 1, & e > \frac{\Delta y}{2} \\ u(t - \Delta t), & -\frac{\Delta y}{2} \leq e \leq \frac{\Delta y}{2} \end{cases} \quad (3.26)$$

določi regulirni signal. Ustrezno zakonitost prikazuje slika 3.19. Pri tem je $u(t - \Delta t)$ pretekla vrednost regulirnega signala (seveda 0 ali 1, pri tem je Δt poljubno majhen).



Slika 3.19: Histereza termostata

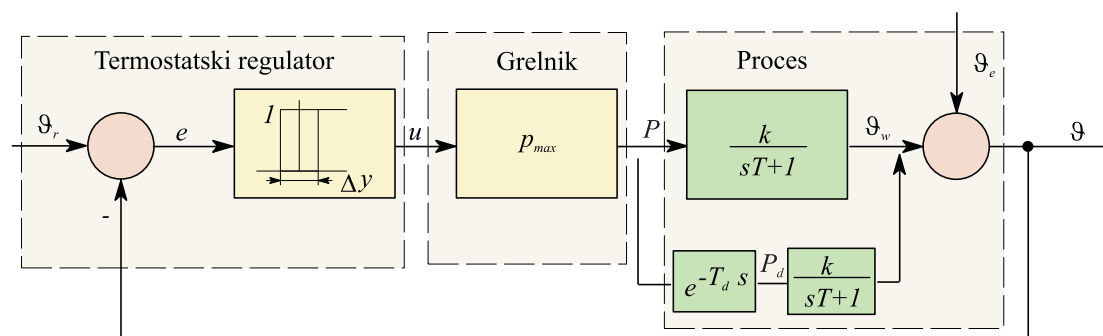
Regulirni signal u vklaplja in izklaplja grelec (npr. preko releja, kontaktorja). Le-tega lahko modeliramo kar z nekim konstantnim ojačenjem p_{max} , oz. z enačbo

$$p = p_{max}u \quad (3.27)$$

Dobljeni signal p predstavlja vhod v model procesa (enačba (3.18) za $T_d = 0$ in enačbi (3.23) 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)

$$\vartheta = \vartheta_w + \vartheta_e \quad (3.28)$$

Bločni diagram regulacijske zanke pri gretju prostora tako prikazuje slika 3.20. V gradniku za proces sta navedeni obe možnosti: model prvega reda in model prvega reda z mrtvim časom.



Slika 3.20: Bločni diagram regulacije temperature prostora

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

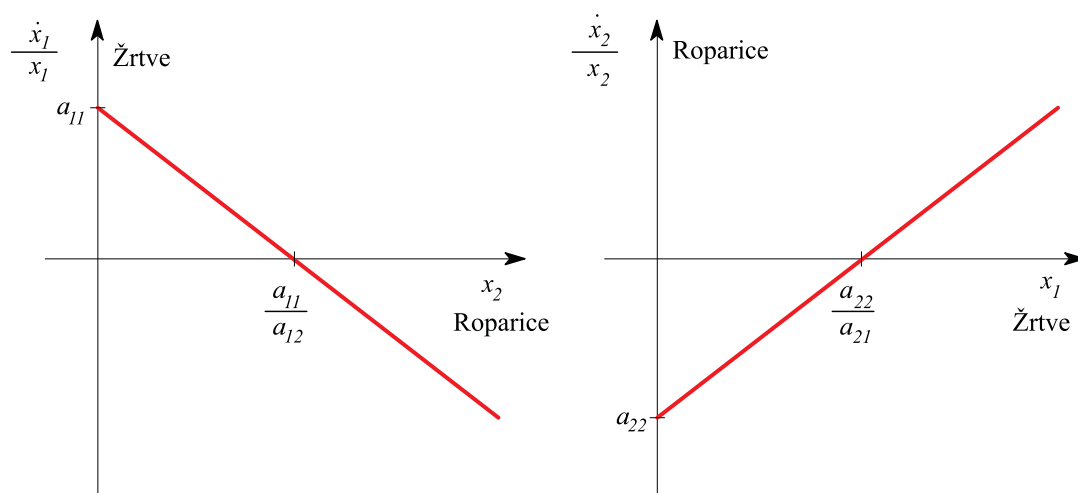
- določitev širine histereze termostata, ki daje majhna nihanja temperature ob ne preveč pogostem preklapljanju grela,
- določitev minimalne moči grela, ki še zagotavlja zadovoljivo regulacijo (sprejemljivo hitrost prilagajanja temperature novi želeni vrednosti, neobčutljivost na motnje itd.),
- študij vpliva motenj na regulirano temperaturo (možno je enostavno simulirati različne motnje na vходу 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 logaritmčna rast ene odvisna le od druge populacije. Logaritmčna rast populacije roparic je sorazmerna populaciji žrtev in logaritmčna rast populacije žrtev je sorazmerna populaciji roparic. Ob predpostavki, da je sorazmernost linearna, dobimo odvisnosti, kot jih prikazuje slika 3.21. 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.



Slika 3.21: Logaritmični rasti populacij žrtev in roparjev

S pomočjo slike 3.21 lahko napišemo naslednji matematični model

$$\begin{aligned}\frac{\dot{x}_1}{x_1} &= a_{11} - a_{12}x_2 \\ \frac{\dot{x}_2}{x_2} &= a_{21}x_1 - a_{22}\end{aligned}\tag{3.29}$$

ali

$$\begin{aligned}\dot{x}_1 &= (a_{11} - a_{12}x_2)x_1 \\ \dot{x}_2 &= (a_{21}x_1 - a_{22})x_2\end{aligned}\tag{3.30}$$

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 (3.30) 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:

$$\begin{aligned} a_{11} &= \left. \frac{\dot{x}_1'}{x_1} \right|_{a_{12}=0} \approx \frac{\Delta x_1' / \Delta t}{x_1} = \frac{10}{2} = 5 \\ a_{22} &= \left. -\frac{\dot{x}_2'}{x_2} \right|_{a_{21}=0} \approx \frac{\Delta x_2' / \Delta t}{x_2} = -\frac{-1}{5} = 0.2 \end{aligned} \quad (3.31)$$

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 km², kjer je povprečno število zajcev $\bar{x}_1 = 500$ in lisic $\bar{x}_2 = 100$, predpostavka 2 daje

$$a_{12} = \left. -\frac{\dot{x}_1''}{\bar{x}_1 \bar{x}_2} \right|_{a_{11}=0} \approx \frac{\Delta x_1'' / \Delta t}{\bar{x}_1 \bar{x}_2} = -\frac{25 \bar{x}_2}{\bar{x}_1 \bar{x}_2} = \frac{25}{500} = 0.05 \quad (3.32)$$

predpostavka 4 pa

$$a_{21} = \left. \frac{\dot{x}_2''}{\bar{x}_1 \bar{x}_2} \right|_{a_{22}=0} \approx \frac{\Delta x_2'' / \Delta t}{\bar{x}_1 \bar{x}_2} = \frac{\bar{x}_1 / 25}{\bar{x}_1 \bar{x}_2} = \frac{1}{25 \bar{x}_2} = \frac{1}{2500} = 0.0004 \quad (3.33)$$

Tako smo prišli do matematičnega modela za obravnavani primer:

$$\begin{aligned}\dot{x}_1 &= 5x_1 - 0.05x_1x_2 \\ \dot{x}_2 &= 0.0004x_1x_2 - 0.2x_2\end{aligned}\tag{3.34}$$

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

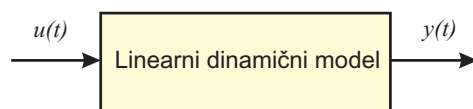
4.

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 algebrske operacije. Zato lahko s pomočjo Laplace-ove transformacije prevedemo diferencialne enačbe v algebrske. 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 transforma v časovno funkcijo, je inverzna Laplace-ova transformacija.

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

Najpogostejše 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

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

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

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 ustrezno preureditvijo pridemo do Laplace-ovega transforma 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

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

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

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

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

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

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

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

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

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

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

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

□

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

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

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

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

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

□

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 ustrezni kvocient, dobimo prenosno funkcijo

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

Z uporabo prenosnih funkcij je možno predstaviti dinamiko sistemov s pomočjo algebraskih 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 vhom 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.

Razen polinomskega zapisa prenosne funkcije (enačba (4.43)) uporabljamo tudi faktorizirano obliko iz katere so razvidni poli ($s = p_1, p_2, \dots, p_n$) in ničle ($s = z_1, z_2, \dots, z_m$)

$$G(s) = \frac{k(s - z_1)(s - z_2) \cdots (s - z_m)}{(s - p_1)(s - p_2) \cdots (s - p_n)} \quad (4.44)$$

k multiplikativna konstanta. Predpostavljamo, da ni večkratnih polov in ničel. Omenimo naj, da k ni neposredno ojačenje sistema, ki pomeni relacijo med ustaljeno vrednostjo izhodnega in vhodnega signala. Ker je ojačenje

$$K = \lim_{s \rightarrow 0} G(s) \quad (4.45)$$

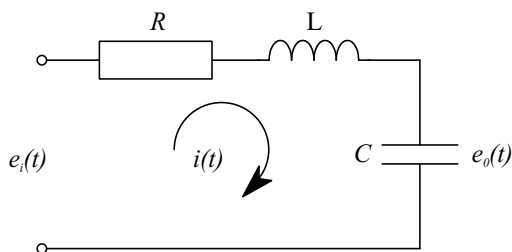
velja $K = \frac{k(-z_1)(-z_2) \cdots (-z_m)}{(-p_1)(-p_2) \cdots (-p_n)}$.

Primer 4.5 Slika 4.2 prikazuje serijsko RLC vezje

Z uporabo prvega Kirchofovega zakona pridemo do enačbe

$$e_i(t) = Ri(t) + L \frac{di(t)}{dt} + \frac{1}{C} \int i(t) dt \quad (4.46)$$

Z Laplace-ovo transformacijo enačbe (4.46) dobimo ob upoštevanju ničelnih začetnih pogojev enačbo



Slika 4.2: Serijsko RLC vezje

$$E_i(s) = \left(R + Ls + \frac{1}{Cs} \right) I(s) \quad (4.47)$$

Če nas zanima prenosna funkcija med tokom $i(t)$ ter vhodno napetostjo $e_i(t)$, naredimo ustrezno deljenje v enačbi (4.47)

$$G_1(s) = \frac{I(s)}{E_i(s)} = \frac{1}{R + Ls + \frac{1}{Cs}} = \frac{Cs}{1 + RCs + LCs^2} = \frac{\frac{1}{L}s}{s^2 + \frac{R}{L}s + \frac{1}{LC}} \quad (4.48)$$

Če za izhodno veličino izberemo napetost na kondenzatorju, velja

$$e_o(t) = \frac{1}{C} \int i \, dt \quad (4.49)$$

$$E_o(s) = \frac{1}{Cs} I(s) \quad (4.50)$$

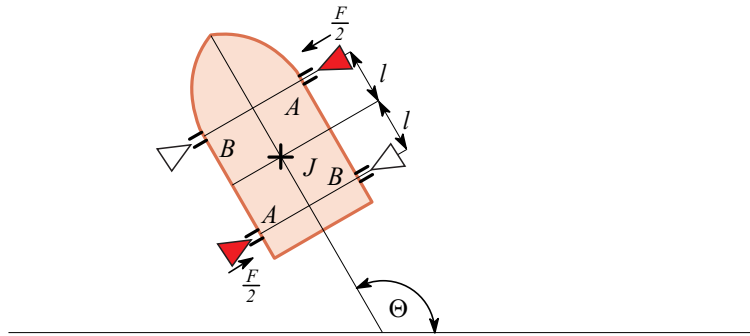
oz.

$$G_2(s) = \frac{E_o(s)}{E_i(s)} = \frac{E_o(s)}{I(s)} \cdot \frac{I(s)}{E_i(s)} = \frac{\frac{1}{LC}}{s^2 + \frac{R}{L}s + \frac{1}{LC}} \quad (4.51)$$

□

Primer 4.6 Slika 4.3 prikazuje problem regulacije smeri leta nekega 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 B ter B



Slika 4.3: Regulacija smeri leta satelita

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 vzbujalnim momentom $T(t)$, oz. ustrezna prenosna funkcija. Z uporabo drugega Newtonovega zakona ob zanemaritvi dušenja pridemo do diferencialne enačbe

$$J \frac{d^2\Theta(t)}{dt^2} = T(t) \quad (4.52)$$

in z Laplace-ovo transformacijo do enačbe

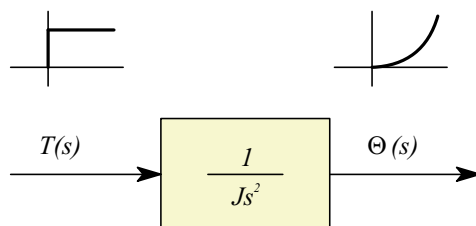
$$Js^2\Theta(s) = T(s) \quad (4.53)$$

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

$$G(s) = \frac{\Theta(s)}{T(s)} = \frac{1}{Js^2} \quad (4.54)$$

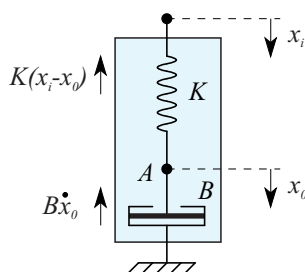
Ustrezno bločno shemo prikazuje slika 4.4. Prikazana sta tudi tipična vhodni in izhodni signal.

□



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

$$\sum F_i = 0 \quad (4.55)$$

oz.

$$-B \dot{x}_o - K(x_o - x_i) = 0 \quad (4.56)$$

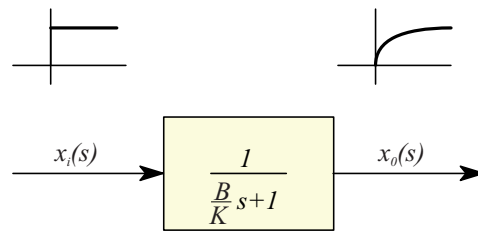
Z ustrezno Laplace-ovo transformacijo pridemo do enačb

$$B s X_o(s) + K X_o(s) = K X_i(s) \quad (4.57)$$

oz.

$$\frac{X_o(s)}{X_i(s)} = \frac{1}{\frac{B}{K}s + 1} \quad (4.58)$$

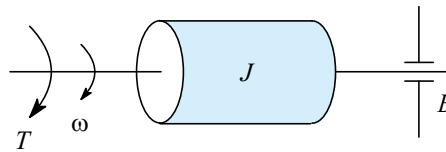
Bločno shemo s tipičnim vhodnim in izhodnim signalom prikazuje slika 4.6.



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

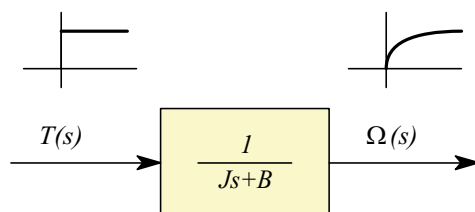
$$J \frac{d\omega(t)}{dt} + B\omega(t) = T(t) \quad (4.59)$$

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

$$\frac{\Omega(s)}{T(s)} = \frac{1}{Js + B} \quad (4.60)$$

Ustrezno bločno shemo prikazuje slika 4.8.

□



Slika 4.8: Bločna shema rotacijskega mehanskega sistema

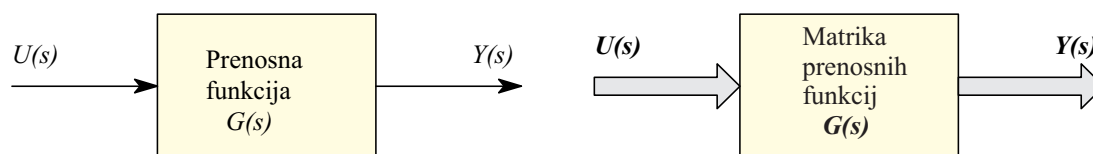
4.5 Bločni diagrami

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 bližje 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 vhodom in izhodom. Često je podan s prenosno funkcijo. Puščici na vходу in izhodu določata smer signala. Signal torej poteka le v smeri puščic.

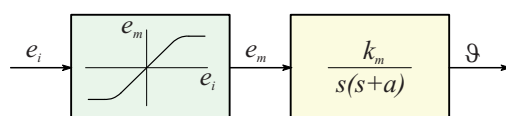
Slika 4.9 prikazuje temeljni element bločnega diagrama - funkcionalni blok. Vhodni ali izhodni signal sta lahko enodimenzionalna (pri univariabilnem sistemu) ali večdimenzionalna (pri multivariabilnem sistemu). Za večdimenzionalne signale je precej v uporabi risanje signalov, kot prikazuje slika 4.9b. Namesto prikazanih



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

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).

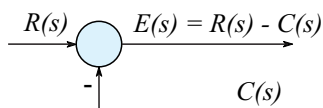
Prav tako uporaba bločnih diagramov ni omejena na linearne sisteme, ampak je povsem splošna. Slika 4.10 prikazuje tipičen nelinearni sistem z ojačevalnikom in rotorsko vzbujanim enosmernim motorjem.



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.

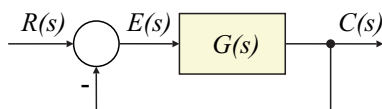


Slika 4.12: Razcepišče

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

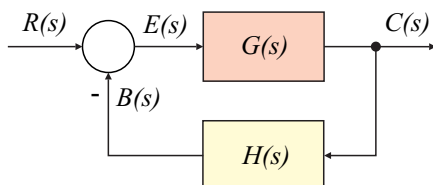
Bločni diagram zaprtozančnega sistema

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 povratnoznančno strukturo prikazuje slika 9.5.



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 9.5 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 $E(s)$:

$$\text{prenosna funkcija direktne veje} = \frac{C(s)}{E(s)} = G(s) \quad (4.61)$$

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

$$\text{odprtozančna prenosna funkcija} = \frac{B(s)}{E(s)} = G(s) H(s) \quad (4.62)$$

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 9.5

$$C(s) = G(s) E(s) \quad (4.63)$$

$$B(s) = H(s) C(s) \quad (4.64)$$

$$E(s) = R(s) - B(s) = R(s) - H(s) C(s) \quad (4.65)$$

Če iz enačb (4.63) in (4.65) izločimo $E(s)$, dobimo

$$C(s) = G(s) [R(s) - H(s) C(s)] \quad (4.66)$$

oziroma

$$\text{zaprtizančna prenosna funkcija} = \frac{C(s)}{R(s)} = \frac{G(s)}{1 + G(s) H(s)} \quad (4.67)$$

S pomočjo zaprtizančne prenosne funkcije izračunamo regulirano veličino s pomočjo enačbe

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

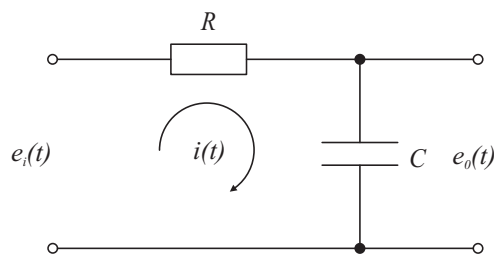
Odziv je torej odvisen od zaprtizančne prenosne funkcije in od vhodnega signala $r(t)$.

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

$$e_o = \frac{Q}{C} = \frac{\int i dt}{C} \quad (4.69)$$

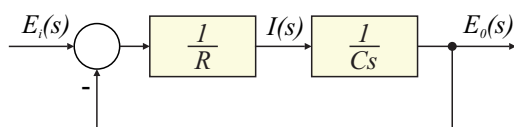
$$i = \frac{e_i - e_o}{R} \quad (4.70)$$

in ustrezna Laplace-ova transformacija

$$E_o(s) = \frac{I(s)}{Cs} \quad (4.71)$$

$$I(s) = \frac{E_i(s) - E_o(s)}{R} \quad (4.72)$$

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

$$i = C \frac{de_o}{dt} \quad (4.73)$$

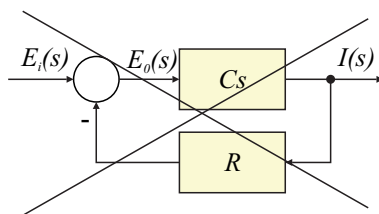
$$e_o = e_i - iR \quad (4.74)$$

oz.

$$I(s) = CsU_C(s) \quad (4.75)$$

$$E_o(s) = E_i(s) - I(s)R \quad (4.76)$$

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 nesprejemljiv. Tako npr. onemogoča numerično ugodno realizacijo (simulacijo) modela na računalniku kakor tudi konsistentno vključitev začetnih pogojev.

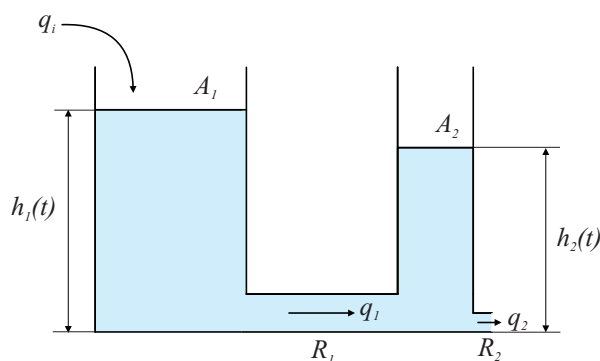
□

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

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

$$A_1 \frac{dh_1(t)}{dt} = q_i(t) - q_1(t) \quad (4.77)$$

Pretok med obema posodama je soracionalen razliki obeh nivojev



Slika 4.18: Hidravlični sistem z dvema nivojskima posodama

$$q_1(t) = \frac{h_1(t) - h_2(t)}{R_1} \quad (4.78)$$

Ravnotežna enačba za drugo posodo je

$$A_2 \frac{d h_2(t)}{dt} = q_1(t) - q_2(t) \quad (4.79)$$

iztok pa je odvisen od nivoja h_2

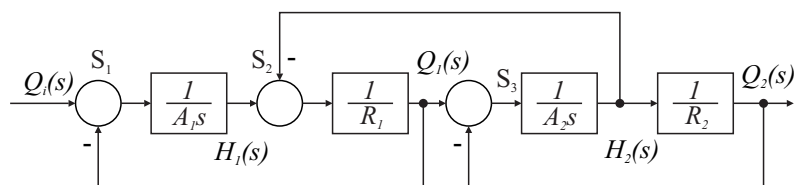
$$q_2(t) = \frac{h_2(t)}{R_2} \quad (4.80)$$

Z Laplace-ovo transformacijo gornjih enačb dobimo

$$\begin{aligned} A_1 s H_1(s) &= Q_i(s) - Q_1(s) \Rightarrow H_1(s) = \frac{1}{A_1 s} [Q_i(s) - Q_1(s)] \\ Q_1(s) &= \frac{H_1(s) - H_2(s)}{R_1} \end{aligned} \quad (4.81)$$

$$\begin{aligned} A_2 s H_2(s) &= Q_1(s) - Q_2(s) \Rightarrow H_2(s) = \frac{1}{A_2 s} [Q_1(s) - Q_2(s)] \\ Q_2(s) &= \frac{H_2(s)}{R_2} \end{aligned} \quad (4.82)$$

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

□

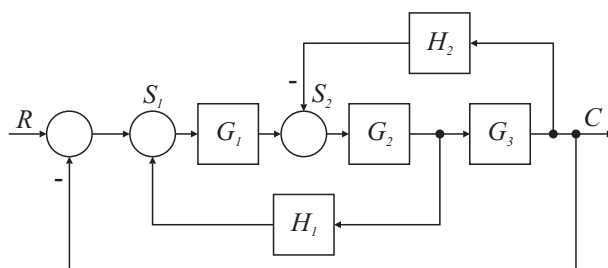
4.5.3 Poenostavljanje bločnih diagramov

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.

Primer 4.11 Poenostavimo bločni diagram na sliki 4.20.

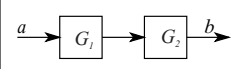
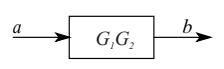
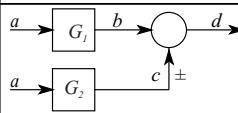
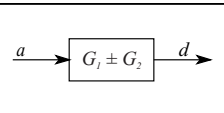
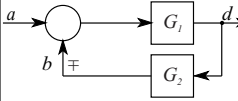
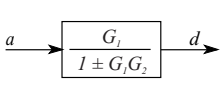
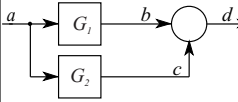
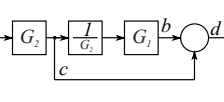
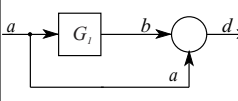
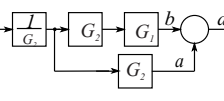
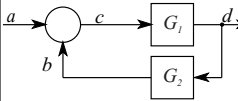
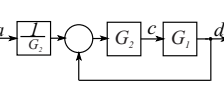
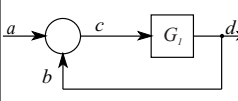
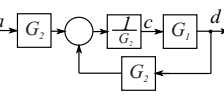
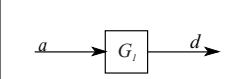
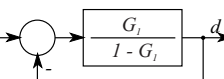


Slika 4.20: Bločni diagram

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

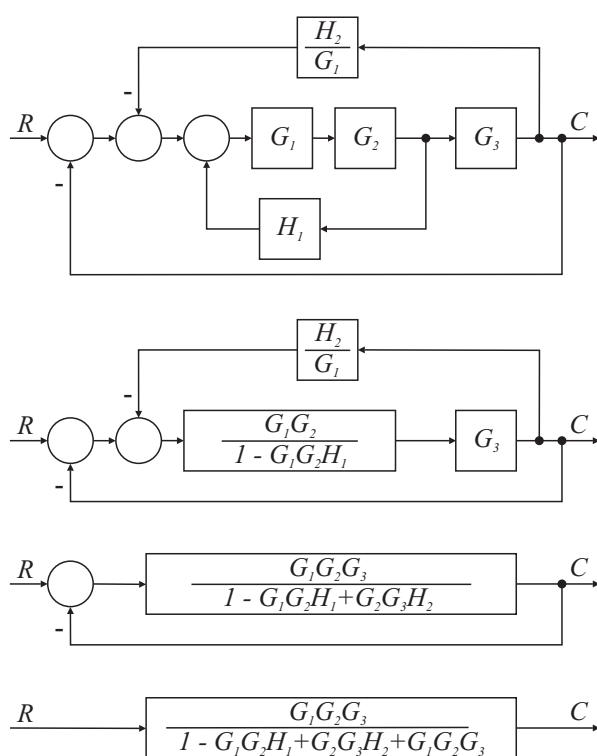
Tabela 4.2: Pravila algebre bločnih shem

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		

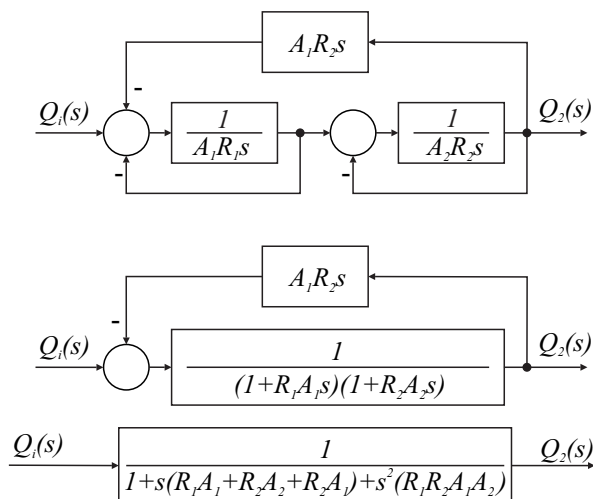
pravila redukcije zaporedne vezave blokov (pravilo 10) ter redukcije zanke (pravilo 12). Postopek prikazuje slika 4.21.

□



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 polinomiški 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 $[0 \text{ } tf]$.

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

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

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

izračuna se 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 `impz` 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

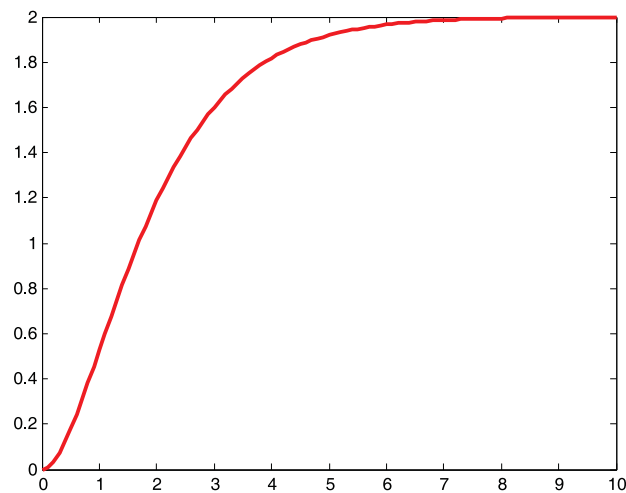
```
G=tf([2],[1 2 1]); tf=10; [y,t]=step(G,tf); plot(t,y)
```

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

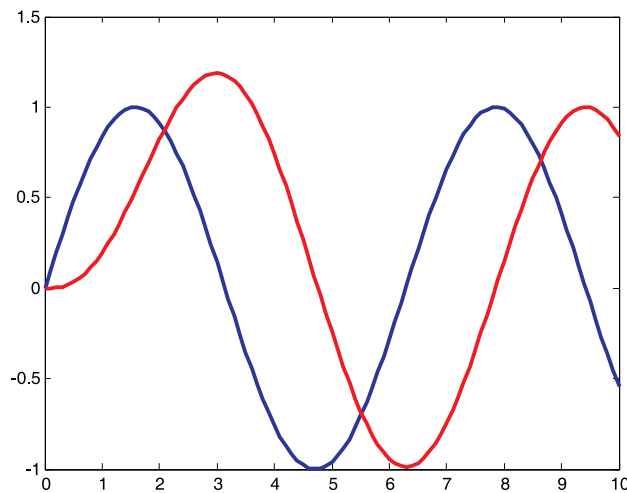
```
G=tf([2],[1 2 1]);t=0:0.1:10; u=sin(t); y=lsim(G,u,t); plot(t,u,t,y)
```

□

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 nastavev parametra `options` se uporablja posebna funkcija `simset`. Če želimo izbrati integracijsko metodo `ode45`, uporabimo ukaz

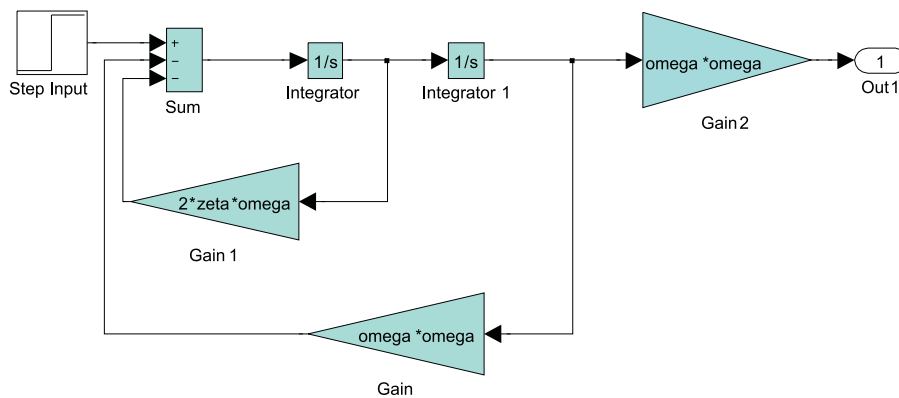
```
options=simset('solver','ode45').
```

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

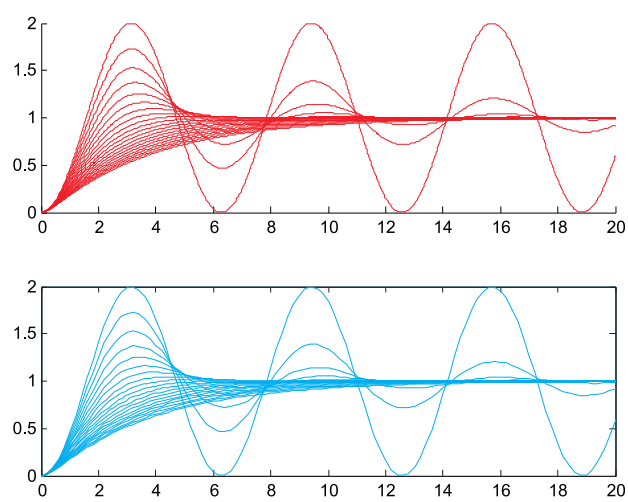
```
omega=1; tf=20; for zeta=0:0.1:2;
    [t,x,y]=sim('secor1',tf);
    Gs=tf(1,[1 2*zeta*omega omega*omega]);
    [y1,t1]=step(Gs,tf);
    subplot(2,1,1);
    hold on
    plot(t,y,'r');
    subplot(2,1,2);
    plot(t1,y1,'b');
    hold on;
end
```

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. Razen tega postopka bomo opisali direktno in implicitno metodo ter več načinov simulacije prenosnih funkcij. Prikazali bomo tudi metodo za simulacijo sistemov z mrtvim časom in način simulacije kompleksnih sistemov. Postopki nas bodo pripeljali do simulacijskih shem, ki so neodvisne od uporabljenega simulacijskega orodja in predstavljajo osnovo pri uporabi simulacijskega orodja. Končno bomo razčlenili tudi koncept digitalne simulacije.

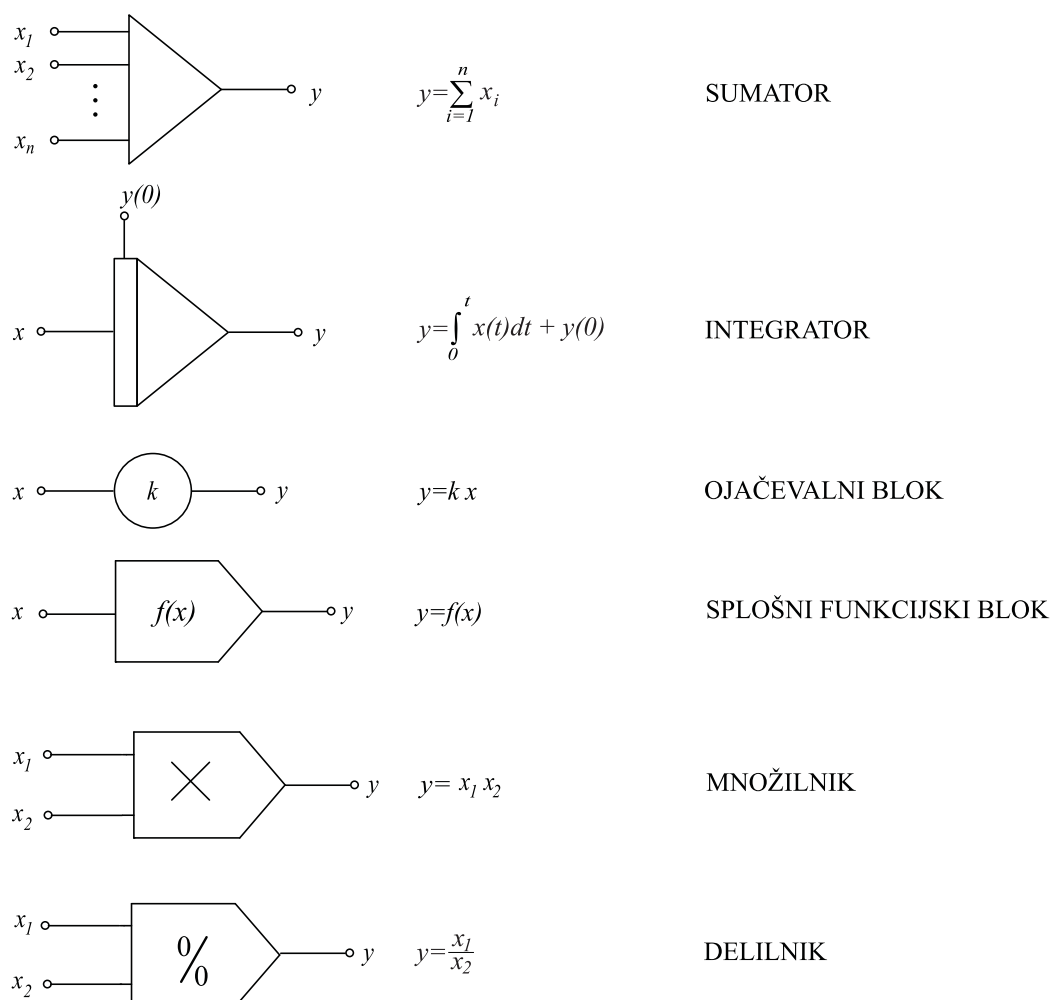
6.1 Simulacijska shema

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:

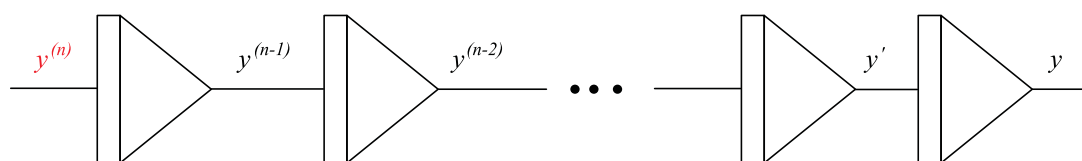
1. korak

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.

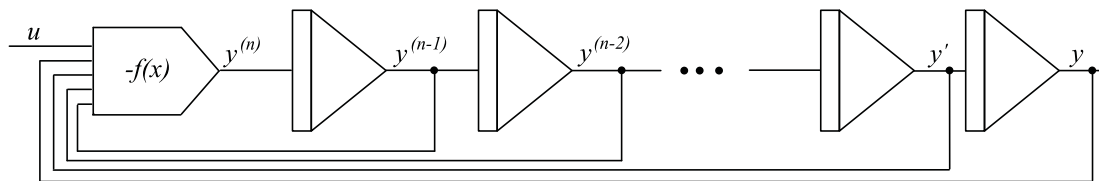


Slika 6.2: 2. korak indirektnega postopka

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 lin-

earnih sistemov bolj smiselno sistem simulirati po konceptu prenosnih funkcij (podpoglavje 6.3).

Uporabnost postopka bomo prikazali na primerih.

Primer 6.1 Temperaturni proces

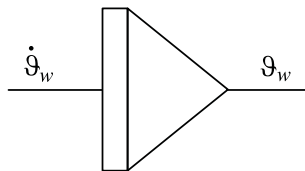
Model temperaturnega procesa je opisan v primeru 3.2. Matematični model opisuje diferencialna enačba

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

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

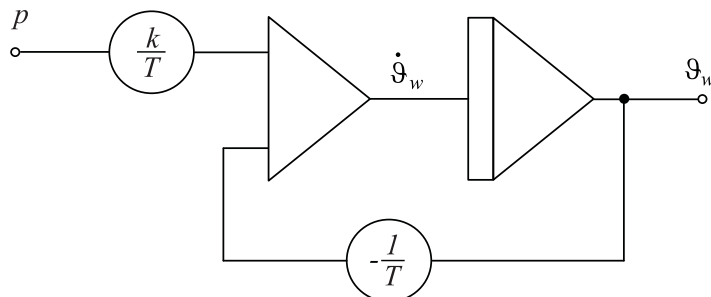
$$\dot{\vartheta}_w = -\frac{1}{T}\vartheta_w + \frac{k}{T}p \quad (6.4)$$

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 blokoma. Potrebna predznaka sta vključena v ojačevalnih blokih. □

Primer 6.2 Avtomobilsko vzmetenje

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

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

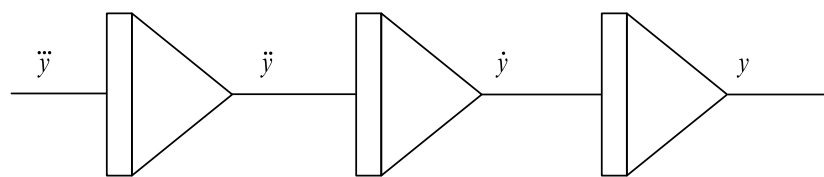
ki jo lahko poenostavimo v obliko

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

1. korak: Preuredimo enačbo (6.6)

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

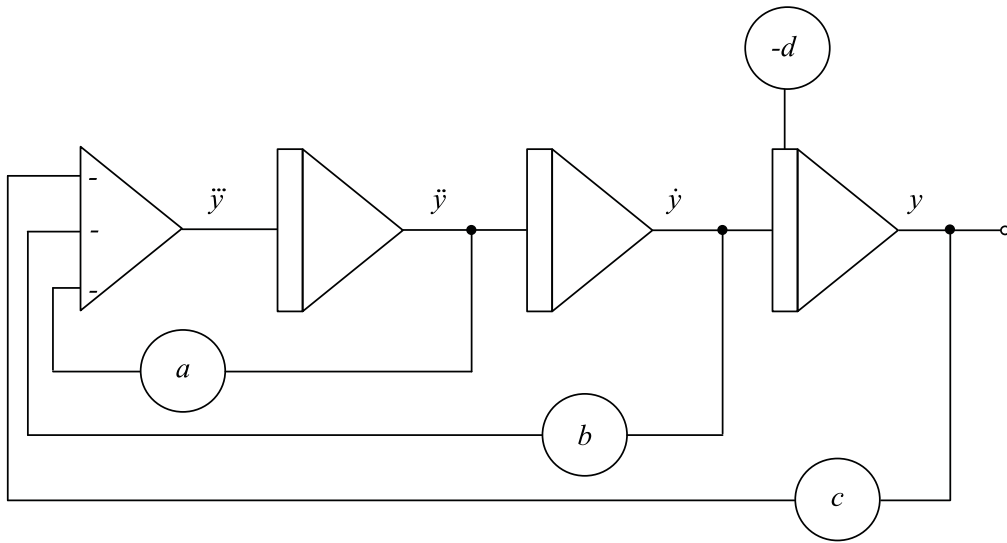
2. korak: Narišemo kaskado treh integratorjev, kar prikazuje slika 6.6



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

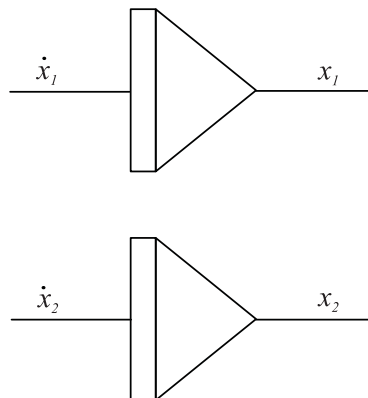


Slika 6.7: Simulacijska shema sistema avtomobilskega vzmetenja

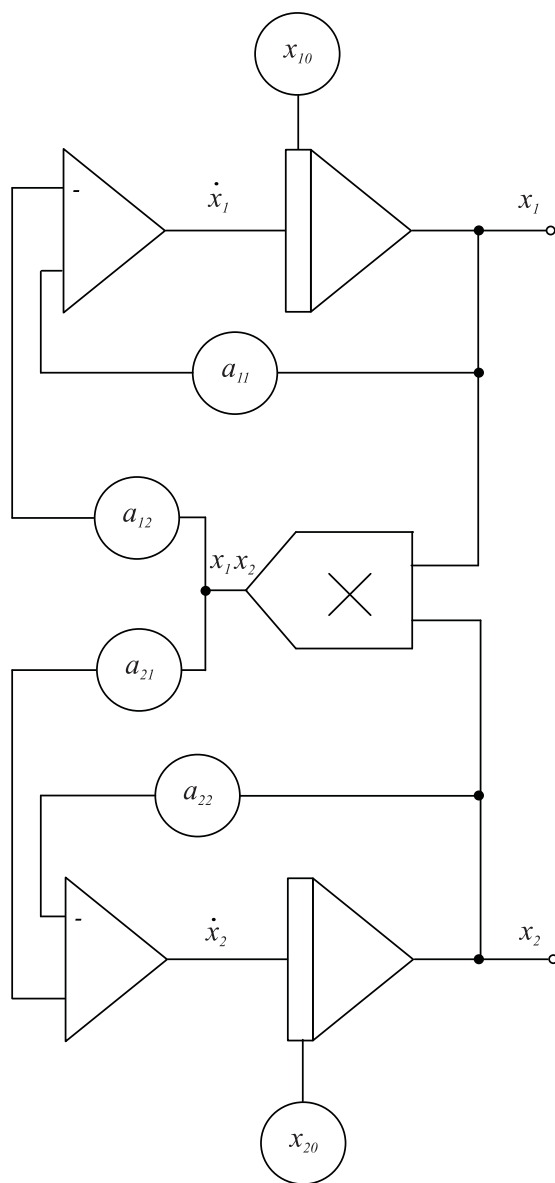
Model ekološkega sistema, v katerem nastopajo roparji in žrtve, smo predstavili v primeru 3.3. Primer prikazuje uporabnost indirektne metode pri reševanju sistema nelinearnih diferencialnih enačb. Matematični model ima obliko

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

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



Slika 6.9: Simulacijska shema ekološkega sistema

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.

□

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:

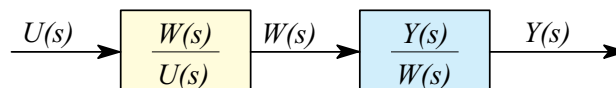
- Če koeficient pri členu z najvišjo potenco v imenovalcu ni enak 1, je potrebno vse člene polinomov v števcu in imenovalcu deliti s tem koeficientom.
- Prenosno funkcijo $\frac{Y(s)}{U(s)}$ razdelimo s pomočjo pomožne spremenljivke $W(s)$ v dve prenosni funkciji; prva predstavlja realizacijo imenovalca

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

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

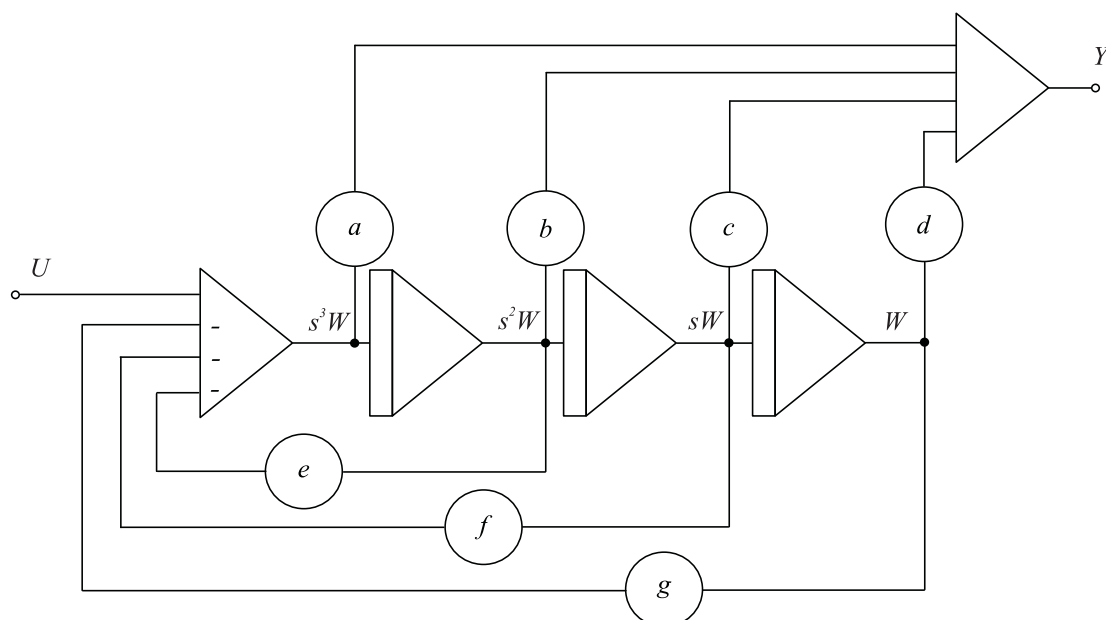
- Preuredimo enačbo (6.10) v obliko

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

- 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 imenovalec 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. $\ddot{y} + e\dot{y} + f\dot{y} + gy = a\ddot{u} + b\dot{u} + cu + du$).

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.

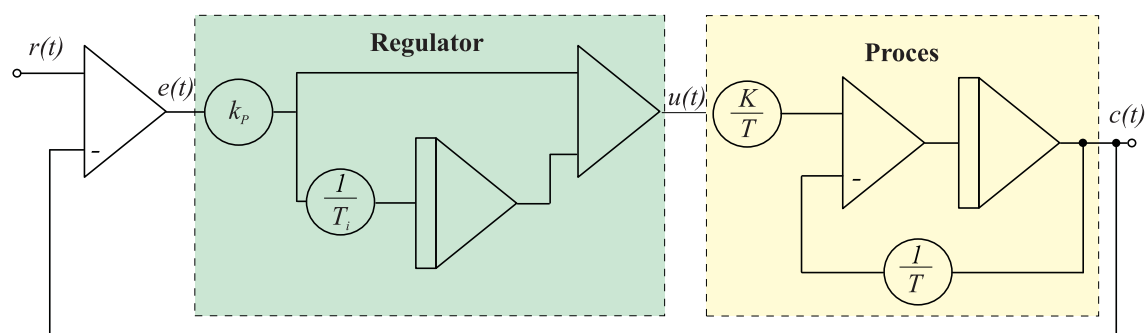
Primer 6.5 Regulacijski sistem vsebuje regulator PI

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

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 $u(t)$. $\square$



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 vešč 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

Okolje Matlab-Simulink je razširitev paketa Matlab in ponuja učinkovito simulacijo sistemov po obravnavanih konceptih. Hkrati ohranja vse funkcije, ki jih omogoča Matlab.

Simulink je namenjen predvsem naslednjima dvema načinoma uporabe:

- definiranju modelov in
- analizi modelov.

V praksi se ta dva načina pogosto uporabljata iterativno, ko želimo s spreminjanjem parametrov sistema doseči določeno želeno obnašanje.

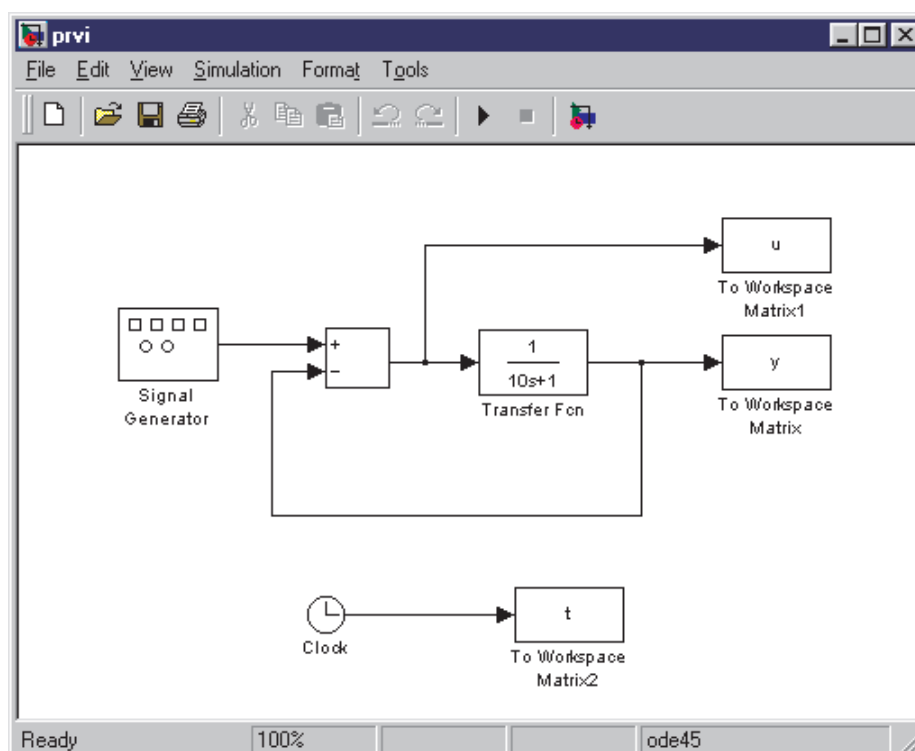
Zaradi uporabniške prijaznosti je paket organiziran v obliki oken, kjer so shranjeni bloki posameznih podsistemov (*block diagram windows*). Načrtovanje novih modelov s pomočjo definiranih podblokov in njihovo editiranje poteka s pomočjo miške. Analiza modela je možna z uporabo ukazov v meniju Simulinka ali pa z uporabo ukazov znotraj ukaznega okna v Matlabu. Vgrajeno orodje za analizo omogoča uporabo različnih simulacijskih algoritmov, `linmod` funkcijo, ki omogoča linearizacijo sistema in `trim` funkcijo, ki izvede analizo ustaljenega stanja. Paket omogoča tudi analizo splošnega linearne sistema.

Rezultate simulacije lahko opazujemo *on-line* z uporabo prikazovalnikov ali pa na koncu simulacije, znotraj Matlab ukaznega okna.

Opisovali bomo predvsem osnovne lastnosti, ki veljajo v sedanji pa tudi v starejših verzijah.

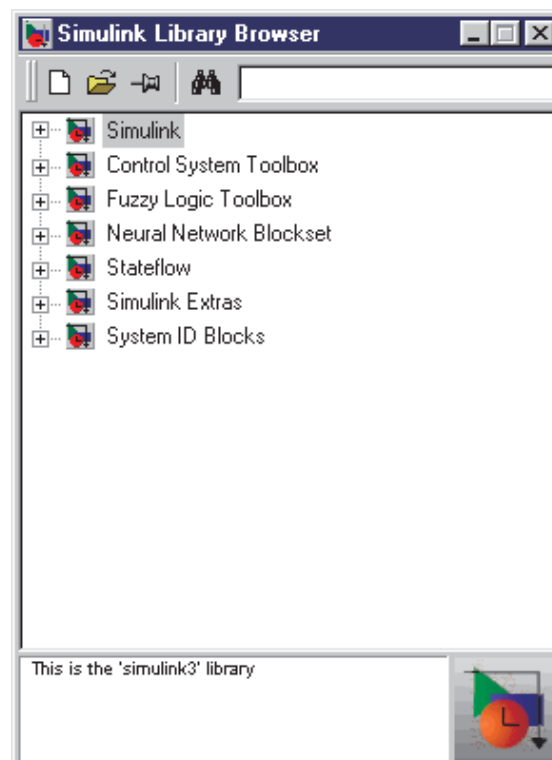
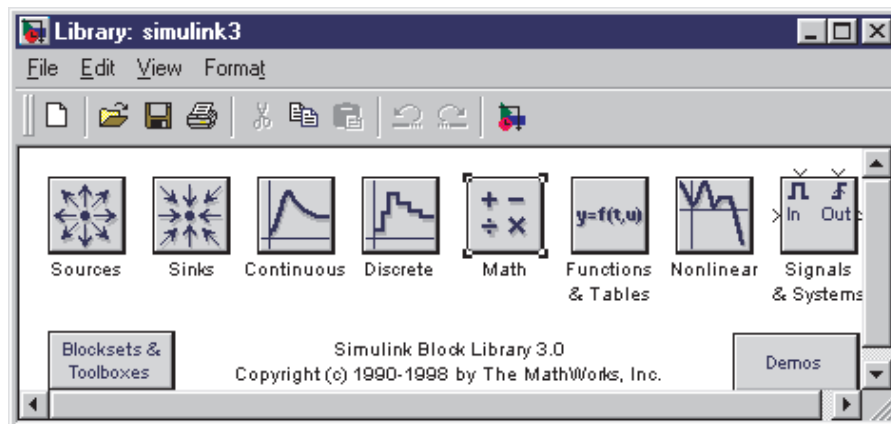
Kako najhitreje začeti z delom

Če želimo kar najhitreje začeti z delom, moramo poznati vsaj nekaj najosnovnejših ukazov. Tak način pristopa je prikazan v nadaljevanju v devetih korakih, v katerih bomo narisali shemo (shranjeno v datoteki s podaljškom *.mdl, ki jo prikazuje slika 7.1).



Slika 7.1: Model prvi.mdl v Simulinku

1. Znotraj Matlab ukaznega okna zaženemo ukaz Simulink, ki odpre Simulink Library Browser. S pritiskom na desni gumb na izbiri Simulink, odpremo lahko tudi Simulink Block Library okno. Okni sta na sliki 7.2.
2. Izberemo opcijo **New** v menuju **File** Simulink Block Library okna, ki odpre novo prazno okno v katerem kreiramo model. Nov model ima ime *Untitled*. Ime lahko ob shranjevanju spremenimo.
3. Odpremo eno ali več knjižnic podblokov (dvojni klik na zeleno knjižnico) in povlečemo (drag) zelene podbloke v aktivno okno.



Slika 7.2: Osnovni okni programskega paketa Simulink (knjižnica blokov in brskalnik po knjižnicah)

4. Povežemo bloke med seboj tako, da potegnemo črte med izhodi in vhodi v bloke. Črto potegnemo tako, da kazalec miške pozicioniramo na izhod bloka in s pritiskom levega gumba kazalec premaknemo do zelene točke.

5. Odpremo enega ali več blokov (dvojni klik z miško na določen blok) in spremenimo njegove parametre. Parametri morajo biti podani v sintaksi jezika Matlab.
6. Shranimo kreirani model z uporabo ukaza **Save As** v meniju **File**.
7. Zaženemo simulacijo z izbiro ukaza **Start** znotraj menuja **Simulation**. Med izvajanjem simulacije se omogoči menujska opcija **Stop**.
8. Znotraj menuja **Simulation** lahko spreminjamo krmilne parametre simulacije v opciji **Parameters**.
9. Za nadzor sistemskih spremenljivk lahko uporabimo prikazovalnike (**Scope block**) ali bloke, ki zapisuje spremenljivke v delovni prostor Matlaba (**To Workspace**).

Delo z objekti

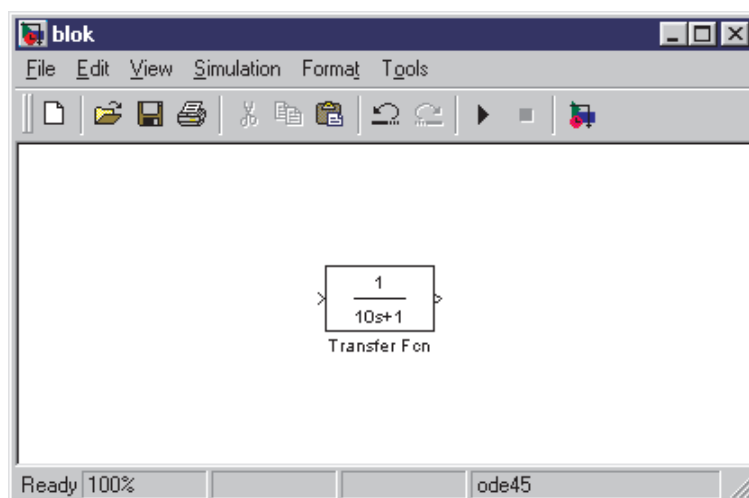
Izbira objektov

Določene funkcije delujejo le, če objekt izberemo tako, da kliknemo nanj. Izbrani objekt je označen z majhnimi črnimi kvadrati v ogliščih ikone. Za izbiro več objektov je potrebno držati pritisnjeno tipko **Shift** in z levim gumbom izbrati zelene objekte. Za izbiro vseh objektov v aktivnem oknu lahko uporabimo ukaz **Select All** v meniju **Edit**.

Množico objektov lahko izberemo tudi tako, da s pritisnjenim levim gumbom označimo polje, kjer se nahajajo objekti, ki jih želimo izbrati.

Delo z bloki

Bloki imajo tako splošne kot specifične lastnosti. Splošne lastnosti so tiste, ki so skupne vsem blokom. Te lastnosti so velikost in pozicija. Specifične lastnosti pa so tiste, ki so lastne samo določenemu bloku. Če preslikamo blok v nov blok se ohranijo vse lastnosti prednika razen pozicije. Blok - prenosno funkcijo prikazuje slika 7.3.



Slika 7.3: Blok - prenosna funkcija v okolju Simulink

Premikanje in kopiranje blokov. Bloke lahko premikamo ali kopiramo iz enega v drugo Simulink okno z vlečenjem (dragging) s pomočjo miške ali z uporabo ukazov **Cut**, **Copy** ali **Paste** v meniju **Edit**.

Uporaba miške. S kurzorjem izberemo blok, ki ga želimo premakniti in ga s pritiskom na levim gumbom izvlečemo na želeno mesto, kjer sprostimo gumb. Če sta prvotna pozicija bloka in končna pozicija bloka v istem aktivnem oknu, potem je rezultat akcije premik bloka. Vse povezave med bloki ostanejo nespremenjene. V primeru, ko je končna pozicija v drugem oknu, se blok kopira na novo pozicijo.

S pritiskom tipke **Ctrl** (ali z uporabo desnega gumba) med to akcijo se blok kopira znotraj aktivnega okna. V tem primeru dobi ime bloka podaljšek v obliki zaporedne številke.

Brisanje blokov. Bloke brišemo tako, da najprej izberemo tiste, ki jih želimo brisati in nato pritisnemo tipko **Delete** ali opcijo **Clear** ali **Cut** v meniju **Edit**.

Spreminjanje velikosti blokov. Dimenzijo bloka lahko spreminjamo z izvlečenjem kvadratkov, ki označujejo izbrani blok.

Editiranje imen blokov. Vsi bloki znotraj okna morajo imeti različna imena. Ime je lahko vidno ali pa nevidno, kar lahko določimo v stilski opciji **Style**. Ime lahko editiramo tako, da izberemo ime bloka in ga nadomestimo z novim, pozicioniramo kurzor na staro ime in vrinemo novo ali napišemo novo ime nekje

v oknu, ga izberemo in shranimo v odložišče (clipboard) s pritiskom tipk **Shift-Delete**, pozicioniramo kurzor na mesto imena in s pritiskom tipk **Shift-Insert** prenesemo novo ime.

Odpiranje blokov. Blok odpremo z dvojnimi klikom nanj. Običajno se odpre okno s parametri bloka, ki jih lahko spreminjamo. Na začetku imajo privzete vrednosti.

Spreminjanje stila. Po definiciji potekajo signali skozi blok od leve proti desni. Vhodi v blok so na levi in izhodi iz bloka na desni. Orientacija bloka se lahko spremeni z uporabo ukazov **Flip Block** in **Rotate Block**, ki so dostopni znotraj menija **Format**. **Rotate Block** ukaz obrne blok v smeri urinega kazalca za 90 stopinj, ukaz **Flip Block** pa za 180 stopinj. Znotraj menija **Format** je omogočeno tudi spreminjanje tipa zapisa v možnosti **Font**, obračanje imena bloka **Flip Name**, odstranitev imena z možnostjo **Hide Name** in senčenje blokov v možnosti **Show Drop Shadow**.

Povezovanje blokov

Za povezovanje blokov uporabljamo črte med izhodi in vhodi blokov. Če je vhod nepovezan je izhod vedno enak nič. Število povezav na posameznem izhodu iz bloka je neomejeno, medtem ko je dovoljena le ena linija na vhod bloka. Seveda tudi ne moremo povezovati vhodov med seboj in ravnotako tudi ne izhodov med seboj.

Risanje povezav med bloki. Dva bloka povežemo tako, da se z miško pozicioniramo na izhod določenega bloka, ki ga želimo povezati in s pritisnjenim levim gumbom izvlečemo črto do zelenega vhoda v blok. Ko sprostimo levi gumb se na mestu vhoda in izhoda bloka pojavi povezava s puščico, ki kaže smer poteka signala. Povezave med bloki lahko rišemo od izhoda na vhod bloka ali pa tudi obratno. Smer signala ostaja nespremenjena, od izhoda bloka na vhod bloka.

Brisanje povezav. Povezavo brišemo tako, da jo izberemo in nato pritisnemo tipko **Delete** ali pa izberemo ukaz **Clear** ali **Cut** v meniju **Edit**.

Segmentiranje in dodajanje povezav. Če želimo določeno povezavo segmentirati na več lomljenih segmentov, potem se pozicioniramo s kurzorjem na mesto, kjer želimo lomljeno črto in pritisnemo tipko **Shift**. Hkrati pritisnemo levi gumb in izvlečemo kurzor na zeleno mesto.

Novo povezavo lahko kreiramo iz katerekoli točke obstoječe povezave. Povezavo dodamo tako, da se pozicioniramo s kurzorjem na mesto, kjer želimo odjemati signal in s pritisnjenim desnim gumbom izvlečemo novo povezavo. Vrednost signala na izhodu iz povezave je enaka vrednosti na odjemnem mestu. Enako lahko storimo s pritiskom levega gumba in hkratnim pritiskom tipke **Ctrl**.

Povezave med bloki so lahko multipleksirane. Informacijo o multipleksiranih signalih lahko prikažemo na shemi na ta način, da v menuji **Format** izberemo možnost **Vector Line Widths** in odebelimo povezave, ki so multipleksirane **Wide Vector Lines**. Hkrati pa lahko v shemo vnesemo informacijo o tem, kaksen tip podatkov se prenaša po povezavi **Port Data Types**.

Menuji

Simulink je organiziran s pomočjo menujev, ki so dostopni na vrhu delovnega okna. V nadaljevanju bomo predstavili pomembnejše zmožnosti posameznih menujev in možne izbire.

File

New Kreiranje novega sistema

Open Izbira sistema, ki že obstaja nekje na disku.

Close Odstranitev sistema iz delovnega spomina.

Save Zapis sistema na datoteko z istim imenom.

Save As Zapis sistema na novo datoteko.

Model Properties Dokumentacija modela (avtor, datum, opis)

Print Tiskanje bločnega diagrama.

Exit Matlab Izstop.

Edit

Undo Prekliči zadnje spreminjanje sheme.

Redo Ponovi zadnje spreminjanje sheme.

Cut Zapis izbranega objekta v v odložišče in brisanje objekta v aktivnem oknu.

Copy Zapis objekta v v odložišče.

Paste Kopiranje vsebine iz odložišča na izbrano pozicijo.

Delete Brisanje izbranega objekta.

Select All Izbira vseh objektov znotraj aktivnega okna.

Copy Model Kopiranje modela.

Create Subsystem Grupiranje izbranih objektov v podsistemski blok.

Mask Subsystem Kreiranje vmesnika (maske) za blok.

Look Under Mask Pogled pod masko.

Go To Library Link V knjižnico blokov iz katere je določen blok.

Break Library Link Prekinitev povezave s knjižnico (kadar želimo spreminjati attribute bloka, ki smo ga vnesli iz knjižnice).

Unlock Library Spreminjanje atributov blokov v določeni knjižnici.

Update Diagram Osveževanje diagrama.

View

Toolbar Izpis orodne vrstice pod menujsko vrstico.

Status Bar Izpis statusne vrstice na spodnjem robu okna. Statusna vrstica vsebuje podatke o simulaciji.

Model Browser Brskalnik modela, ki je uporaben v primeru, ko je model sestavljen iz množice podmodelov.

Block Data Tips Izpis oblačka z atributi nad blokom, ki se ga dotaknemo s kurzorjem.

Library Browser Prikaz knjižničnega brskalnika.

Zoom In Povečava diagrama.

Zoom Out Odstranitev povečave.

Fit System to View Prilagajanje izgleda.

Normal Običajni izgled.

Simulation

Start Zagon simulacije modela v aktivnem oknu.

Stop Prekinitev simulacije.

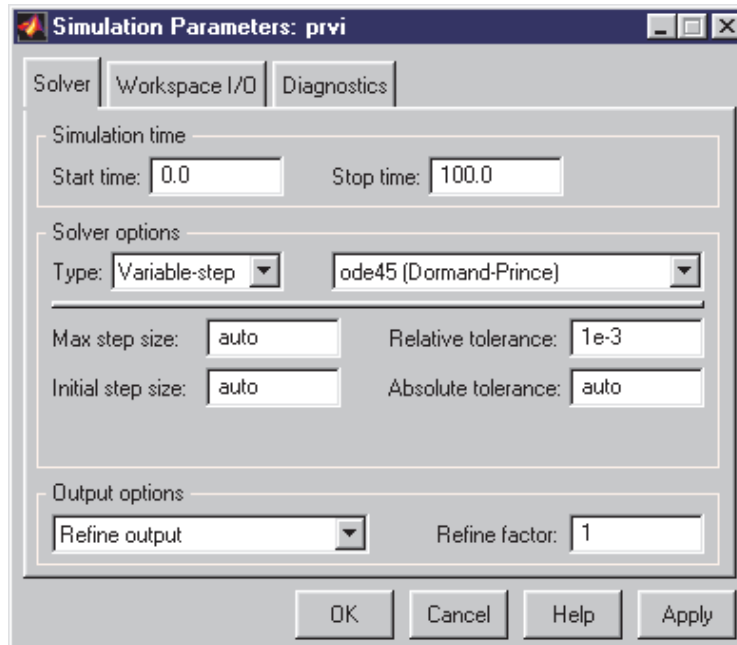
Configuration parameters Editiranje parametrov simulacije. Nastavitev začetnega in končnega časa simulacije (**Start time**, **Stop time**), nastavitev integracijskega algoritma (**Solver options**), maksimalnega in minimalnega koraka pri integraciji (**Max step size**, **Min step size**) in relativne ter absolutne napake pri simulaciji (**Relative tolerance**, **Absolute tolerance**).

Med simulacijo se računski korak sproti prilagaja (spreminja med **Min Step Size** in **Max Step Size**), tako da napaka ni večja od dopustne (**Tolerance**). Medtem ko v enostavnejših primerih lahko pustimo **Min Step Size** in **Tolerance** na privzetih (default) vrednostih, pa moramo **Max Step Size** izbrati tako, da bo na opazovanem intervalu (**Stop Time** - **Start Time**) dovolj izračunanih točk (npr.

$$\text{Max Step Size} = (\text{Stop Time} - \text{Start Time})/100$$

100 točk). Ustrezno času opazovanja (le ta zavisi od dolžine prehodnih pojavov, časovnih konstant, lastnih in vzbujaalnih frekvenc, ...) moramo izbrati

tudi parametre blokov za opazovanje.



Slika 7.4: Simulacijski parametri

Format V tem menuju je omogočeno redefiniranje orientacije blokov, kar je bilo opisano že v prejšnjih poglavjih, izgleda blokov (**Foreground Color**, **Background Color**) in okna (**Screen Color**).

Knjižnice blokov

V tem razdelku si bomo ogledali pomembnejše bloke iz nekaterih knjižnic. Večina blokov omogoča nastavljanje parametrov s pomočjo interaktivnega okna, ki ga odpremo z dvojnimi klikom na izbranem bloku. Interaktivno okno vsebuje naslednje elemente:

- ime in tip bloka na vrhu okna,
- kratek opis bloka in njegovo funkcijo,
- polje, kjer lahko vnašamo funkcijske parametre, ki morajo biti v sintaksi MATLABA,

- v spodnjem delu so gumbi **OK** s katerim potrdimo trenutne izbrane vrednosti parametrov, **Cancel**, ki omogoča prekinitev in nastavitev starih vrednosti, ki so bile nastavljene pred vstopom v interaktivno okno in **Help**, ki nudi pomoč uporabniku.

Simulink shranjuje vrednosti parametrov, ki so podane skozi interaktivna okna in jih prenaša v Matlab, ko zaženemo simulacijo.

V naslednjih razdelkih bodo opisane knjižnice blokov, ki so standardne v Simulinku.

Sources

Knjižnica, ki je prikazana na sliki 7.5, vsebuje generatorje (izvore) naslednjih signalov:

Constant Generator konstantne vrednosti.

Signal Generator Generator različnih periodičnih signalov.

Step Generator stopničaste funkcije.

Ramp Generator linearno naraščajočega signala.

Sine Wave Generator sinusnega signala.

Repeating Sequence Ponavljajoča sekvenca izhodnega signala glede na čas.

Discrete Pulse Generator Diskretni pulzni generator.

Pulse Generator Pulzni generator.

Chirp Signal Sinusni signal s spremenljivo frekvenco.

Clock Generira in izpiše sistemski čas.

Digital Clock Generira in izpiše sistemski čas v digitalni obliki.

From File Generiranje signala s pomočjo datoteke v Matlabu.

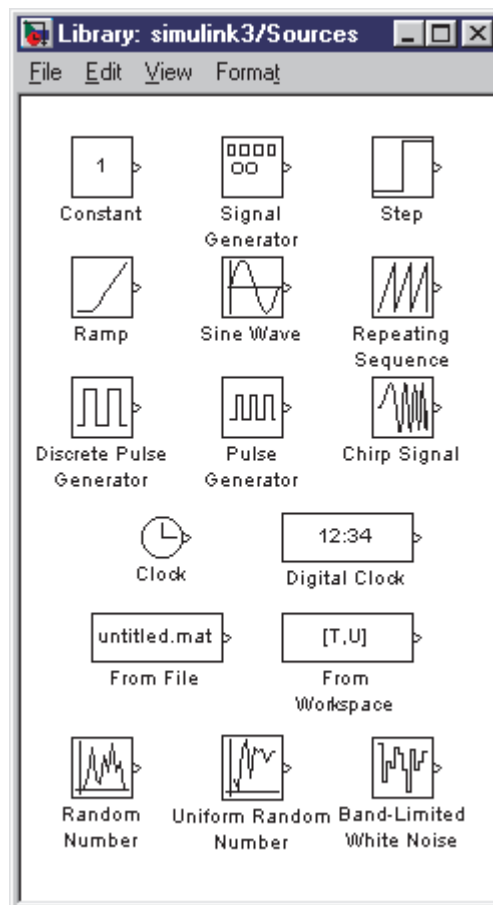
From Workspace Generiranje signala s pomočjo spremenljivke v Matlabu.

Random Number Generator naključnih števil.

Band-Limited White Noise Generator belega šuma z določeno pasovno širino.

Sinks

Knjižnica, ki vsebuje prikazovalnike in ponore signalov, je prikazana na sliki 7.6.



Slika 7.5: Knjižnica izvorov

Scope Prikazovalnik signalov med simulacijo.

XY Graph Prikazovalnik v obliki Matlab diagrama.

Display Prikazovalnik vrednosti signala.

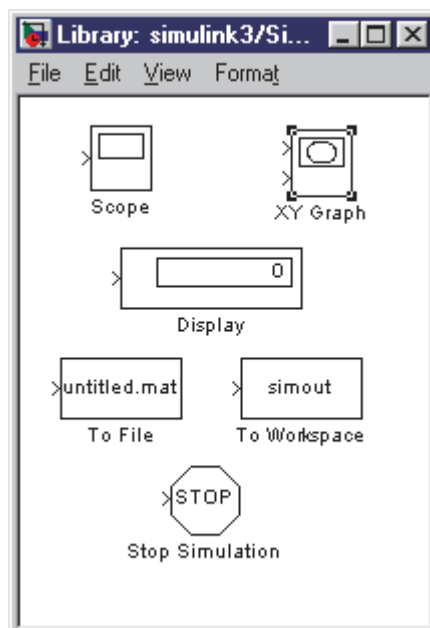
To File Zapisovanje signala na datoteko.

To Workspace Zapisuje signala v delovni pomnilnik Matlaba v obliki vektorja.

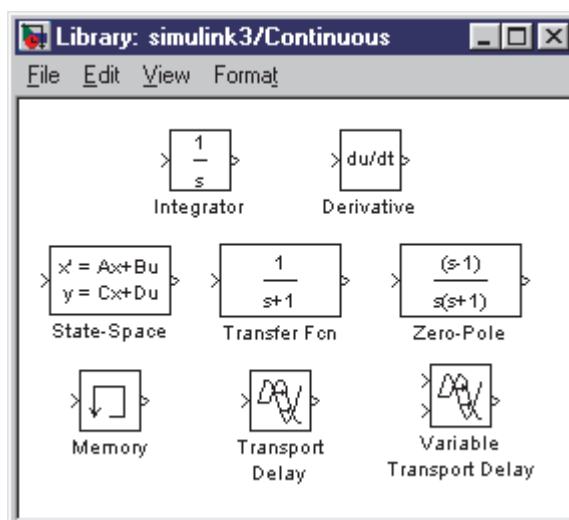
Stop Simulation Zaključek simulacije pri določenem pogoju.

Continuous

Knjižnica časovno zveznih blokov je na sliki 7.7.



Slika 7.6: Knjižnica ponorov signalov



Slika 7.7: Knjižnica zveznih blokov

Integrator Integriranje vhodnega signala.

Derivative Odvajanje vhodnega signala.

State-Space Zvezni model podan v prostoru stanj.

Transfer Fcn Zvezna prenosna funkcija - polinomska oblika.

Zero-Pole Zvezni model podan z ničlami, poli in ojačenjem - faktorizirana oblika.

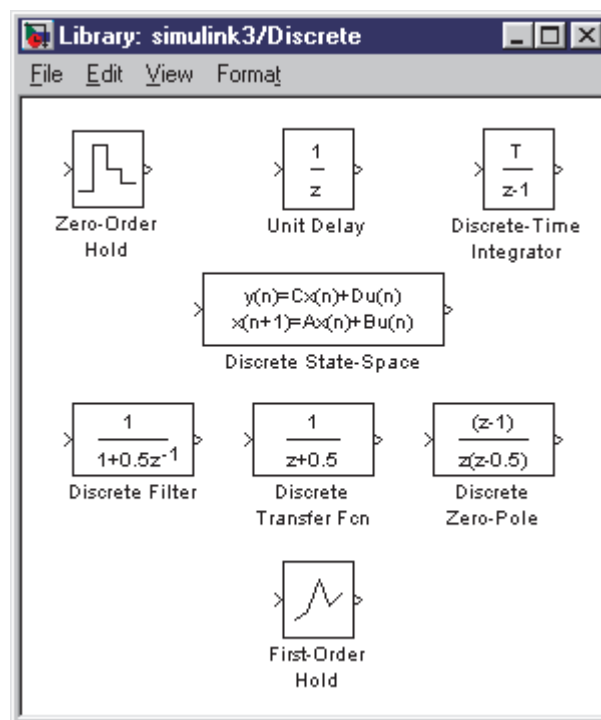
Memory Zakasnitev enega integracijskega koraka.

Transport Delay Transportni mrtvi čas.

Variable Transport Delay Spremenljiv transportni čas.

Discrete

Slika 7.8, prikazuje knjižnico časovno diskretnih blokov.



Slika 7.8: Knjižnica diskretnih blokov

Zero-Order Hold Zadrževalnik ničtega reda.

Unit Delay Časovna zakasnitev signala za en vzorčni interval.

Discrete-Time Integrator Diskretni integrator.

Discrete State-Space Diskretni sistem podan v prostoru stanj.

Discrete Filter Filter.

Discrete Transfer Fcn Diskretna prenosna funkcija - polinomska oblika.

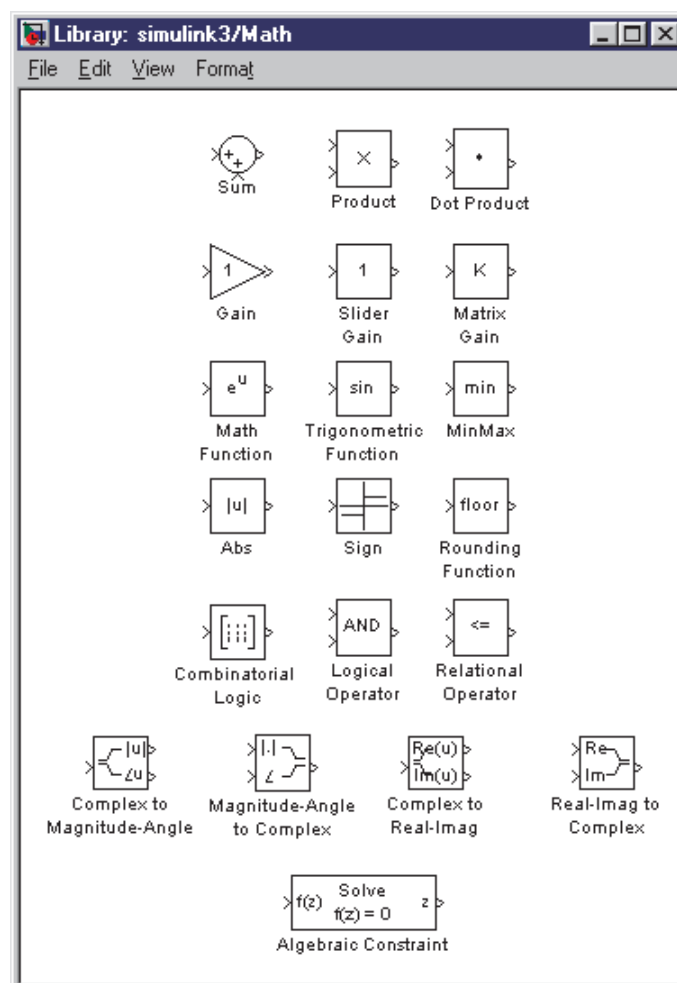
Discrete Zero-Pole Diskretna prenosna funkcija podana z ničlami, poli in

ojačanjem - faktorizirana oblika.

First-Order Hold Zadrževalnik prvega reda.

Math

Slika 7.9, prikazuje knjižnico matematičnih blokov.



Slika 7.9: Knjižnica matematičnih blokov

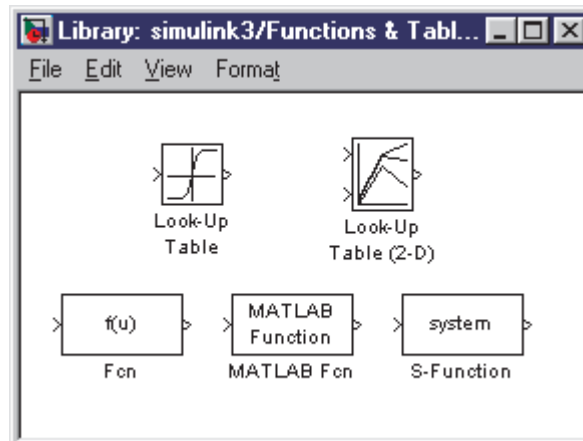
Sum Sumator - sešteva ali odšteva signale.

Gain Signal množi s poljubno realno konstanto.

Omenili smo le bloka, ki se uporabljata skoraj v vsaki simulacijski shemi. Funkcije ostalih blokov so razvidne iz ikon.

Functions and Tables

Slika 7.10, prikazuje knjižnico matematičnih funkcijskih in tabelarnih blokov.



Slika 7.10: Knjižnica matematičnih funkcijskih in tabelarnih blokov

Look-Up Table Generator funkcije ene neodvisne spremenljivke.

Look-Up Table (2-D) Generator funkcije dveh neodvisnih spremenljivk.

Fcn Realizacija matematičnega izraza v sintaksi Matlab.

MATLAB Fcn Realizacija Matlab funkcije na vhodu v blok.

S-Function Realizacija S-funkcije v obliki bloka.

Nonlinear

Knjižnico nelinearnih blokov prikazuje slika 7.11.

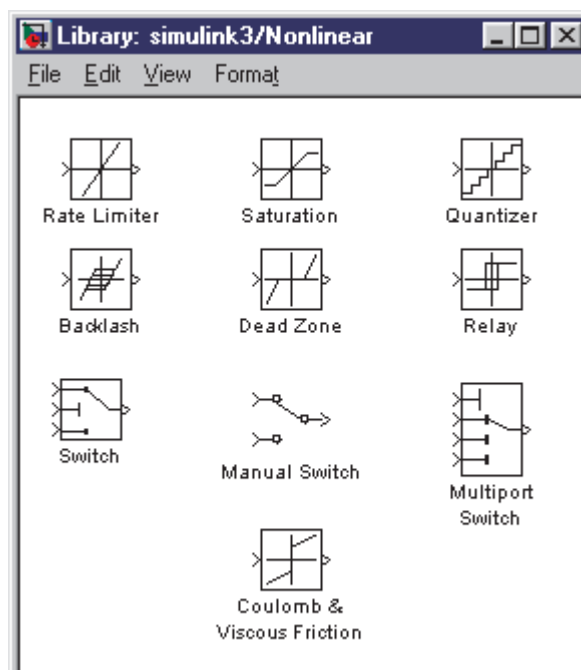
Funkcije blokov so razvidne iz ikon.

Signals and Systems

Knjižnica blokov, ki omogočajo povezave med objekti znotraj Simulinka je predstavljena na sliki 7.12.

In1 Omogoča povezavo z zunanjim vhodnim portom.

Out1 Omogoča povezavo z zunanjim izhodnim portom.



Slika 7.11: Knjižnica nelinearnih blokov

Mux Združevanje množice skalarnih vrednosti v vektor.

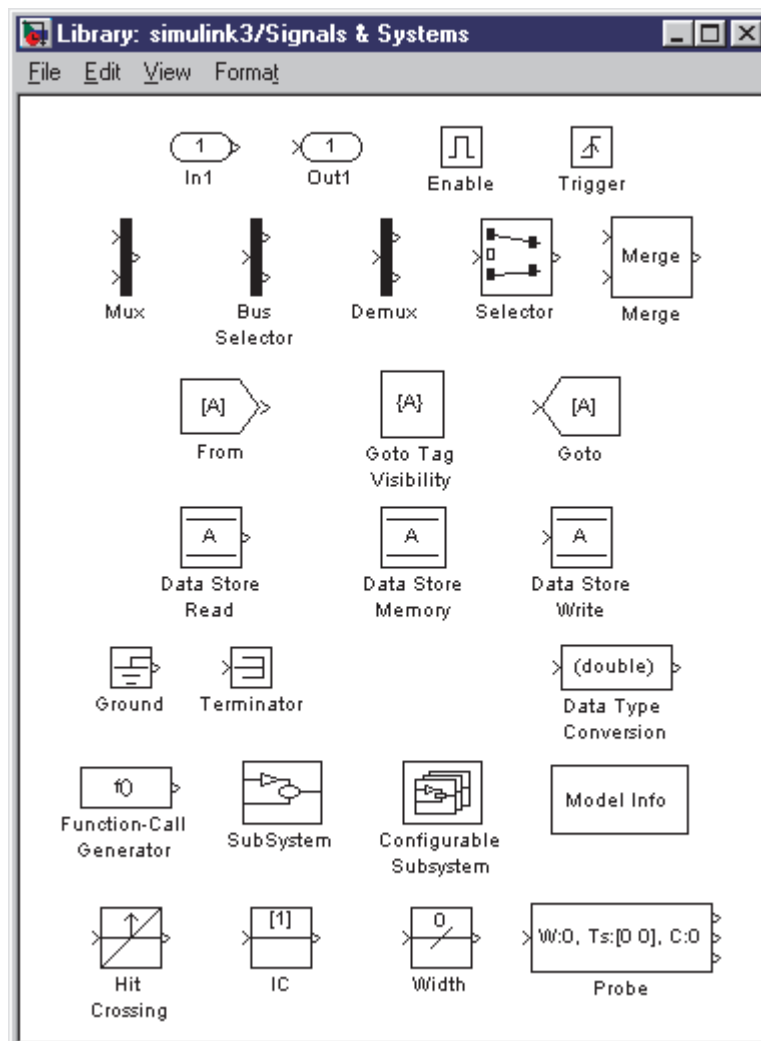
Demux Razdruževanje vektorske vrednosti v več skalarnih. Funkcije ostalih blokov so razvidne iz ikon.

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

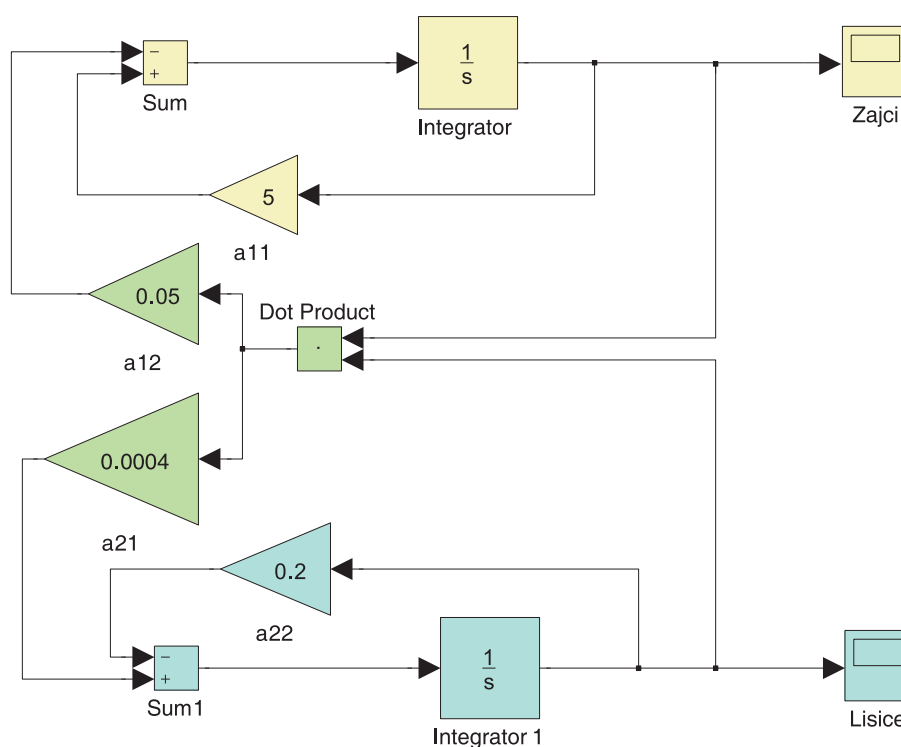


Slika 7.12: Knjižnica blokov povezav in sistemov

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 dvojnim 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.

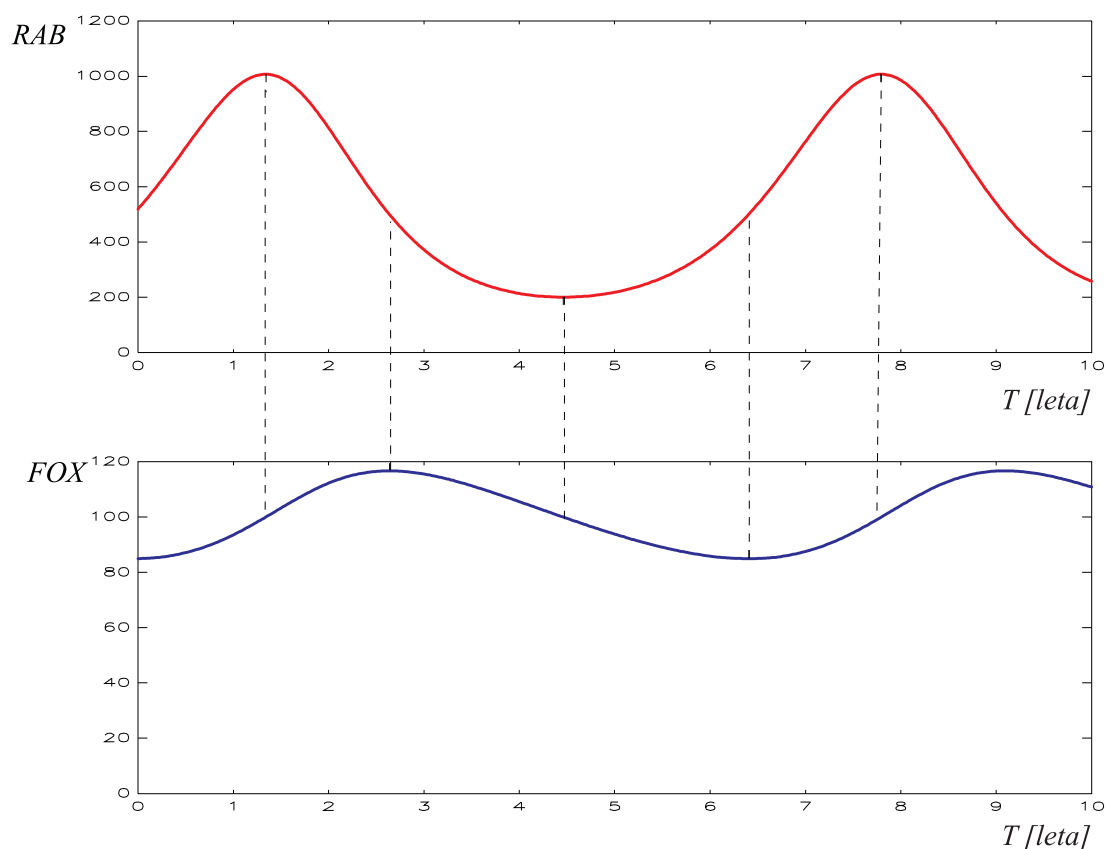
Simulink shemo prikazuje slika 7.13.



Slika 7.13: Simulacijska shema v Simulinku za ekološki primer zajcev in lisic

Rezultate simulacije prikazuje slika 7.14.

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



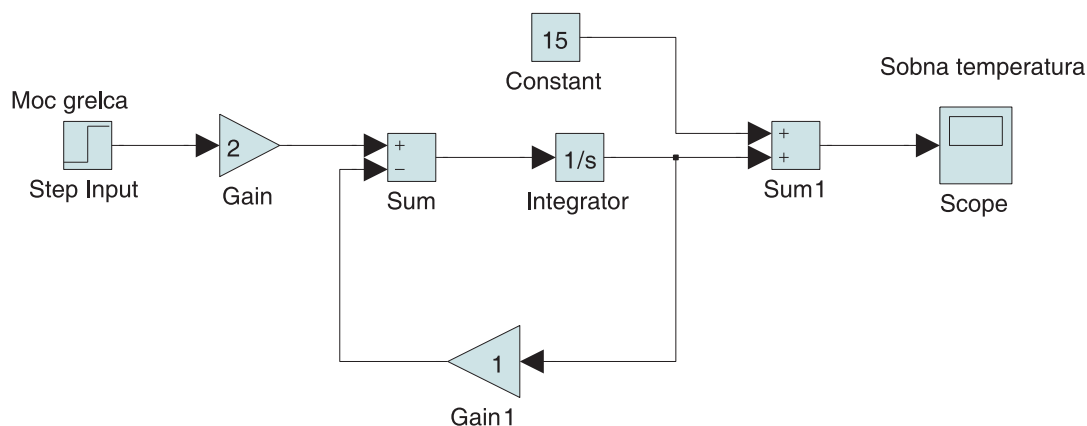
Slika 7.14: Časovna poteka števila zajcev in lisic

□

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 (3.17) smo dodali še en sumator, s pomočjo katerega smo iz temperature v delovni točki in temperature okolice izračunali absolutno temperaturo. Simulink shemo prikazuje slika 7.15.

Simulacijski model procesa nadgradimo v shemo za regulacijo po sliki 3.20. 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 7.15: 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

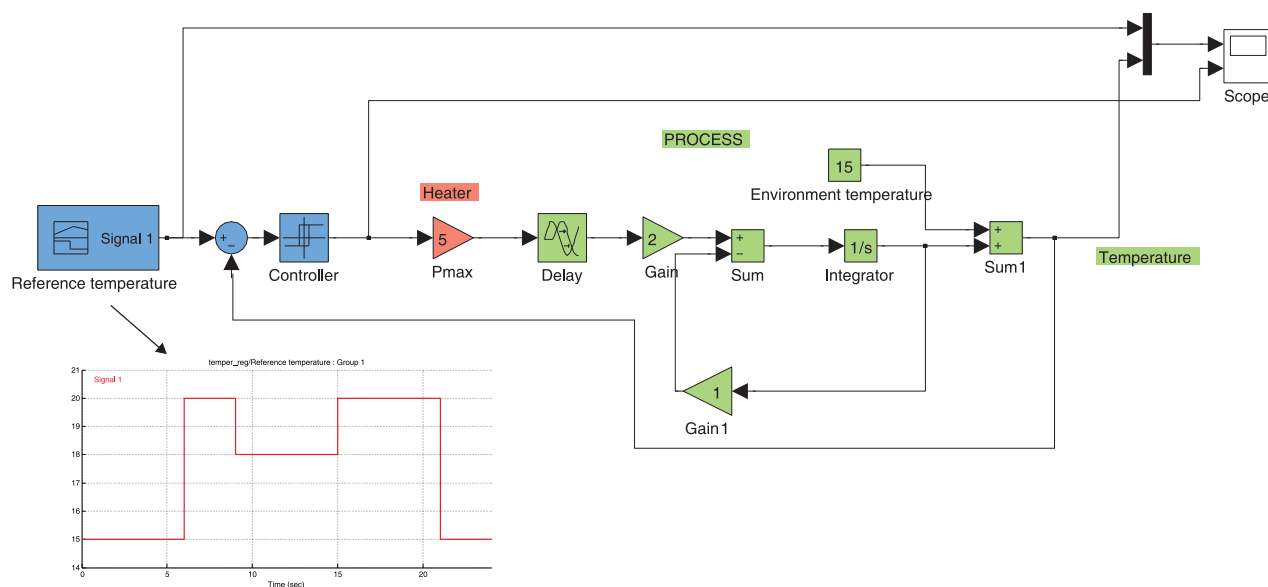
Discontinuous uporabimo **Relay** blok (privzeto ime smo nadomestili z imenom **Controller**). Bloku podamo vrednosti, pri katerih vklopi oz. izklopi ter vklopno in izklopno vrednost. Grelo simuliramo z ojačevalnim blokom **pmax**, dodatno zakasnitev pa z blokom **Transport Delay** iz knjižnice Continuous. Simulink shemo prikazuje slika 7.16.

Temperatura v prostoru ter vklapljanje in izklapljanje grela sta prikazana na sliki 7.17.

□

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



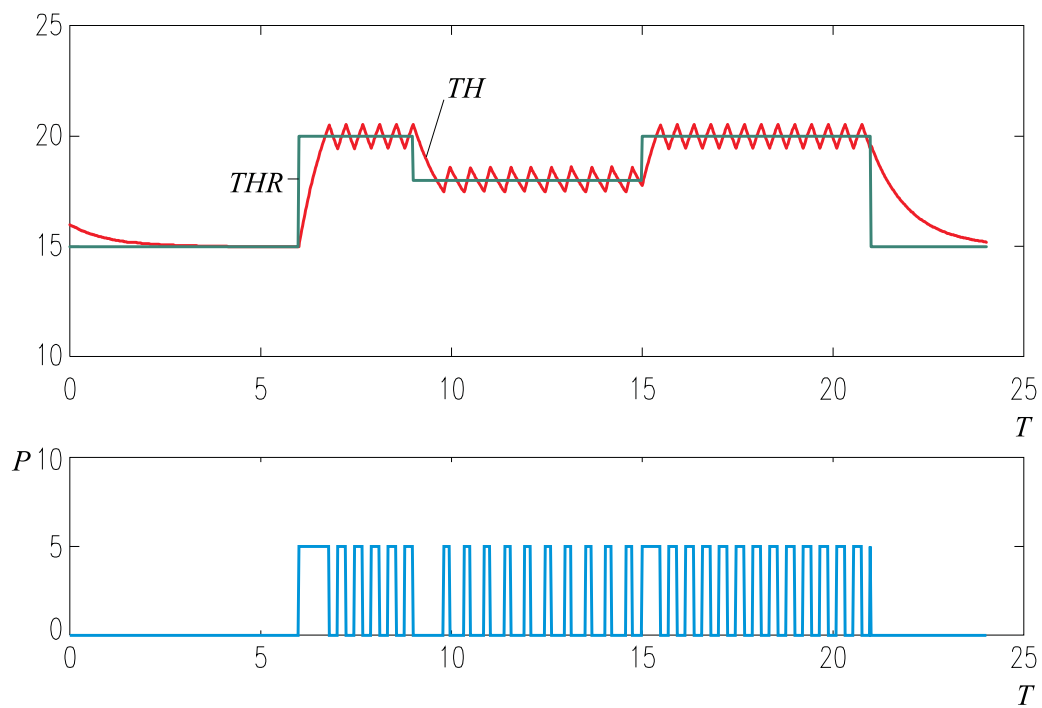
Slika 7.16: Simulacijska shema regulacije temperature v prostoru in prikaz referenčnega signala

shemo prikazuje slika 7.18.

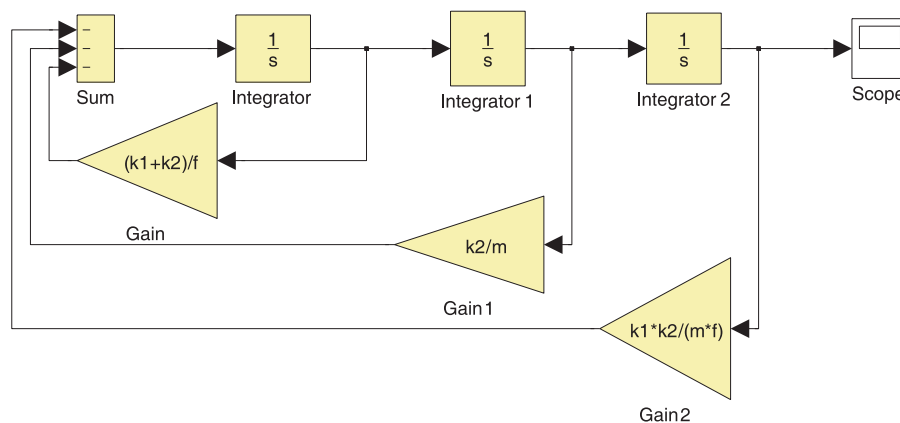
Opazujemo pomik karoserije pri negativnem začetnem pogoju $y(0) = -0.05$ m. Praktično to lahko pomeni, da se je voznik vsedel v avtomobil in v trenutku $t=0$ izstopil iz vozila. Opazujemo, kako se karoserija vrne v mirovno lego. Rezultate simulacije prikazuje slika 7.19.

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.

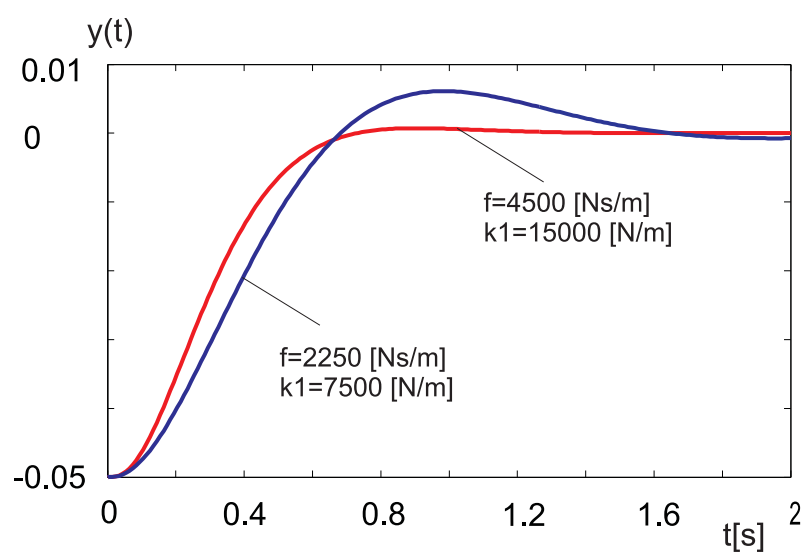
□



Slika 7.17: Rezultati temperaturnega regulacijskega problema



Slika 7.18: Simulink shema modela avtomobilskega vzmetenja



Slika 7.19: 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 vzbujaanja.

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 2.26, stopnica - slika 2.18, linearno naraščajoči signal - slika 2.17, parabola, ...), periodične signale (sinusni ali kosinusni signal - sliki 2.21 in 2.22, pravokotni impulz - slika 2.25, vlak impulzov - slika 2.27), 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 vzbujaanje. Odziv na stopničasto vzbujaanje 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 vzbujaajo 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 vzbujujanju 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

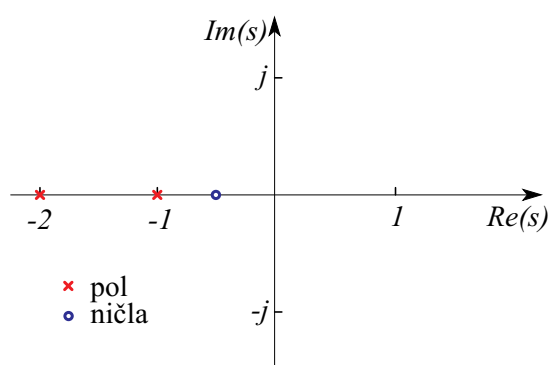
Omenili smo, da je prenosna funkcija kvocient dveh polinomov spremenljivke s . Oglejmo si sistem, ki ga opisuje diferencialna enačba

$$\ddot{y} + 3\dot{y} + 2y = 2\dot{u} + u \quad (8.7)$$

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 $B(s) = 0$, pa 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

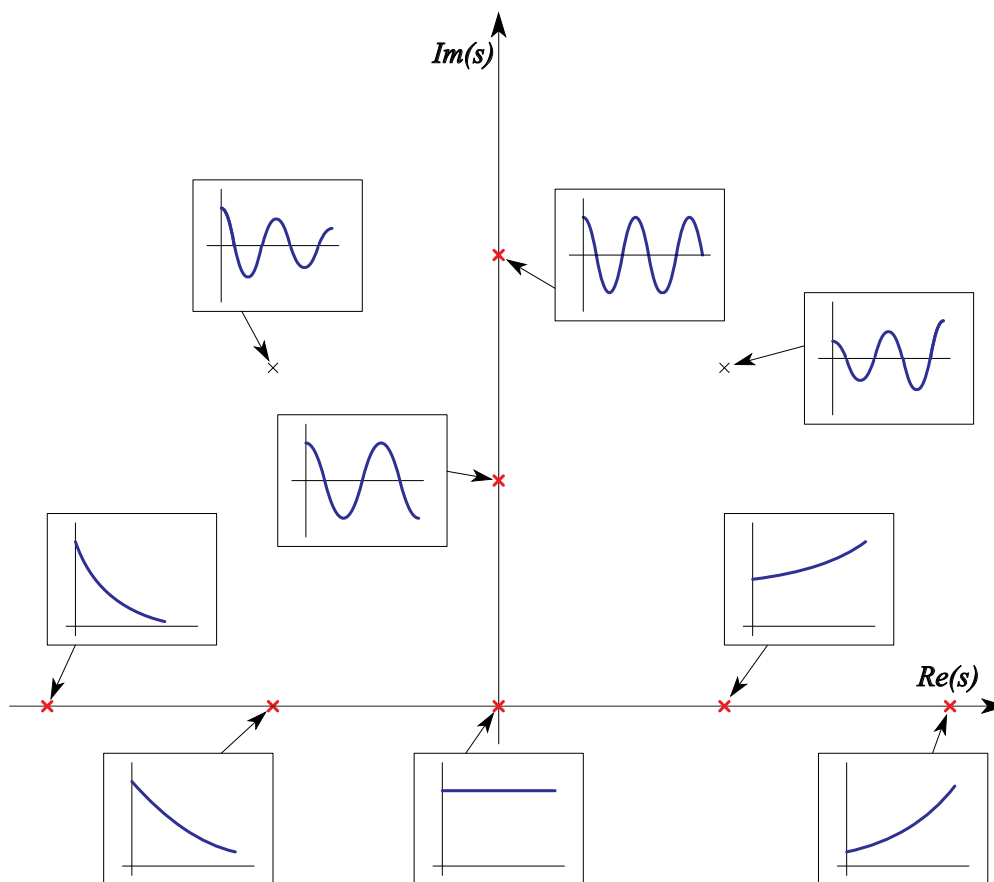
S pomočjo inverzne Laplace-ove transformacije izračunajmo iz prenosne funkcije odziv sistema na δ impulz. Slednje omogoča analizo vpliva polov in ničel na časovni odziv. Prenosno funkcijo $G(s)$ razvijemo v parcialne ulomke

$$G(s) = \frac{2s + 1}{(s + 1)(s + 2)} = \frac{-1}{s + 1} + \frac{3}{s + 2} \quad (8.9)$$

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 oddaljeni 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 linearnega 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.

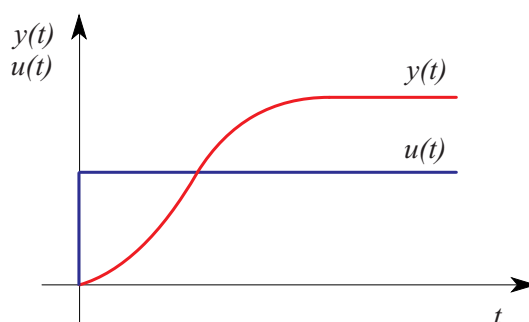
8.3 Razvrstitev sistemov

Proporcionalni, integrirni in diferencirni sistemi

Proporcionalni sistem z vhodom $u(t)$ 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 vzbujujanju 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 vzbujujanju $u(t)$ predstavlja slika 8.3.



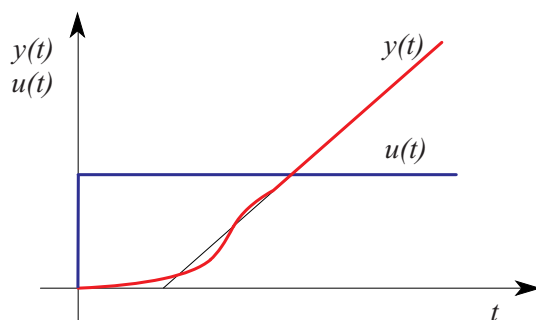
Slika 8.3: Karakteristični odziv proporcionalnega sistema

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 $D(0) \neq 0$. Taki sistemi so mejno stabilni in nimajo pravega ustaljenega stanja, saj odziv po prehodnem pojavu narašča pri stopničastem vzbujujanju. Tipičen integrirni proces predstavljajo razne vrste enosmernih motorjev, katerih zasuk po prehodnem pojavu linearno narašča pri konstantni vzbujujalni 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 ($j = 0$).

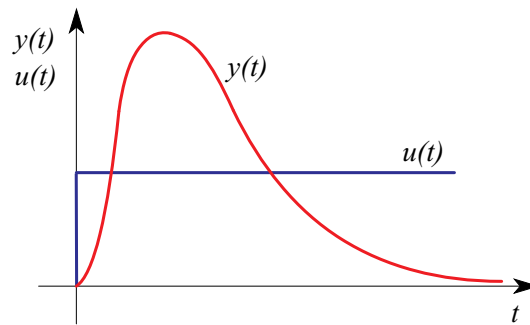
Vrsta sistema odločilno vpliva na vedenje regulacijskih (povratnozančnih) sistemov. Z višanjem vrste se zmanjšuje pogrešek v ustaljenem stanju oz. povečuje točnost regulacijskega sistema. Hkrati pa se povečujejo stabilnostni problemi, tako da je potrebno najti nek kompromis.

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

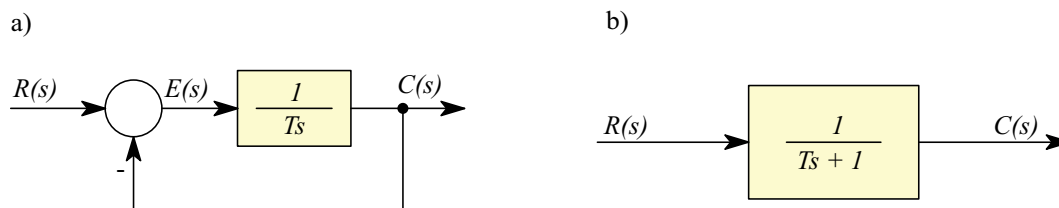
V časovnem prostoru je red sistema določen z redom najvišjega odvoda izhodne spremenljivke v diferencialni enačbi ali s številom spremenljivk stanja pri zapisu v prostoru stanj. Pri zapisu s prenosno funkcijo pa je red sistema določen s stopnjo polinoma v imenovalcu. Glede na to definicijo imamo sisteme 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

$$Y(s) = \frac{k}{Ts + 1} \cdot \frac{U_0}{s} \quad (8.16)$$

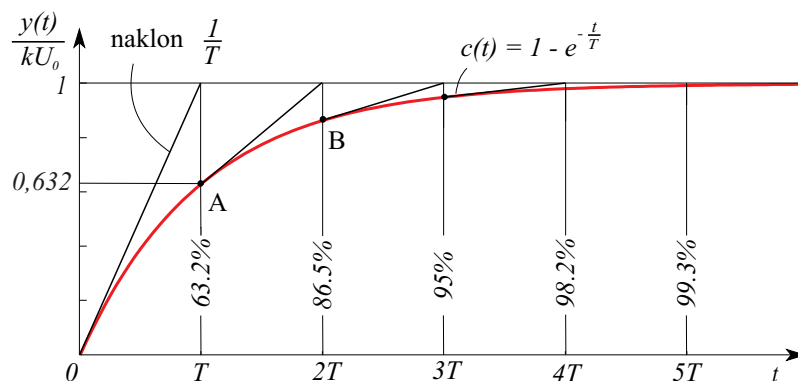
Če razvijemo $Y(s)$ po parcialnih ulomkih, dobimo

$$Y(s) = k U_0 \left[\frac{1}{s} - \frac{T}{Ts + 1} \right] \quad (8.17)$$

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

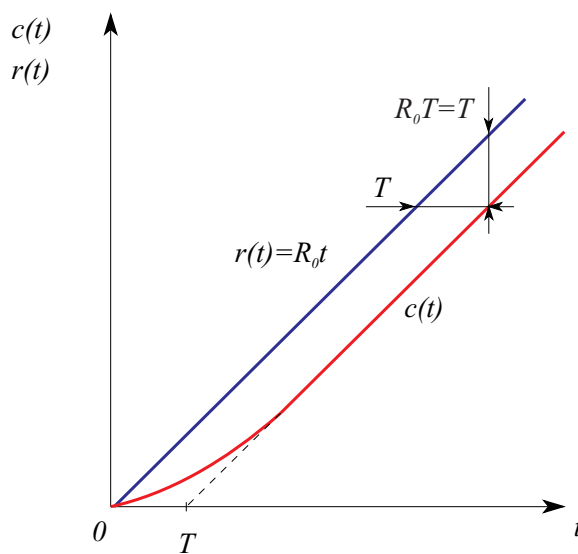
$$e(t) = R_0 t - c(t) = R_0 T (1 - e^{-\frac{t}{T}}) \quad (8.27)$$

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)

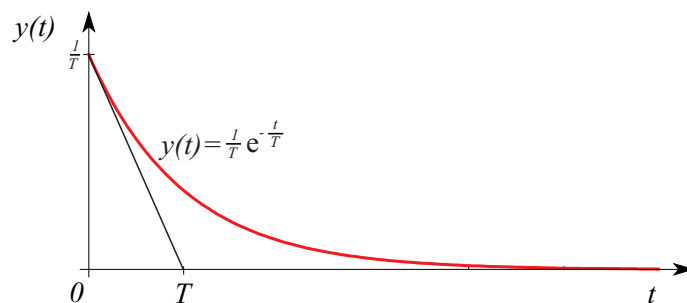
Ker je Laplace-ov transform δ impulza 1, določimo izhodni signal z enačbo

$$Y(s) = \frac{k}{Ts + 1} \quad (8.29)$$

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.

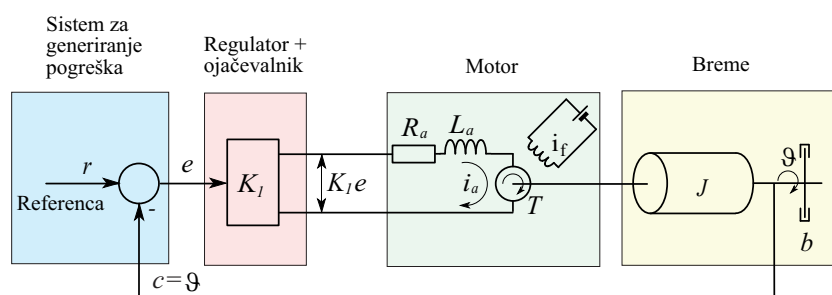
Slika 8.9: Odziv sistema 1. reda na δ impulz

8.4.2 Sistem drugega reda

Zakovitosti 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

$$\begin{aligned} K &= \frac{k_1 k_2}{R_a} \\ B &= b + \frac{k_2 k_3}{R_a} \end{aligned} \quad (8.36)$$

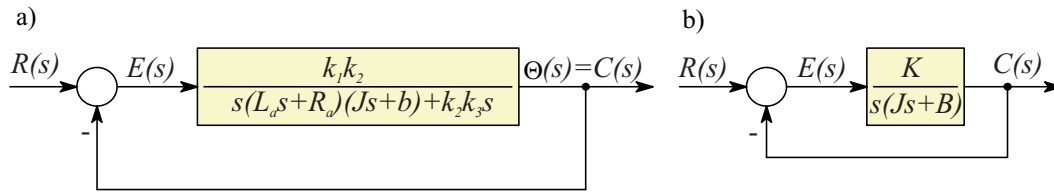
velja prenosna funkcija direktne veje (motorja)

$$G(s) = \frac{K}{s(Js + B)} \quad (8.37)$$

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 4.51

$$G_2(s) = \frac{E_o(s)}{E_i(s)} = \frac{\frac{1}{LC}}{s^2 + \frac{R}{L}s + \frac{1}{LC}} \quad (8.43)$$

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) $R_{kr} = 2\sqrt{\frac{C}{L}}$ ($R_{kr}^2 - 4\frac{L}{C} = 0$)

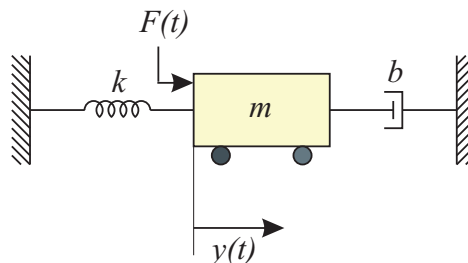
$$\zeta = \frac{R}{R_{kr}} = \frac{R}{2} \sqrt{\frac{C}{L}} \quad (8.45)$$

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 $G(s) = \frac{Y(s)}{F(s)}$

$$-ky(t) - b\dot{y}(t) + F(t) = m\ddot{y}(t) \quad (8.46)$$

$$\ddot{y}(t) + \frac{b}{m}\dot{y}(t) + \frac{k}{m}y(t) = \frac{1}{m}F(t) \quad (8.47)$$

$$s^2Y(s) + \frac{b}{m}sY(s) + \frac{k}{m}Y(s) = \frac{1}{m}F(s) \quad (8.48)$$

$$G(s) = \frac{Y(s)}{F(s)} = \frac{\frac{1}{m}}{s^2 + \frac{b}{m}s + \frac{k}{m}} \quad (8.49)$$

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 $0 < \zeta < 1$)

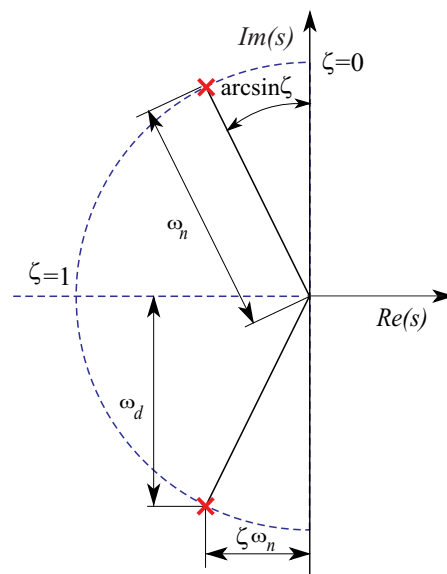
V tem primeru lahko zapišemo $Y(s)/U(s)$ kot

$$\frac{Y(s)}{U(s)} = \frac{\omega_n^2}{(s + \zeta\omega_n + j\omega_d)(s + \zeta\omega_n - j\omega_d)} \quad (8.52)$$

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

$$\begin{aligned} y(t) &= 1 - e^{-\zeta\omega_n t} \left(\cos \omega_d t + \frac{\zeta}{\sqrt{1 - \zeta^2}} \sin \omega_d t \right) = \\ &= 1 - \frac{e^{-\zeta\omega_n t}}{\sqrt{1 - \zeta^2}} \sin \left(\omega_d t + \arctg \frac{\sqrt{1 - \zeta^2}}{\zeta} \right) \quad t \geq 0 \end{aligned} \quad (8.54)$$

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

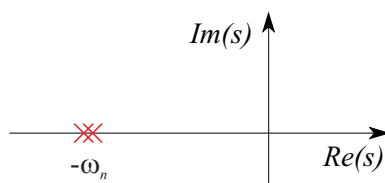
$$y(t) = 1 - \cos \omega_n t \quad t \geq 0 \quad (8.56)$$

Iz te enačbe nazorno vidimo, da je frekvenca ω_n tista frekvenca, s katero bi sistem nihal, 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

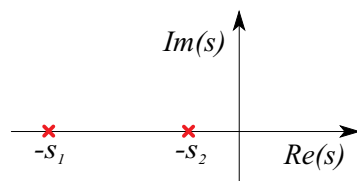
$$\frac{Y(s)}{U(s)} = \frac{\omega_n^2}{(s + \omega_n)^2} \quad (8.57)$$



Slika 8.14: Lega polov pri $\zeta = 1$

Odziv pri vzburjanju 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

$$y(t) = 1 + \frac{\omega_n}{2\sqrt{\zeta^2 - 1}} \left(\frac{e^{-s_1 t}}{s_1} - \frac{e^{-s_2 t}}{s_2} \right) \quad t \geq 0 \quad (8.60)$$

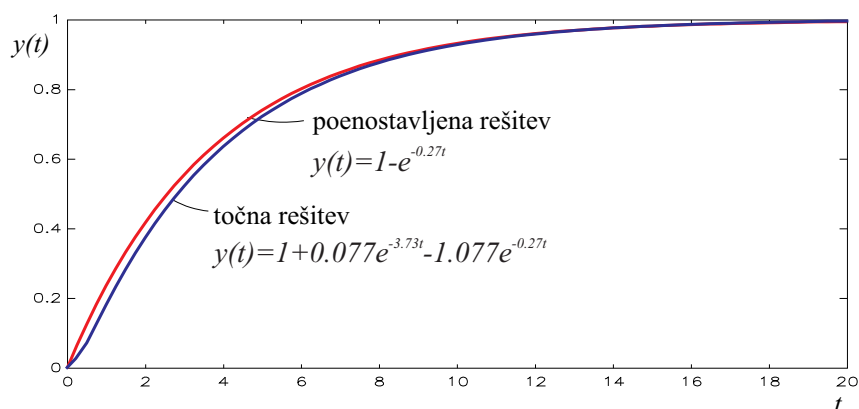
Pri tem je $s_1 = [\zeta + \sqrt{\zeta^2 - 1}]\omega_n$ in $s_2 = [\zeta - \sqrt{\zeta^2 - 1}]\omega_n$. Torej vsebuje odziv dva eksponentno upadajoča člena. Če je $\zeta \gg 1$, en eksponentni člen upada mnogo hitreje, kot drugi, tako da ga lahko zanemarimo (člen z manjšo časovno konstanto ima tudi 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 definiranjem sistema drugega reda, iz katerega izhajamo, ojačenje $Y(0)/U(0) = 1$, je prenosna funkcija poenostavljenega sistema

$$\frac{Y(s)}{U(s)} = \frac{s_2}{s + s_2} = \frac{1}{s} - \frac{1}{s + s_2} \quad (8.61)$$

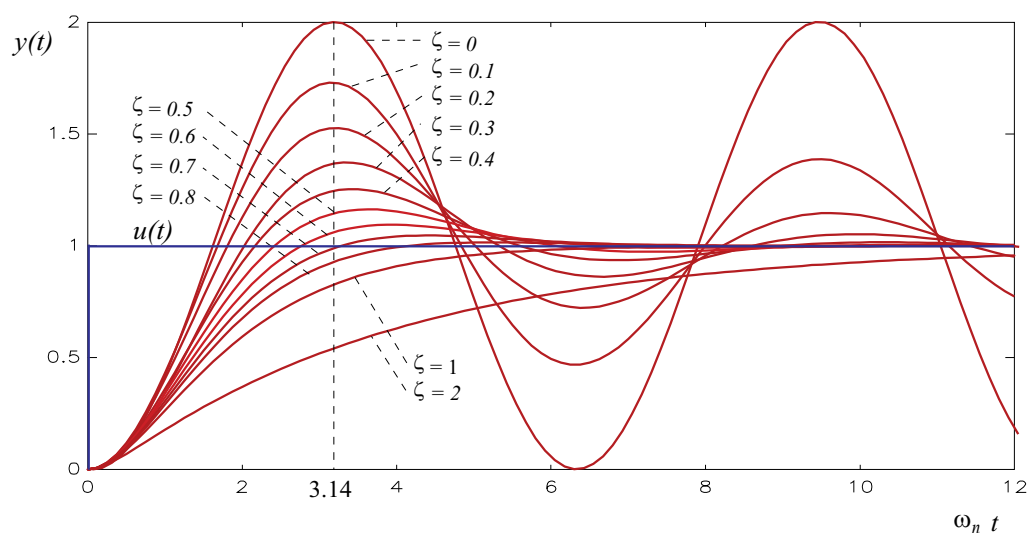
in ustrezen časovni odziv pri vzburjanju z enotino stopnico

$$y(t) = 1 - e^{-s_2 t} = 1 - e^{-(\zeta - \sqrt{\zeta^2 - 1})\omega_n t} \quad t \geq 0 \quad (8.62)$$

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 vzbujujanju 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 aperioidičnimi odzivi pa je najhitrejši odziv pri kritičnem dušenju. Sicer pa aperioidično 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 vzbujanju. 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 nezakasnjenega 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

$$y(t) = k_I t \quad (8.65)$$

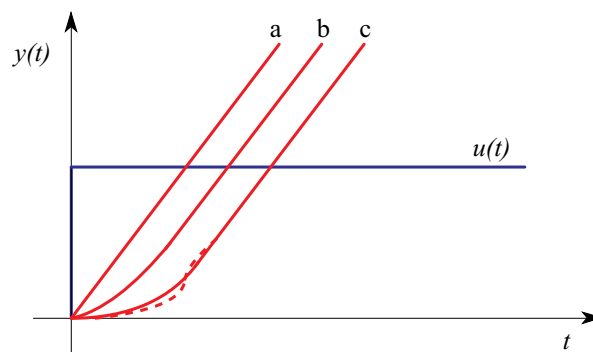
Kadar pa imajo integrirni značaj procesi, se zaradi shranjevalnikov energije pojavi zakasnjeno delovanje (dodatna zakasnitev prvega, drugega, ... višjega reda). Tipičen primer predstavlja prenosna funkcija enosmernega motorja, ki jo podaja enačba (8.37) oz. bločni diagram 8.11b

$$G(s) = \frac{Y(s)}{U(s)} = \frac{K}{s(Js + B)} \quad (8.66)$$

To je integrirni sistem z zakasnitvijo prvega reda ali $I1$ sistem. Odziv motorja (zasuk) pri stopničastem vzbujanju 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 linerano 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

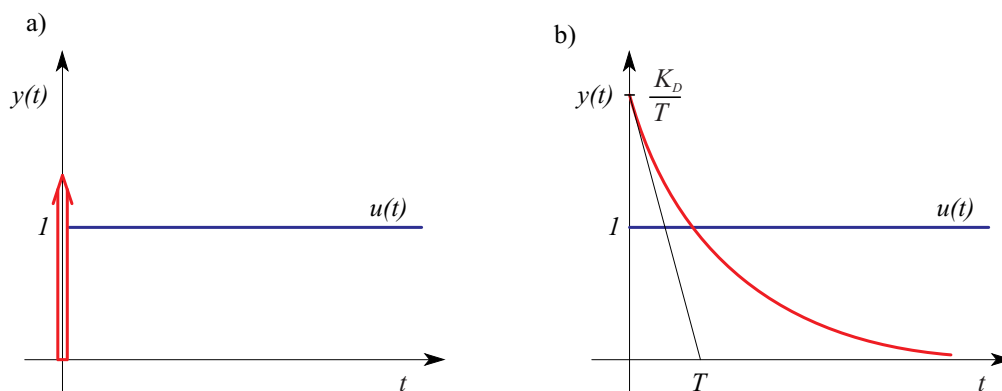
$$G(s) = \frac{Y(s)}{U(s)} = \frac{k_D s}{Ts + 1} \quad (8.72)$$

Časovni odziv sistema $D1$ pri vzbujanju 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 vzbujanju 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

$$y(t) = u(t - T_m) \quad (8.74)$$

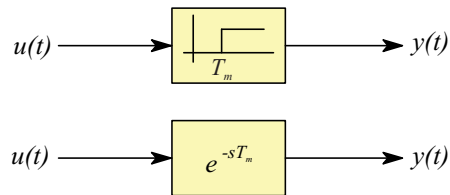
kjer je $y(t)$ izhodna veličina, $u(t)$ vhodna veličina in T_m mrtvi čas. Prenosno funkcijo takega sistema dobimo z upoštevanjem teorema premika in ima obliko

$$G(s) = \frac{Y(s)}{U(s)} = \frac{U(s)e^{-sT_m}}{U(s)} = e^{-sT_m} \quad (8.75)$$

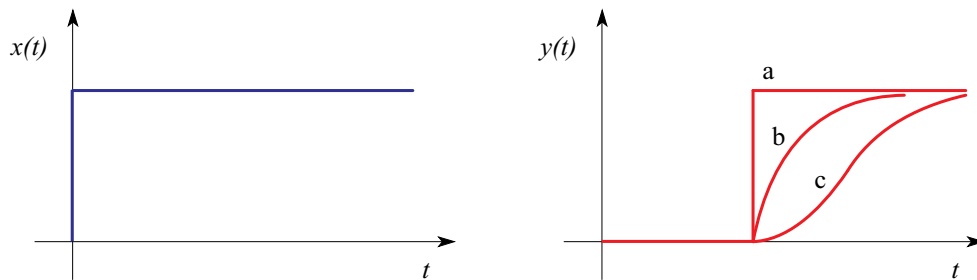
V bločnem diagramu ga predstavimo z eno od ikon, ki sta prikazani na sliki 8.20.

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 vzbužani 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:

$$\begin{array}{cccccc}
 s^n & a_0 & a_2 & a_4 & a_6 & \dots \\
 s^{n-1} & a_1 & a_3 & a_5 & a_7 & \dots \\
 s^{n-2} & b_1 & b_2 & b_3 & b_4 & \dots \\
 s^{n-3} & c_1 & c_2 & c_3 & c_4 & \dots \\
 s^{n-4} & d_1 & d_2 & d_3 & d_4 & \dots \\
 \vdots & \vdots & \vdots & & & \\
 s^3 & e_1 & e_2 & e_3 & & \\
 s^2 & f_1 & f_2 & & & \\
 s^1 & g_1 & & & & \\
 s^0 & h_1 & & & &
 \end{array}$$

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:

$$\begin{array}{ll}
 c_1 = \frac{b_1 a_3 - a_1 b_2}{b_1} & d_1 = \frac{c_1 b_2 - b_1 c_2}{c_1} \\
 c_2 = \frac{b_1 a_5 - a_1 b_3}{b_1} & d_2 = \frac{c_1 b_3 - b_1 c_3}{c_1} \\
 c_3 = \frac{b_1 a_7 - a_1 b_4}{b_1} & \vdots \\
 \vdots &
 \end{array} \quad (8.82)$$

Postopek nadaljujemo tako dolgo, da dobimo $n + 1$ vrstic. Tako dobimo Routhovo shemo trikotne oblike. Vse elemente v eni vrstici lahko zaradi enostavnejšega nadaljnjega 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

$$s^4 + 2s^3 + 3s^2 + 4s + 5 = 0 \quad (8.83)$$

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

$$\begin{array}{ccc} s^4 & 1 & 7 \quad 6 \\ s^3 & 5 & 5 \\ s^2 & 6 & 6 \\ s^1 & \lim_{\varepsilon \rightarrow 0} \varepsilon & \\ s^0 & 6 & \end{array}$$

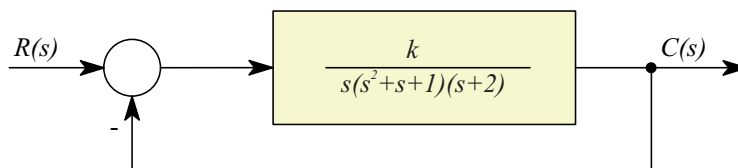
Če je $\varepsilon > 0$, ni spremembe predznaka v prvi koloni, sistem je stabilen. Če je $\varepsilon < 0$, imamo dve spremembi znaka, torej dva korena v desnem delu ravnine s . Ker pa je $\lim_{\varepsilon \rightarrow 0} \varepsilon = 0$, to pomeni, da sta dva korena na imaginarni osi (konjugirano kompleksni 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

$$0 < k < \frac{14}{9} \quad (8.89)$$

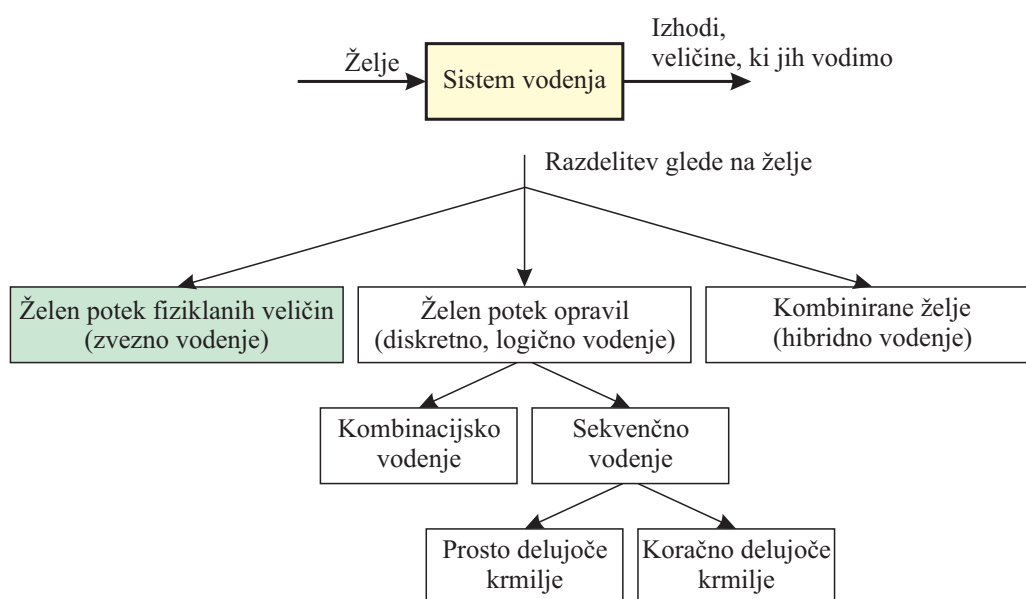
Pri ojačenju $\frac{14}{9}$ sistem nedušeno niha.

□

9.

Vodenje sistemov

Uvodne značilnosti sistemov vodenja smo spoznali v poglavju 1. S sistemi vodenja želimo doseči želeno obnašanje. Zato lahko tudi glede na različnost želja naredimo pomembno razdelitev sistemov vodenja, kot prikazuje slika 9.1.



Slika 9.1: Razdelitev sistemov vodenja glede na želje

Razlikujemo med zveznim, diskretnim-logičnim in hibridnim vodenjem.

Zvezno vodenje

Pri zveznem vodenju je potrebno doseči želeni potek fizikalnih veličin (npr. temperatura, tlak, pomik, ...). Vodenje je zlasti značilno za procesno industrijo. Prevladujejo zvezni ali vzorčeni signali. Sem spada eden od najzahtevnejših načinov vodenja - regulacija. Vodenje se običajno izvede s pomočjo industrijskih regulatorjev ali procesnih računalnikov. V nadaljevanju se bomo pretežno posvetili prav regulaciji.

Diskretno, logično vodenje

Tu je želja povezana predvsem z učinkovitim razvrščanjem opravil, sestavljanjem, ...itd. Bolj strokovno govorimo o želenem zaporedju diskretnih stanj. Vodenje je zlasti značilno za izdelčno ali kosovno industrijo. V takih sistemih prevladujejo binarni signali. Vodenje običajno izvedeno s programirljivimi logičnimi krmilniki (PLK, angl. Programmable logical controller). To vrsto vodenja delimo na kombinacijsko in sekvenčno vodenje.

Kombinacijsko vodenje. Krmilje realiziramo z osnovnimi logičnimi funkcijami. Logične funkcije preslikajo vhode v izhode.

Sekvenčno vodenje. Tu razen osnovnih logičnih funkcij potrebujemo še pomnilniške člene - npr. bistabilna vezja. Na ta način lahko programiramo neko sekvenco opravil. Poznamo prosto delujoča krmilja in koračno delujoča krmilja. Pri prosto delujočih krmiljih se na vhodih lahko pojavijo poljubne kombinacije binarnih signalov v poljubnem zaporedju. Pri koračno delujočih krmiljih pa so možne le določene kombinacije vhodov, ki se menjavajo v določenem zaporedju.

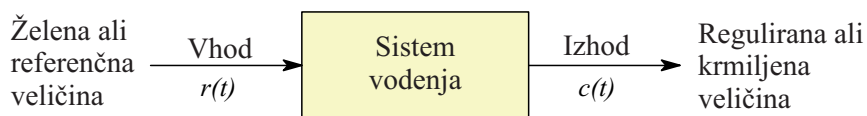
Hibridno vodenje

Tu so želje sestavljene: doseči želimo nek potek fizikalnih veličin, pa tudi optimalno razporejanje opravil. Za to vrsto vodenja so tipični šaržni procesi.

Omenjena delitev se tudi lepo ujema z delitvijo procesov v podpoglavju 2.3.

□

Kadar proučujemo dinamične lastnosti v sistemih vodenja (npr. dinamične lastnosti različnih regulacijskih algoritmov), uporabljamo bločne diagrame. Vsak sistem vodenja lahko ponazorimo z blokom, kot ga prikazuje slika 9.2.

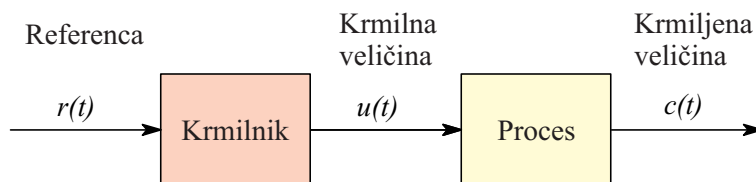


Slika 9.2: Sistem vodenja

Izhodu sistema pravimo glede na vrsto sistema regulirana ali krmiljena veličina, vходу pa želena ali referenčna veličina. Pogosta zahteva sistema vodenja je ta, da se regulirana ali krmiljena veličina čim boljše ujema z referenčno oz. želeno veličino ne glede na motnje, ki vplivajo na sistem vodenja.

9.1 Odprtozančni sistem vodenja

V številnih primerih lahko uporabimo odprtozančni sistem vodenja (krmilni sistem), t.j. vodenje brez povratne zanke. Prikazuje ga slika 9.3.



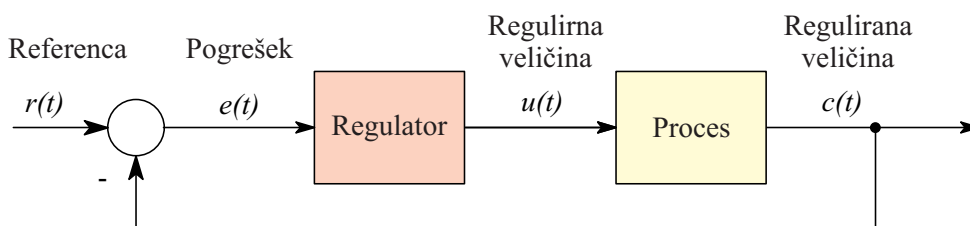
Slika 9.3: Bločni diagram krmiljenja

V tem primeru referenca $r(t)$ (referenčna veličina, želena veličina) deluje na krmilnik, ki s pomočjo krmilne veličine $u(t)$ zagotavlja, da je izhod procesa oz. krmiljena veličina v določeni korelaciji z referenčno veličino.

Tipičen primer krmiljenja predstavlja ogrevanje prostora z grelom in časovnim krmilnikom, ki periodično vključuje in izključuje grelo. Uporabnik lahko le nastavi razmerje med časom vklopa in izklopa. Z določenimi izkušnjami lahko uporabnik nastavi razmerje, ki omogoča približno želeno temperaturo. Vendar takšno vodenje ne upošteva spreminjanja zunanje temperature, možne motnje v sistemu kot npr. odpiranje vrat ali oken. Zato je jasno, da je kvaliteta vodenja slaba, temperatura pa le redko enaka želena.

9.2 Zaprtozančni sistem vodenja

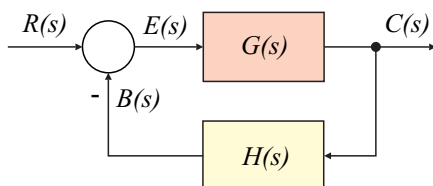
Kar manjka krmilnemu sistemu za točnejše in prilagodljivejše delovanje, je povratna zanka iz izhodne proti vhodni ali referenčni veličini. Sistem z dodano povratno zanko, ki ga prikazuje slika 9.4, imenujemo zaprtozančni sistem vodenja ali regulacijski sistem.



Slika 9.4: Bločni diagram zaprtozančnega regulacijskega sistema

Da dobimo točnejšo regulirano veličino $c(t)$, jo moramo primerjati z referenco $r(t)$, regulator pa definira ustrezno regulirno veličino na osnovi pogreška tako, da le-ta zmanjšuje pogrešek. Človek je prav gotovo najkompleksnejši tovrstni regulacijski sistem, saj vsebuje nešteto povratnozančnih struktur, ki omogočajo kompleksne operacije in koordinirano delovanje.

Bločno shemo linearne zaprtozančnega sistema posplošimo tako, kot kaže slika 9.5.



Slika 9.5: Posplošeni bločni diagram zaprtozančnega regulacijskega sistema

Bloka, ki na sliki 9.4 opisujeta regulator in proces in se nahajata v direktni veji, združimo v en blok $G(s)$. V povratno zanko pa smo dodali blok $H(s)$, ki lahko modelira merilni sistem (senzor), včasih pa je to del regulacijskega algoritma.

9.3 Sledilno delovanje regulacijskega sistema

Sledilno delovanje je eno najpogostejše zahtevanih delovanj regulacijskih sistemov. Pri tem mora biti regulacijski sistem načrtan tako, da regulirana veličina čim bolj verno sledi referenčni veličini. Slika 9.4 predstavlja bločno shemo takega sistema. Tipični primeri sledilne regulacije so: zagotovitev hitrostnega profila pri delovanju dvigala, zagotovitev določenega temperaturnega profila pri številnih procesih v kemični industriji, sledenje strelne naprave tarči, regulacije zasuka vesoljskih ladij itd.

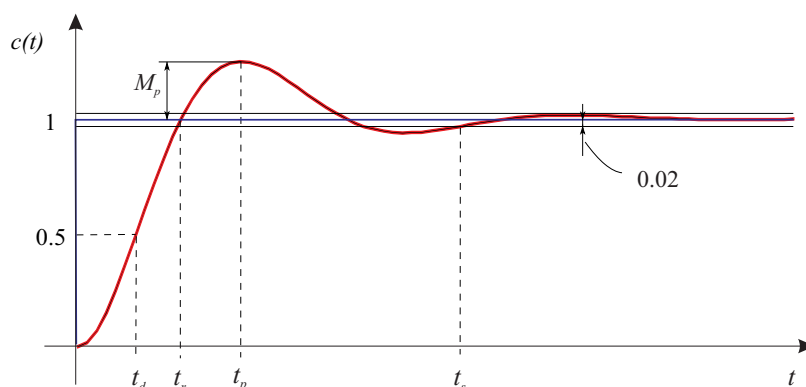
Pri načrtovanju sledilne regulacije je manj pomembno, kako regulacijski sistem deluje pri morebitnih motilnih signalih.

9.3.1 Pokazatelji kvalitete pri sledilni regulaciji

Za idealno sledenje bi seveda želeli, da je zaprtozančna prenosna funkcija

$$\frac{C(s)}{R(s)} = 1 \quad (9.1)$$

kar pa zaradi masne ali energijske vztrajnosti sistemov (zaradi zakasnitev) seveda ni možno doseči. Kaj pa lahko dosežemo, prikazuje slika 9.6. Na njej sta tipična poteka referenčne veličine, ki se v času $t = 0$ spremeni za konstantno vrednost, in ustrezne regulirane veličine.



Slika 9.6: Odziv regulacijskega sistema na stopničasto vzbujanje

Dobro delovanje regulacijskega sistema v številnih primerih ponazorimo (modeliramo) s pomočjo sistema 2. reda (glej enačbo 8.42 in sliko 8.17) z dušilnim

koeficientom ζ nekaj manjšim od 1 (podkritično dušenje), torej ima regulacijski sistem en konjugirano kompleksni par polov. Načrtovanje regulacijskih sistemov z željo, da dosežemo ustrezno lego dominantnega para zaprtizančnih polov, je eden od najpogostejših pristopov (angl. pole placement). Tudi če je red celotnega regulacijskega sistema višji, običajno smatramo, da je en par polov dominanten, drugi poli pa so bolj oddaljeni od imaginarne osi in torej nimajo izdatnejšega vpliva na dinamiko prehodnega pojava. Ker pa je predpis delovanja regulacijskega sistema s parametroma ζ in ω_n (oz. neposredno z lego dominantnega para polov) teoretično bolj zahteven, se je uveljavil bolj inženirski način s t.i. pokazatelji kvalitete, ki pa so seveda v določeni povezavi z lego polov (oz. ζ in ω_n).

Inženirski pokazatelji, ki opisujejo učinkovitost delovanja regulacijskega sistema, so naslednji:

t_d - čas zakasnitve. Čas zakasnitve je čas, v katerem regulirana veličina prvič doseže 50% končne vrednosti.

t_r - čas vzpona. Čas vzpona je čas, v katerem odziv sistema naraste iz 10% na 90% (pri nadkritično dušenih sistemih), ali iz 0% na 100% končne vrednosti (pri podkritično dušenih sistemih).

t_p - čas maksimalnega prevzpona. Čas maksimalnega prevzpona je čas, v katerem odziv doseže maksimalni prevzpon.

M_p - maksimalni (procentni) prevzpon. Maksimalni prevzpon je za odziv na sliki 9.6 določen z razliko med maksimalno vrednostjo odziva in vrednostjo 1. Če ustaljeno stanje ni določeno z vrednostjo ena, potem je prevzpon podan z enačbo (ob predpostavki, da lahko regulacijski sistem modeliramo s sistemom 2. reda)

$$M_p = \frac{c(t_p) - c(\infty)}{c(\infty)} = e^{-\zeta \frac{\pi}{\sqrt{1-\zeta^2}}} \quad 0 \leq \zeta < 1 \quad (9.2)$$

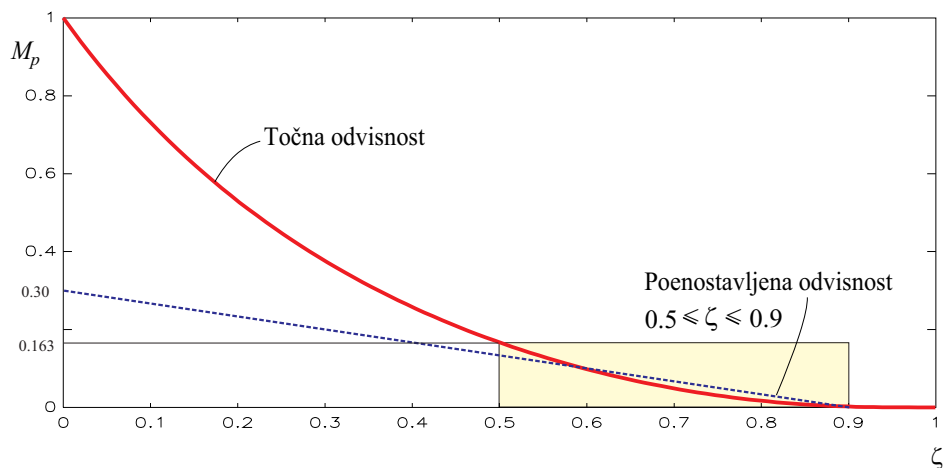
Če želimo prevzpon izraziti v procentih, je potrebno izraz pomnožiti s 100%

$$M_p[\%] = e^{-\zeta \frac{\pi}{\sqrt{1-\zeta^2}}} \cdot 100\% \quad (9.3)$$

Za $0.5 \leq \zeta \leq 0.9$ lahko izraz 9.2 aproksimiramo z linearno odvisnostjo

$$M_p \doteq 0.3(1 - \frac{\zeta}{0.9}) \quad (9.4)$$

Točno in poenostavljeno odvisnost M_p od ζ prikazuje slika 9.7.



Slika 9.7: Odvisnost prevzpona od dušilnega koeficienta

Vrednost prevzpona omogoča učinkovito vrednotenje relativne stabilnosti sistema.

t_s - umiritveni čas. Umiritveni čas je čas, ki ga potrebuje odziv, da doseže in ostane znotraj tolerančnega področja okoli ustaljene vrednosti (običajno $\pm 2\%$). Umiritveni čas je odvisen od največje časovne konstante regulacijskega sistema.

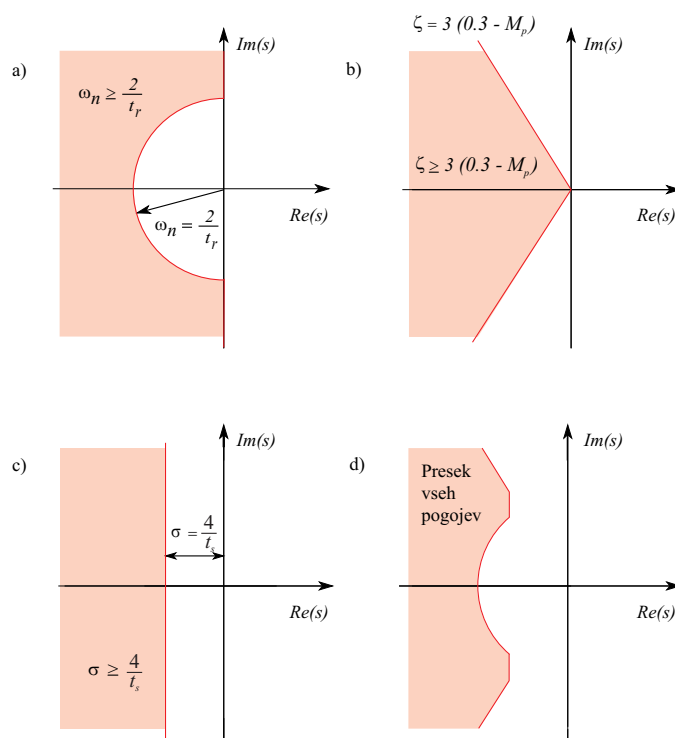
Pri načrtovanju regulacijskih sistemov predpišemo inženirske pokazatelje. Iz pokazateljev izračunamo lego polov. ζ je običajno med 0.5 (temu ustreza $M_p = 0.163$ oz. 16,3%) in 0.9 ($M_p = 0.015$ oz. 1,5%). Vrednosti $\zeta < 0.5$ povzročijo prevelik prevzpon, vrednosti $\zeta > 0.9$ pa prepočasen prehodni pojav. Želimo doseči čas vzpona $\leq t_r$, maksimalni prevzpon $\leq M_p$ in umiritveni čas $\leq t_s$, če so t_r , M_p in t_s predpisane vrednosti.

Natančne relacije med matematičnimi (ζ in ω_n) in inženirskimi pokazatelji se dajo izračunati, a so kar zapletene. Povezavo med M_p in ζ podaja enačba 9.2 in poenostavljena odvisnost 9.4. Veljajo poenostavljene zveze med matematičnima pokazateljema ζ in ω_n ter inženirskimi pokazatelji

$$\omega_n \geq \frac{2}{t_r}$$

$$\begin{aligned}\zeta &\geq 3(0.3 - M_p) \\ \sigma = \zeta\omega_n &\geq \frac{4}{t_s}\end{aligned}\tag{9.5}$$

S pomočjo teh neenačb lahko določimo področje lege želenih polov v ravnini s . Jasno je, da mora področje ležati v levem delu ravnine s , saj moramo najprej doseči stabilno delovanje. Ustrezno področje prikazuje slika 9.8.

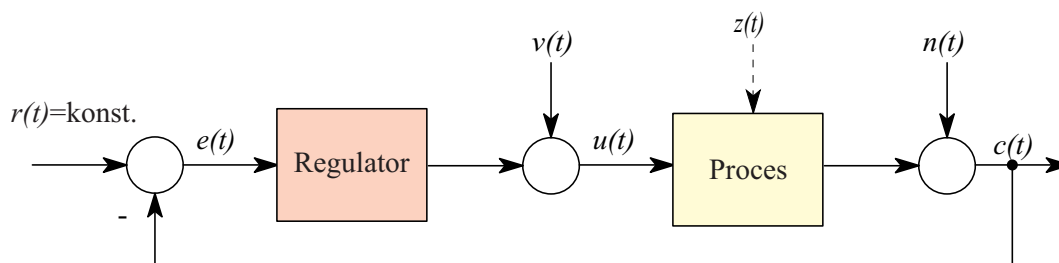


Slika 9.8: Območje za določitev lege polov: a) zahteva za čas vzpona
b) zahteva za prevzpon
c) zahteva za umiritveni čas
d) presek vseh zahtev

9.4 Regulacijsko delovanje regulacijskega sistema

Kljub temu, da je načrtovanje sledilnega regulatorja običajno bolj podrobno obravnavano v literaturi, pa je v regulacijskih sistemih običajno bolj pomembno, da le-ta učinkovito odpravlja motnje, oz. da le-te čim manj vplivajo na regulirano

veličino. Takemu načinu delovanja regulacijskega sistema pravimo regulacijsko delovanje. Motnje lahko vstopajo v regulacijsko zanko pri regulirni veličini, pri regulirani veličini, ali pa nekje vmes. Ustrezno bločno shemo prikazuje slika 9.9.



Slika 9.9: Bločna shema regulacijskega sistema v prisotnosti motenj

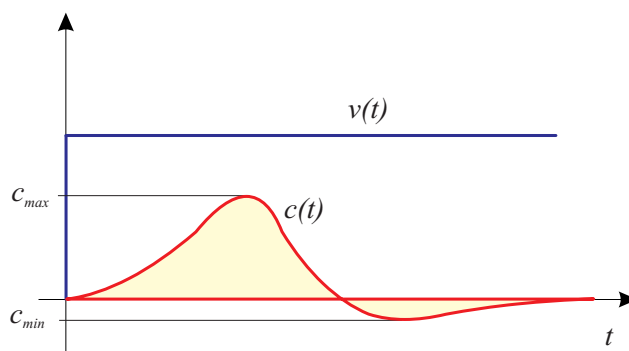
Učinkovitost reagiranja na spremenjen referenčni signal nas pri tej regulaciji ne zanima, ker predpostavimo, da je želena vrednost nastavljena na neko konstantno vrednost in se ne spreminja. Tipični primeri so: regulacija temperature v prostoru, ko je želena temperatura konstantna, regulator pa mora čimbolj izločiti vplive motenj (sprememba zunanje temperature, odpiranje oken, vrat,...). Pri tem naj omenimo, da je za načrtovanje regulatorja bistveno, ali delujejo motnje na vходу procesa ($v(t)$), na izhodu ($n(t)$), ali pa nastopajo nekje vmes ($z(t)$). Če smo načrtali sledilni regulator, le-ta optimalno odpravlja tudi motnjo na izhodu (če je iste oblike kot referenčni signal), saj $r(t)$ in $n(t)$ delujeta v isti točki regulacijskega sistema. Motnje na vходу procesa ali motnje, ki delujejo v notranjosti procesa, pa zahtevajo regulacijsko delovanje regulacijskega sistema, torej drugačen regulator.

Slika 9.10 prikazuje tipična signala za motnjo $v(t)$ na regulirni veličini (konstantna oz. stopničasta sprememba) in ustrezno regulirano veličino $c(t)$.

Idealno delovanje regulatorja bi seveda bilo, da se vpliv motnje sploh ne bi opazil, oz. da bi veljala prenosna funkcija

$$\frac{C_v(s)}{V(s)} = 0 \quad (9.6)$$

To seveda zaradi zakasnitev v dinamiki procesa ni možno doseči. Vendar dober regulator izdatno zaduši vpliv motenj in jih v čim krajšem času izniči. Z nastavitvijo parametrov regulatorja moramo doseči čim manjše vrednosti c_{max} in c_{min} , pa tudi približno enake, saj to pomeni, da bi motnja povzročila spremembo regulirane veličine v približno enakem obsegu v obe smeri (npr. temperatura bi pri željeni vrednosti $20^\circ C$ ostala v področju med $19^\circ C$ in $21^\circ C$). Primeren kriterij je tudi minimiziranje ploščine pod krivuljo $|c(t)|$ ali $c^2(t)$, kot določa integral



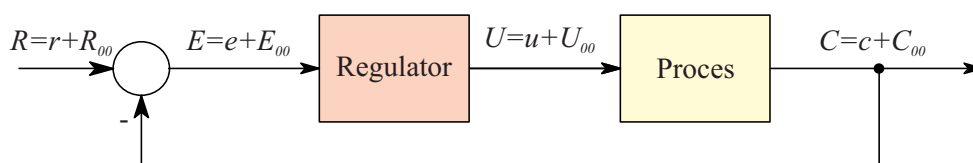
Slika 9.10: Odziv na motnjo na regulirni veličini v regulacijskem sistemu

$$C_1 = \int_0^\infty |c(t)| dt \quad C_2 = \int_0^\infty c^2(t) dt \quad (9.7)$$

Pri načrtovanju regulacijskega delovanja je torej manj pomembno, kako regulacijski sistem sledi morebitnim spremembam reference.

9.5 Obravnava regulacijskega sistema v delovni točki

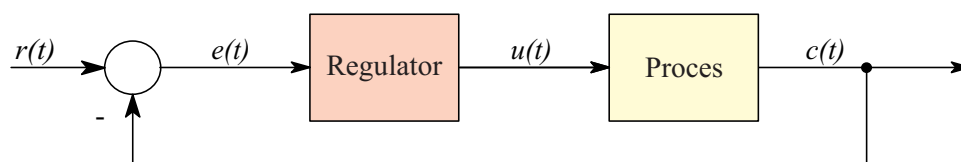
Običajno si lahko vse signale v regulacijskih sistemih predstavljamo tako, da so sestavljeni iz nekih delovnih, enosmernih vrednosti in iz manjših sprememb v okolici delovne točke. Ustrezne oznake prikazuje slika 9.11.



Slika 9.11: Obravnava regulacijskega sistema v okolici delovne točke

R_{00} , E_{00} , U_{00} in C_{00} so enosmerne vrednosti signalov oz. vrednosti signalov, ki v regulacijskem sistemu določajo delovno točko. Tako vrednost reference R_{00} povzroči pogrešek E_{00} , regulirno veličino U_{00} in regulirano veličino C_{00} . Običajno velja (če ni ustaljenega pogreška) $E_{00} = R_{00} - C_{00} = 0$. Nadalje bomo obravnavali samo take regulacijske sisteme, ki se v bližnji okolici delovne točke vedejo

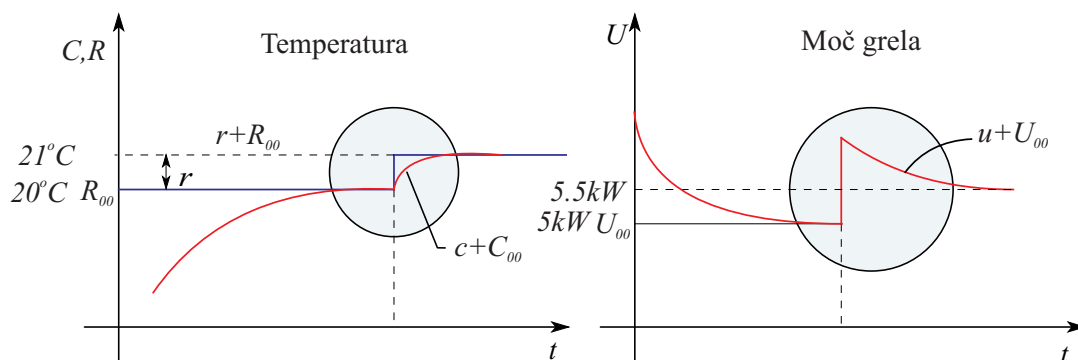
linearno. Zaradi zakona superpozicije lahko od vseh nastopajočih signalov odštejemo vrednosti delovnih točk ($R_{00}, E_{00}, C_{00}, U_{00}$) in dobimo regulacijski sistem, ki ga bomo v nadaljevanju obravnavali in ga prikazuje slika 9.12. Takemu modelu regulacijskega sistema pravimo tudi deviacijski model.



Slika 9.12: Deviacijski model regulacijskega sistema

Pogosto bomo pri nadaljnji obravnavi ob takih oznakah, kot jih prikazuje slika 9.12, imeli v mislih le spremembe signalov v okolici delovne točke in torej vrednosti delovne točke ne bomo jemali v obzir.

Za boljše razumevanje si pogledjmo naslednji zgled. V prostoru imamo nastavljeno želeno temperaturo 20°C . Ta temperatura se vzpostavi po določenem prehodnem pojavu in zahteva moč grela 5 kW . Če spremenimo želeno temperaturo za 1°C , se po prehodnem pojavu moč grela poveča za 500 W . Prehodna pojava prikazuje slika 9.13.



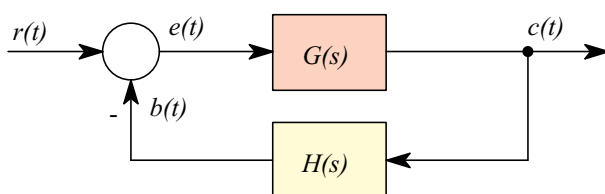
Slika 9.13: Prehodna pojava pri regulaciji temperature v prostoru

Pri nadaljnji obravnavi regulacijskih sistemov bomo imeli pogosto v mislih le prehodni pojav, ki je na sliki 9.13 obkrožen. Velikokrat bomo analizirali sisteme pri enotini skočni spremembi referenčnega signala (enotina stopnica). To ne pomeni, da se je referenčni signal dejansko spremenil iz 0 na 1 problemske enote, ampak v delovni točki za neko konstantno vrednost, ki pa smo jo normirali na vrednost ena zaradi preglednejše in enostavnejše nadaljnje obravnave, kar ne predstavlja nikakršne izgube na splošnosti.

9.6 Ustaljeno stanje regulacijskih sistemov

Regulacijski sistem ni potrebno analizirati le v času prehodnega pojava ampak tudi v ustaljenem stanju. Običajno želimo, da je po določeni spremembi referenčnega signala ali po nastopu motnje pogrešek v ustaljenem stanju enak nič ali vsaj čim manjši, ker naj bi bil regulacijski sistem čim bolj točen.

Za analizo ustaljenega stanja bomo uporabili posplošeni bločni diagram, kot ga prikazuje slika 9.14.



Slika 9.14: Posplošeni bločni diagram zaprtozančnega regulacijskega sistema

Signal pogreška je podan z enačbo

$$\begin{aligned} e(t) &= r(t) - b(t) \quad \text{oz.} \\ E(s) &= R(s) - B(s) = R(s) - H(s)C(s) \end{aligned} \quad (9.8)$$

Pogrešek v ustaljenem stanju je definiran z enačbo

$$e_{ss} = \lim_{t \rightarrow \infty} e(t) = \lim_{s \rightarrow 0} sE(s) \quad (9.9)$$

Pogrešek pa lahko s pomočjo Laplace-ove transformacije izračunamo le, če $sE(s)$ ne vsebuje polov na imaginarni osi ali v desni polravnini. S pomočjo slike 9.14 lahko zapišemo enačbo

$$E(s) = \frac{R(s)}{1 + G(s)H(s)} \quad (9.10)$$

oz. pogrešek v ustaljenem stanju

$$e_{ss} = \lim_{s \rightarrow 0} \frac{sR(s)}{1 + G(s)H(s)} \quad (9.11)$$

Iz enačbe (9.11) vidimo, da je pogrešek v ustaljenem stanju odvisen od referenčne veličine ter od odprtozančne prenosne funkcije $G(s)H(s)$.

Zapišimo odprtozančno prenosno funkcijo $G(s)H(s)$ v obliki, iz katere sta razvidni vrsta sistema in ojačenje sistema.

$$G(s)H(s) = \frac{K}{s^j} \frac{B(s)}{A(s)} \quad (9.12)$$

K je konstanta (ojačenje), j pa je število polov v koordinatnem izhodišču oz. vrsta sistema. Preostanek $\frac{B(s)}{A(s)}$ je proporcionalni sistem (torej brez polov ali ničel v koordinatnem izhodišču) z ojačenjem 1, torej velja

$$B(0) \neq 0 \quad A(0) \neq 0 \quad \lim_{s \rightarrow 0} \frac{B(s)}{A(s)} = 1 \quad (9.13)$$

Vrsta sistema odločilno vpliva na pogrešek v ustaljenem stanju. Prav tako pa na pogrešek vpliva tudi vrsta referenčnega signala.

Pogrešek v ustaljenem stanju pri stopničastem referenčnem signalu

Če je referenčni signal stopničaste oblike velikosti R_0 , je ustrezni Laplace-ov transform $\frac{R_0}{s}$ in enačba (9.11) se spremeni v obliko

$$e_{ss} = \lim_{s \rightarrow 0} \frac{sR(s)}{1 + G(s)H(s)} = \frac{R_0}{1 + \lim_{s \rightarrow 0} G(s)H(s)} \quad (9.14)$$

Definirajmo konstanto pozicijskega pogreška kot

$$K_p = \lim_{s \rightarrow 0} G(s)H(s) \quad (9.15)$$

Pogrešek v ustaljenem stanju izrazimo s konstanto pozicijskega pogreška z upoštevanjem enačb (9.14) in (9.15)

$$e_{ss} = \frac{R_0}{1 + K_p} \quad (9.16)$$

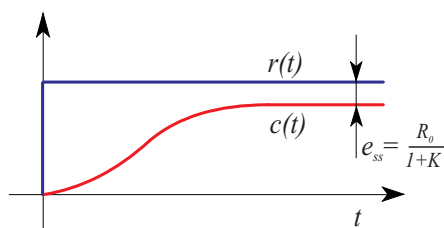
Pri stopničastem referenčnem signalu je torej pogrešek odvisen od konstante pozicijskega pogreška. Za sistem ničte vrste ($j = 0$, proporcionalni sistem) velja

$$K_p = \lim_{s \rightarrow 0} \frac{K B(s)}{A(s)} = K \quad e_{ss} = \frac{R_0}{1 + K} \quad (9.17)$$

Za sistem prve ali višje vrste ($j \geq 1$) pa je

$$K_p = \lim_{s \rightarrow 0} \frac{K B(s)}{s^j A(s)} = \infty \quad e_{ss} = 0 \quad (9.18)$$

Iz opisane analize sledi, da nastopi pogrešek v ustaljenem stanju pri stopničastem vhodnem signalu v primeru, če odprtozanka prenosna funkcija ne vsebuje vsaj enega pola v koordinatnem izhodišču. V takem primeru zmanjšamo pogrešek le z ustreznim povečanjem ojačenja K , kar pa običajno povzroči stabilnostne probleme. Značilni odziv prikazuje slika 9.15



Slika 9.15: Odziv proporcionalnega sistema na stopničasti referenčni signal

Tudi s povečanjem števila polov v koordinatnem izhodišču se poslabšajo stabilnostne razmere. Zato je potrebno izbrati čim nižjo vrsto, ki še zagotavlja ničelni pogrešek v ustaljenem stanju (v obravnavanem primeru 1. vrsta).

Pogrešek v ustaljenem stanju pri linearno naraščajočem referenčnem signalu

Laplace-ov transform linearno naraščajočega signala $r(t) = R_0 t$ ($t \geq 0$) je $R(s) = \frac{R_0}{s^2}$, zato je pogrešek

$$e_{ss} = \lim_{s \rightarrow 0} \frac{sR(s)}{1 + G(s)H(s)} = \frac{R_0}{\lim_{s \rightarrow 0} sG(s)H(s)} \quad (9.19)$$

Definirajmo konstanto hitrostnega pogreška kot

$$K_v = \lim_{s \rightarrow 0} sG(s)H(s) \quad (9.20)$$

Z upoštevanjem te definicije je pogrešek v ustaljenem stanju

$$e_{ss} = \frac{R_0}{K_v} \quad (9.21)$$

Za sistem ničte vrste ($j = 0$) velja

$$K_v = \lim_{s \rightarrow 0} \frac{s K B(s)}{A(s)} = 0 \quad e_{ss} = \infty \quad (9.22)$$

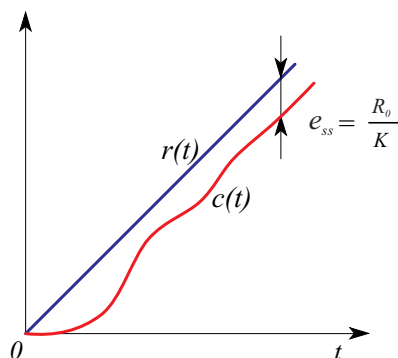
in za sistem prve vrste ($j = 1$)

$$K_v = \lim_{s \rightarrow 0} \frac{s K B(s)}{s A(s)} = K \quad e_{ss} = \frac{R_0}{K} \quad (9.23)$$

Za sistem druge ali višje vrste ($j \geq 2$) pa velja

$$K_v = \lim_{s \rightarrow 0} \frac{s K B(s)}{s^j A(s)} = \infty \quad e_{ss} = 0 \quad (9.24)$$

Analiza je pokazala, da sistem ničte vrste ni zmožen slediti linearno naraščajočemu referenčnemu signalu. Sistem prve vrste lahko sledi referenčnemu signalu s konstantnim pogreškom, ki je sorazmeren hitrosti reference (R_0) in obratnoso-razmeren ojačenju (K). Slika 9.16 prikazuje tipičen odziv sistema prve vrste na linearno naraščajoči referenčni signal pri enotini povratni zanki. Sistemi višje vrste nimajo pogreška v ustaljenem stanju.



Slika 9.16: Odziv sistema prve vrste na linearno naraščajoči referenčni signal pri enotini povratni zanki

Tabela 9.1 prikazuje zveze med konstantami pogreškov, pogreški v ustaljenem stanju, vrstami referenčnih signalov ter vrstami sistemov.

Tabela 9.1: Konstante pogreškov in pogreški v ustaljenem stanju

	Vrsta vhodnega signala		Stopničasta referenca	Lin. narašč. referenca
Vrsta sistema (j)	K_p	K_v	$e_{ss} = \frac{R_0}{1+K_p}$	$e_{ss} = \frac{R_0}{K_v}$
0	K	0	$\frac{R_0}{1+K}$	∞
1	∞	K	0	$\frac{R_0}{K}$
2	∞	∞	0	0

Poudariti je potrebno, da obravnavana izraza konstanti pozicijskega in hitrostnega pogreška ne pomenita, da govorimo o poziciji in hitrosti kot o fizikalnih veličinah, ampak imamo v mislih neko poljubno regulirano veličino in njeni prvi odvod. Uporabljeni izrazi so zgodovinsko pogojeni z dejstvom, da so bili prvi regulacijski sistemi pozicijski servosistemi. Glavni smisel definiranih konstant pogreškov je v tem, da konstanta pozicijskega pogreška definira pogrešek v ustaljenem stanju pri

stopničastem referenčnem signalu (konstantna pozicija referenčnega signala), konstanta hitrostnega pogreška pa definira pogrešek v ustaljenem stanju pri linearno naraščajočem referenčnem signalu (konstantna hitrost referenčnega signala).

9.7 Stabilnost regulacijskih sistemov

Pri načrtovanju kakršnega koli regulacijskega sistema je prva in najpomembnejša zahteva, da je le-ta stabilen. Zato je preučevanje stabilnosti osrednjega pomena pri analizi in načrtovanju regulacijskih sistemov.

Zaprtozančna prenosna funkcija sistema, ki ga prikazuje slika 9.14 je

$$G_r(s) = \frac{C(s)}{R(s)} = \frac{G(s)}{1 + G(s)H(s)} \quad (9.25)$$

Za stabilnost so odločilne lege polov zaprtozančne prenosne funkcije $G_r(s)$ oz. lege korenov karakteristične enačbe

$$1 + G(s)H(s) = 0 \quad (9.26)$$

Osnovna definicija stabilnosti regulacijskega sistema je enaka, kot pri splošnih sistemih:

Regulacijski sistem je stabilen, če je izhodni signal omejen pri kakršnem koli omejenem vhodnem signalu (BIBO stabilnost; bounded input, bounded output).

Da se dokazati, da je sistem po tej definiciji stabilen natanko tedaj, če ima zaprtozančni sistem vse pole (oz karakteristična enačba vse korene) v levem delu ravnine s .

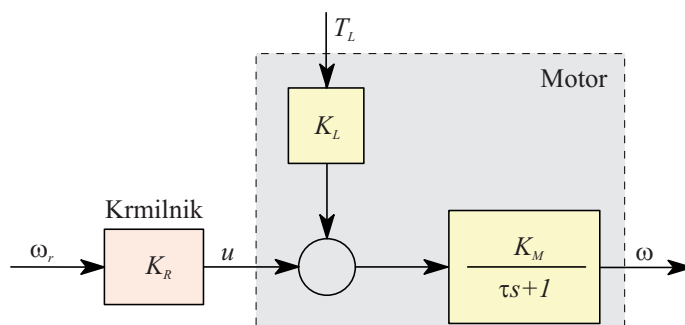
Čeprav smo predstavili stabilnost pri referenčnem vhodnem signalu, je stabilnost linearne regulacijskega sistema lastnost sistema in je neodvisna od tega, kje je sistem vzbuja. To je razvidno tudi iz dejstva, da je karakteristični polinom neodvisen od mesta izbire vhodnega in izhodnega signala v regulacijski zanki.

Stabilnost sistema lahko analitično določamo s pomočjo računalniških programov,

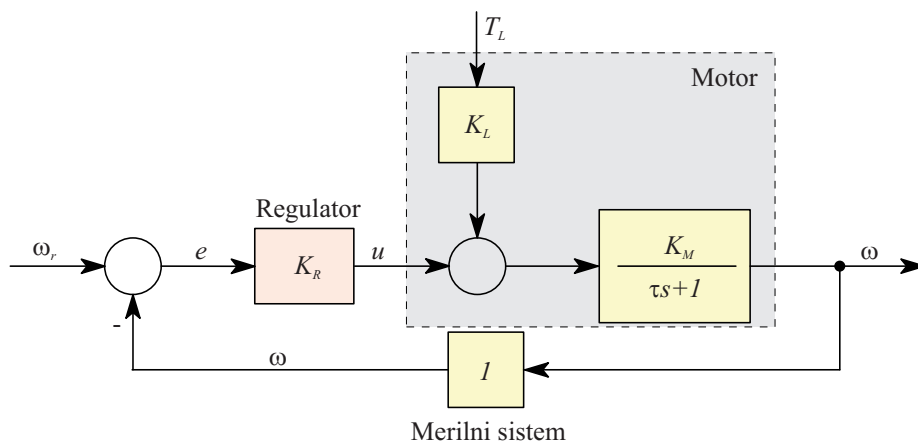
ki izračunajo korene karakteristične enačbe, s pomočjo Routh-ovega stabilnostnega kriterija ali s pomočjo številnih drugih metod v prostoru stanj ali v frekvenčnem prostoru.

9.8 Nekateri preostali učinki povratne zanke

Razen omenjenih lastnosti si oglejmo še nekatere dobre lastnosti povratnoznančnih sistemov in prednosti regulacijskih sistemov pred krmilnimi sistemi. Pri tem si bomo pomagali z bločnima diagramoma na slikah 9.17 in 9.18.



Slika 9.17: Bločni diagram krmiljenja hitrosti



Slika 9.18: Bločni diagram regulacije hitrosti

Obe sliki vsebujeta model enosmernega motorja. Vhod v model je enosmerna napetost na rotorskem vzbujujanju u , izhod pa kotna hitrost ω . V model je vključena tudi bremenska motnja T_L . Model enosmernega motorja je sistem

prvega reda z ojačenjem K_M in s časovno konstanto τ . Slika 9.17 prikazuje krmilni sistem, slika 9.18 pa regulacijski sistem za regulacijo hitrosti vrtenja. V obeh primerih smo uporabili proporcionalni krmilnik oz. regulator. Analiza obeh shem nas pripelje do naslednjih zaključkov:

Robustnost na spremembe parametrov

Regulacijski sistem je za faktor $1 + K_R K_M$ manj občutljiv na spremembe ojačenja K_M kot krmilni sistem. Podobno velja tudi za druge parametre. Pravimo, da je regulacijski sistem bolj robusten. Lastnost je zelo pomembna, saj se parametri procesa med obratovanjem pogosto spreminjajo. Faktor $\frac{1}{1 + K_R K_M}$ se imenuje občutljivostni faktor.

Primer: če je $1 + K_R K_M = 10$, potem npr. 10% sprememba na parametru K_M povzroči le 1% spremembo na kotni hitrosti ω . V krmilnem sistemu 10% sprememba na parametru K_M povzroči 10% spremembo na kotni hitrosti ω .

Odpravljanje motenj

V krmilnem sistemu motnja T_L povzroči spremembo v kotni hitrosti

$$\Delta\omega = K_L K_M T_L$$

v regulacijskem sistemu pa spremembo

$$\Delta\omega = \frac{K_L K_M T_L}{1 + K_R K_M}$$

V ustaljenem stanju v regulacijskem sistemu stopničasta bremenska motnja T_L za faktor $1 + K_R K_M$ manj vpliva na spremembo kotne hitrosti kot v krmilnem sistemu. Faktor $\frac{1}{1 + K_R K_M}$ se imenuje regulacijski faktor.

Pohitritev delovanja

Krmilni sistem ima časovno konstanto τ , regulacijski sistem pa $\frac{\tau}{1 + K_R K_M}$. Torej ima regulacijski sistem za faktor $1 + K_R K_M$ krajšo časovno konstanto kot krmilni sistem, torej je lahko ob ustrezni izbiri parametrov bistveno hitrejši.

9.9 Nekateri dodatni pojmi, ki jih srečujemo v regulaciji

Servosistem

Servosistem je povratnozančni regulacijski sistem, v katerem je regulirana veličina pozicija, hitrost ali pospešek mehanskega dela. Zato sta izraza servosistem in pozicijski (ali hitrostni ali pospeškovni) regulacijski sistem identična. Servosistemi se veliko uporabljajo v sodobni industriji, zlasti v robotiki, letalski, vesoljski tehniki itd.

Procesni regulacijski sistem

Procesni regulacijski sistem je sistem, v katerem je regulirana veličina temperatura, tlak, pretok, nivo tekočine ali pH . Tovrstni regulacijski sistemi so zelo pogosti zlasti v kemični industriji, pa tudi v vsakdanjem življenju (regulacija temperature).

Stohastični regulacijski sistem

Stohastični regulacijski sistemi so sistemi, v katere vstopajo naključni (stohastični) signali predvsem kot motilni signali. Deterministični regulatorji so sicer do neke mere uspešni tudi pri odpravljanju stohastičnih motenj, vendar je za optimalno delovanje potrebno načrtati posebne regulatorje, ki upoštevajo statistične parametre stohastičnih signalov.

Adaptivni regulacijski sistem

Dinamične karakteristike večine regulacijskih sistemov niso konstantne. Učinke majhnih sprememb sicer povratna zanka zadovoljivo odpravi. Pri znatnejših spremembah parametrov pa mora imeti sistem sposobnost prilagajanja na te spremembe. Adaptivni regulacijski sistem avtomatsko zazna spremembe v parametrih in v skladu s tem prilagodi regulacijski algoritem. Adaptivni sistemi so torej

časovno spremenljivi sistemi. Predstavljajo eno najsodobnejših raziskovalnih področij avtomatike.

Samoučeči regulacijski sistem

če skušamo avtomatizirati sisteme vodenja, v katerih je povratnozančni regulator človek-operator, pridemo nemalokrat do problemov, ko želimo delovanje operaterja popisati z enačbami. Človek ima namreč lastnost učenja. Z več izkušnjami postane boljši regulator, kar je seveda treba upoštevati. Avtomatski regulacijski sistemi s podobnimi lastnostmi se imenujejo samoučeči regulacijski sistemi (learnig control systems). Tudi samoučeči regulacijski sistemi predstavljajo eno najsodobnejših področij, ki se zelo močno povezuje s področji umetne inteligence, ekspertnih sistemov in nevronske mreže.

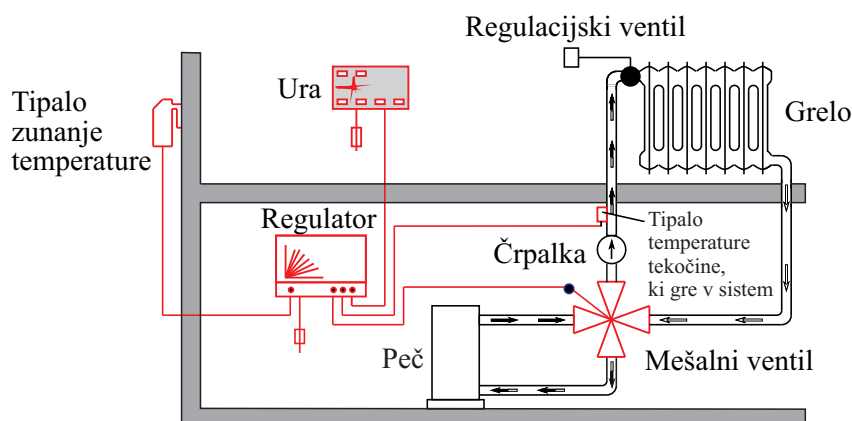
9.10 Primeri sistemov vodenja

Regulacija temperature v prostoru

Za regulacijo temperature v prostoru obstajajo številni načini, ki v glavnem zavisijo od vrste grela, ki ga uporabljamo. V glavnem uporabljamo t.i. stopenjsko (ON-OFF) regulacijo v povezavi z električnimi greli, pa tudi bolj napredno regulacijo pri sistemih, ki imajo določeno akumulacijo toplotne energije. Stopenjska (ON-OFF) regulacija je sicer najcenejša, vendar povzroča nezaželeno nihanje temperature.

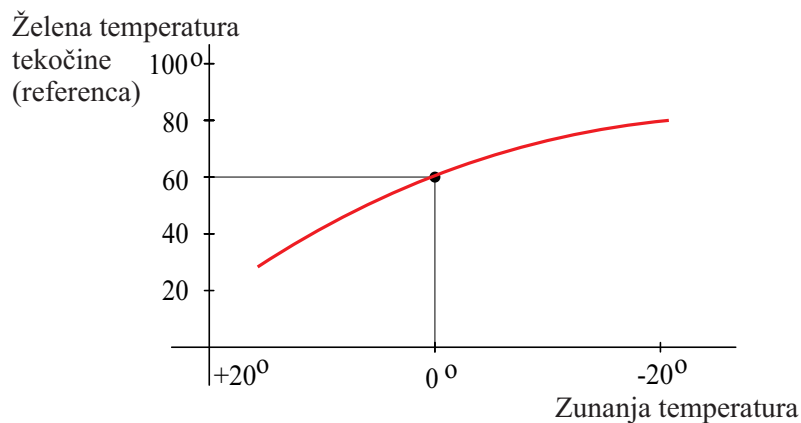
Slika 9.19 prikazuje sistem vodenja centralnega ogrevanja.

Regulacijski del sestavljajo tipalo zunanje temperature, tipalo, ki meri temperaturo tekočine, ki jo pošiljamo v sistem, regulator, ura, ki se nahaja v prostoru, ter mešalni ventil (končni izvršni člen), ki ga odpira in zapira motor (aktuator). Proces sestavljajo peč za centralno ogrevanje, črpalka, grelo v prostoru, ustrezen povezovalni sistem in seveda samo ogrevanje prostora. Osnovna zanka omogoča regulacijo temperature tekočine, ki jo pošiljamo v sistem. Regulator glede na razliko med želeno in dejansko temperaturo tekočine, ki jo meri tipalo, daje motorju ustrezen signal za zasuk ventila. Ventil je mešalnega tipa in omogoča želeno temperaturo tekočine na ta način, da v pravilnem razmerju meša tekočino v peči (ta



Slika 9.19: Regulacija temperature v prostoru

je na konstantni, precej visoki temperaturi in regulirana z ustrezno regulacijsko zanko v kotlu) in povratno tekočino, ki se v sistemu precej ohladi. Značilnost regulacijskega sistema pa je v tem, da referenca za temperaturo tekočine ni fiksno nastavljena na regulatorju, ampak se preračuna preko t.i. družine statičnih karakteristik iz zunanje temperature. Tako pri nižjih zunanjih temperaturah polnimo sistem z bolj vročo tekočino. Tak koncept omogoča velik prihranek, saj je največji vir izgub pri centralnem ogrevanju prav v tem, da polnimo sistem s tekočino, ki je toplejša od potrebne. Eno od možnih statičnih karakteristik prikazuje slika 9.20.



Slika 9.20: Statična karakteristika za izračun želene temperature

Uporabnik neposredno torej ne more nastaviti želene temperature, ker je le-ta odvisna od zunanje temperature, lahko pa vpliva na obliko statične karakteristike tako, da ji spreminja naklon ali jo vzporedno premika in na ta način naredi optimalno prilagoditev za njegov prostor.

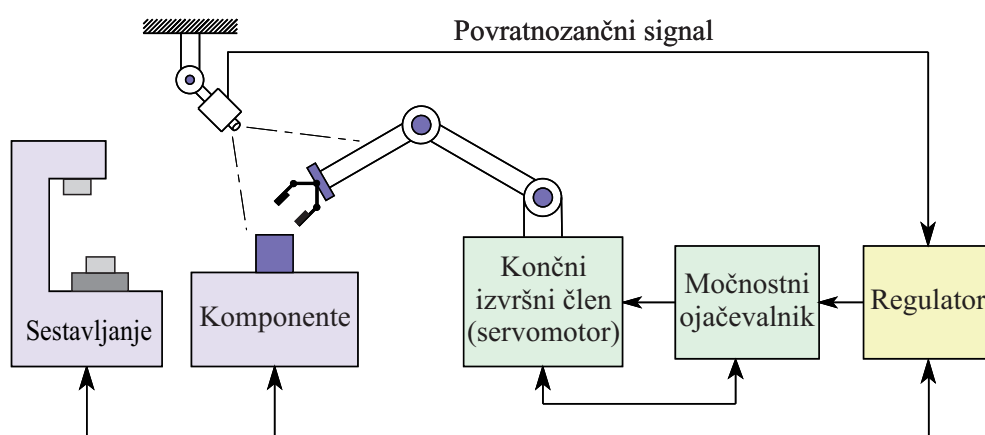
Ura običajno le vklaplja in izklaplja ogrevalni sistem (mešalni ventil) v želenih časovnih intervalih (npr. preklon med dnevnim in nočnim režimom), medtem ko želeno temperaturo v prostoru dosežemo z dodatnim regulacijskim ventilom na grelu.

Robotsko vodenje

Industrijski roboti se često uporabljajo v industriji za povečanje produktivnosti. Robot lahko opravlja enostavna (monotona) ali pa tudi kompleksna dela, ne da bi se pri tem pojavljale napake. Dela lahko v okolju, ki ga človek ne prenese brez škodljivih posledic; to je pri ekstremnih temperaturah, pri ekstremnih tlakih, v onesnaženem okolju, v vodi ali v vakuumu.

Industrijski robot rokuje s predmeti različnih oblik in različnih tež. Zato ima vsaj roko, zapestje in prijemalko. Imeti mora določeno moč in zmožnost določene gibljivosti. Najsodobnejši roboti se znajo sami gibati v omejenem prostoru tovarne.

Industrijski roboti morajo imeti določena tipala. Pri enostavnejših izvedbah so v okolici robota vgrajena mikrostikala. Robot se dotakne predmeta in se potem giblje po komandah mikrostikal. Najsodobnejši roboti pa uporabljajo optične naprave za zaznavanje kot npr. TV kamero. Na osnovi razpoznavanja vzorcev, ki jih daje TV slika, se robot ustrezno giblje. Tak robotski regulacijski sistem prikazuje slika 9.21.

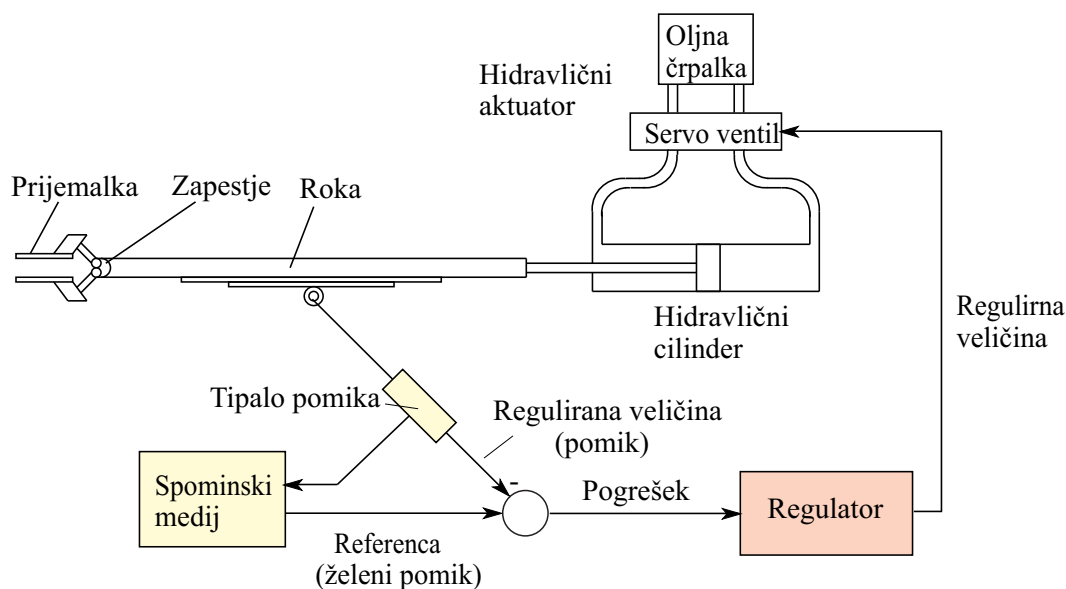


Slika 9.21: Robotski regulacijski sistem

Za ustrezno prepoznavanje vzorcev je seveda nujno potreben mikroračunalnik. V določenih računalniško vodenih aplikacijah robot prepozna prisotnost in orientacijo mehanskega dela s postopkom prepoznavanja vzorcev. Nato robot prime mehanski del ter ga položi na mesto, kjer nato več delov združi v določen predmet.

Regulacija robotske roke

Slika 9.22 prikazuje sistem za regulacijo robotske roke. Primer je poenostavljen in kaže regulacijo pozicije v eni sami smeri oz. premikanje v eni prostostni stopnji. Dejansko ima kompletni mehanizem običajno sedem prostostnih stopenj (roka tri, zapestje tri in prijemalka eno (stisk)).



Slika 9.22: Regulacija robotske roke

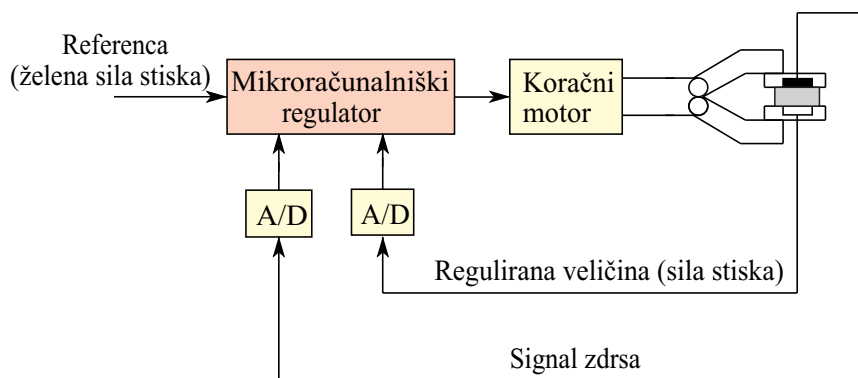
Za pozicioniranje roke in zapestja uporabljamo pozicijski regulacijski sistem. Ker robotska roka običajno potrebuje hitrost in moč, se kot vir moči uporablja hidravlični ali pnevmatski sistem. Za srednje moči so primerni tudi enosmerni motorji, za majhne moči pa koračni motorji.

Robotsko roko premika regulacijski sistem. Le-ta na osnovi referenčnega signala, ki je zapisan v spominskem mediju računalnika, omogoča ustrezno gibanje. Pri sodobnih robotskih sistemih je običajni postopek, da človek operator ročno vodi robota po želenih trajektorijah, računalnik pa snema ustrezne referenčne signale.

Temu pravimo učenje robota. V fazi delovanja je nato robot ob optimalnem sledenju posnetim referenčnim signalom sposoben opravljati želene funkcije.

Regulacija sile stiska v robotovi prijemalki

Slika 9.23 prikazuje shematski prikaz robotskega sistema za regulacijo sile stiska v prijemalki. Če je sila stiska premajhna, robot predmet izpusti, če pa je prevelika,

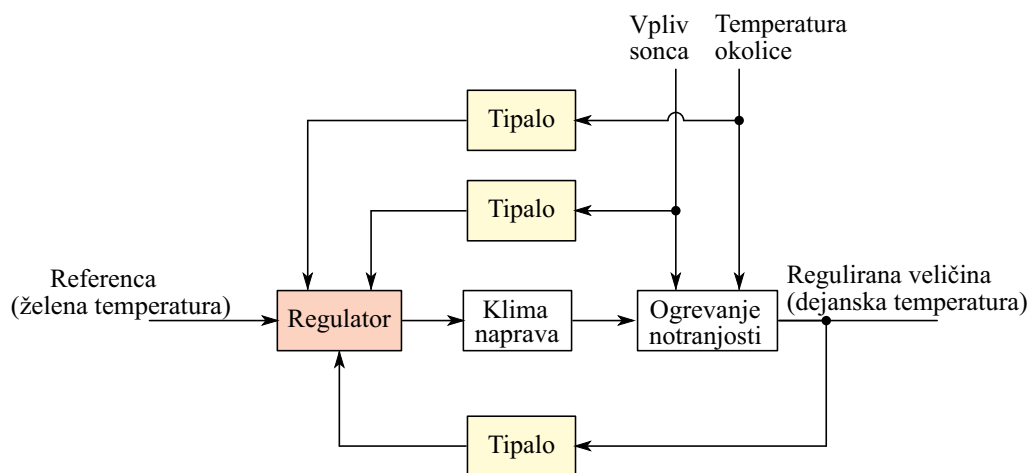


Slika 9.23: Regulacija sile stiska v prijemalki

lahko poškoduje ali uniči predmet. V sistemu, ki ga prikazuje slika 9.23, je sila stiska prednastavljena (regulirana) na neko srednjo vrednost. Prijemalka vzame in dvigne predmet s prednastavljeno silo stiska. V roki pa je vgrajeno tudi tipalo, ki detektira morebitni zdrs. Tako tipalo običajno vsebuje svetlobno diodo in fototranzistor. V primeru zdrsa dobi mikroračunalnik oz. regulator zahtevo, da poveča silo stiska. Na ta način se vzpostavi ustrezna sila, ki prepreči zdrs, a hkrati prepreči tudi poškodbo predmeta.

Regulacija notranje temperature v avtomobilu

Slika 9.24 prikazuje shematski diagram sistema za regulacijo temperature v notranjosti avtomobila. Regulator primerja želeno in merjeno temperaturo, hkrati pa sprejema tudi informacijo o zunanji temperaturi in o sončnem sevanju preko ustreznih tipal. Ta dva signala obravnavamo kot motnji. Zato je ta sistem kombinacija odprtozančnega vodenja (krmiljenja) in zaprtozančne regulacije. Medtem ko zaprtozančna regulacija deluje običajno, t.j. na osnovi pogreška merjene in



Slika 9.24: Regulacija notranje temperature v avtomobilu

regulirane veličine, pa je za krmiljenje značilno, da lahko odpravlja motnje brez bistvenih zakasnitev, saj istočasno, ko le-te delujejo na proces, dobi regulator že informacijo o teh motnjah in ustrezno ukrepa. Tako krmiljenje je tem bolj učinkovito, čim bliže je motnja vходу v proces, saj jo je v tem primeru možno praktično neposredno odšteti. Na osnovi štirih signalov regulator določi ustrezni regulirni signal, ki ga posreduje klimatski napravi.

10.

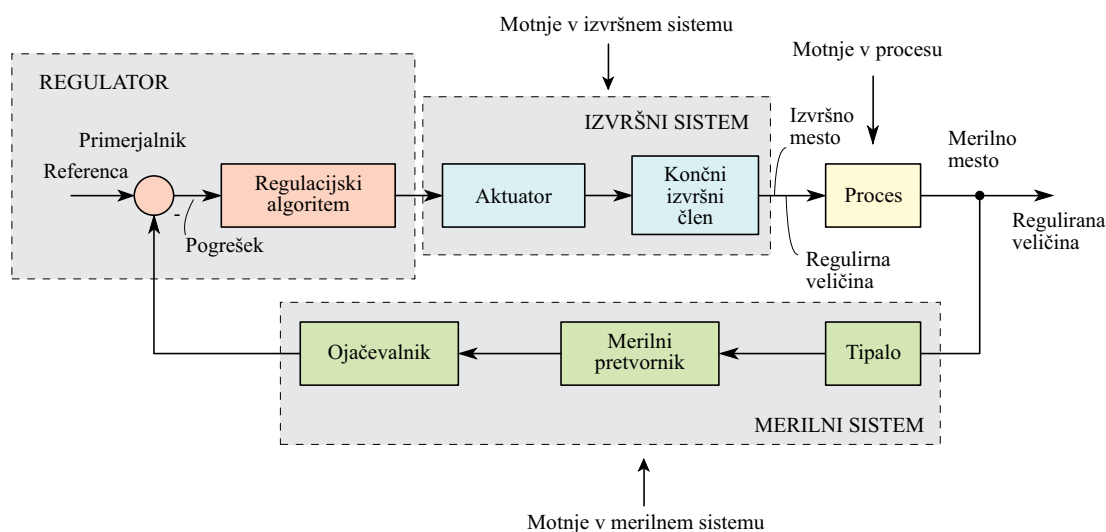
Industrijski regulacijski algoritmi

Analiza in načrtovanje regulacijskih algoritmov predstavljata enega najpomembnejših, gotovo pa zlasti s teoretičnega vidika enega najčudovitejših in najbolj izzivalnih področij vodenja sistemov. Regulacijski sistemi so lahko zelo enostavni ali pa silno zamotani, tako da postopki analize in načrtovanja zahtevajo dobro poznavanje številnih zahtevnih metod ob ustreznem predznanju matematike, fizike in teorije sistemov. Regulacijski algoritmi so v smislu uporabnosti doživeli velik razmah tudi po zaslugi razvoja orodij za modeliranje, simulacijo in računalniško podprto načrtovanje.

Čeprav poteka večina raziskav v zadnjem obdobju na področjih kot so prediktivno in adaptivno vodenje ter na uporabi znanj umetne inteligence (ekspertni sistemi, nevronske mreže, mehka (fuzzy) logika), je namen tega poglavja, da prikaže bolj klasične, a v industriji še vedno daleč najpogostejše uporabljane regulacijske algoritme, njihov vpliv na delovanje regulacijske zanke ter postopke za njihovo načrtovanje (Kuo, 2010, Ogata, 2010, Zupančič, 2010a, Zupančič, 2010b). Pri tem se bomo osredotočili na enozančne *PID* regulatorje.

Zaradi boljšega razumevanja praktičnih problemov, ki nastopajo pri izvedbi regulacije, si oglejmo razširjeni bločni diagram na sliki 10.1.

Tipalo odtipava regulirano veličino na merilnem mestu. Signal, ki ga daje tipalo, moramo pogosto pretvoriti z merilnim pretvornikom in ga nato ojačati z ojačevalnikom (npr. v standardni signal $4 - 20\text{ mA}$). Na poti od merilnega mesta se vnašajo motnje, ki jih imenujemo motnje v merilnem sistemu (ali tudi motnje na regulirano veličino). Pretvorjena regulirana veličina vstopa v regulator in se tam



Slika 10.1: Bločna shema regulacijskega sistema

primerja z želeno oz. referenčno vrednostjo. Iz pogreška med njima regulacijski algoritem določi signal (npr. $4 - 20 \text{ mA}$), ki preko aktuatorja (npr. motor) in končnega izvršnega člena (npr. ventil, loputa) na izvršnem mestu deluje na proces. Na poti od regulatorja do procesa lahko delujejo motnje, ki jih imenujemo motnje v izvršnem sistemu (ali tudi motnje na regulirno veličino).

Vse elemente regulacijskega sistema razen procesa imenujemo regulirni sistem. Na ta način lahko regulacijski sistem prikažemo le z dvema blokoma. Tipalo, merilni pretvornik in ojačevalnik združimo v en blok, ki ga imenujemo merilni sistem (tudi kot del opreme je to lahko ena enota, ki iz regulirane veličine generira npr. signal $4 - 20 \text{ mA}$). Aktuator in končni izvršni člen pa z enim izrazom imenujemo izvršni sistem (tudi kot del opreme je to lahko enota, ki ima npr. vhod $4 - 20 \text{ mA}$, izhod pa zasuk ventila).

Regulator predstavlja inteligentni del regulacijskega sistema, saj na podlagi razlike med želeno in pretvorjeno regulirano veličino določa regulirni signal s pomočjo ustreznega regulacijskega algoritma. Razen tega ima regulator še razne vrste dodatkov, ki omogočajo večjo učinkovitost: alarmiranje, filtriranje, zaščita pred integralnim pobegom, preklon ročno-avtomatsko, direktno in reverzno delovanje, itd.

10.1 Razdelitev industrijskih regulatorjev

Regulatorje v industriji delimo lahko po različnih kriterijih. Glede na energijo, ki jo potrebujejo za delovanje, so lahko:

- regulatorji brez pomožne energije (samodelujoči) ali
- regulatorji s pomožno energijo.

Prvi za svoje delovanje uporabljajo kar energijo medija, ki ga regulirajo. Torej so v eni enoti zajete funkcije merilnega sistema, izvršnega sistema in regulatorja. Za delovanje regulatorjev s pomožno energijo pa potrebujemo dodatno (pomožno) energijo.

Glede na vrsto medija ali uporabljene pomožne energije pa delimo regulatorje na

- mehanske,
- pnevmatske,
- hidravlične in
- električne.

Glavne značilnosti regulatorjev brez pomožne energije so naslednje:

- uporabljajo se za posebne namene,
- izdelujejo se masovno,
- so poceni,
- so zelo zanesljivi,
- izpolnjujejo varnostne zahteve,
- območje signalov je omejeno in neprilagodljivo,
- dinamične lastnosti so zelo omejene (običajno dosežemo le proporcionalno karakteristiko),

- točnost je slabša,
- poseg v zanko ni možen, torej ni modularnosti.

Glavne značilnosti regulatorjev s pomožno energijo pa so naslednje:

- regulatorji so dosti bolj splošno namenski,
- območje signalov je prilagodljivo,
- dinamične lastnosti so zelo raznovrstne,
- omogočajo veliko točnost,
- poseg v zanko je možen, zato omogočajo veliko modularnost,
- so precej dražji.

Daleč največ se zaradi cenenosti in fleksibilnosti uporabljajo električni regulatorji. Le-te delimo na

- analogne in
- digitalne.

Analogni regulatorji so izvedeni z električnimi oz. elektronskimi komponentami, digitalni pa s sodobnimi računalniškimi komponentami.

Glede na vrsto regulirnega signala, ki bistveno vpliva na dinamične lastnosti regulacijske zanke, razdelimo industrijske regulatorje na

- zvezno delujoče regulatorje in
- stopenjsko (nezvezno, diskontinuirno) delujoče regulatorje.

Pri zvezno delujočih regulatorjih lahko regulirna veličina zavzame poljubno vrednost znotraj regulirnega območja. Zato je takšna regulacija bolj točna kot pri stopenjskih regulatorjih, pri katerih lahko regulirna veličina zavzame samo nekatere vrednosti.

Glede na dinamične značilnosti delimo zvezne linearne regulatorje na

- proporcionalne (P),
- integrirne (I),
- proporcionalno - integrirne (PI),
- proporcionalno - diferencirne (PD) in
- proporcionalno - integrirno - diferencirne regulatorje (PID).

Pri stopenjsko delujočih regulatorjih lahko zavzame regulirna veličina dve ali največ tri različne vrednosti, tako da često govorimo o

- dvopoložajnih (ON-OFF) in
- tropoložajnih regulatorjih.

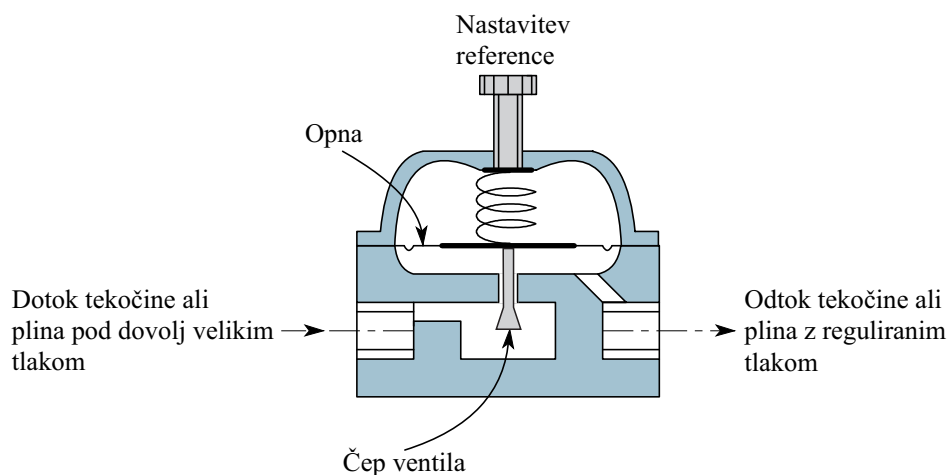
Stopenjski regulatorji se zaradi enostavnosti in cenenosti veliko uporabljajo v industriji pa tudi v raznih hišnih napravah (regulacija temperature s termo-akumulacijsko pečjo, regulacija temperature v likalniku, regulacija nivoja tekočine v izplakovalniku WC, ...). S pomočjo dodatne povratne zveze okoli preklopnega elementa lahko dinamične lastnosti še izboljšamo.

Primer 10.1 Zvezni regulator brez pomožne energije za uravnavanje tlaka tekočine ali plina.

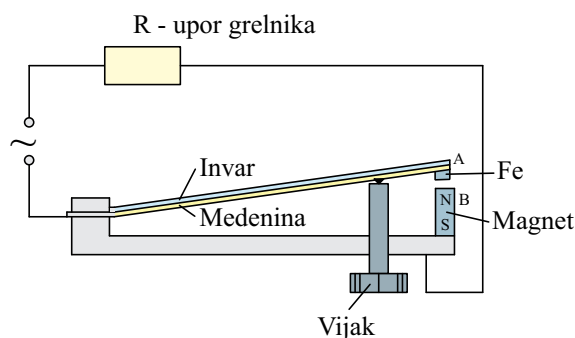
Regulator, ki se uporablja za regulacijo tlaka v plinih in tekočinah, prikazuje slika 10.2. Referenčno veličino nastavimo z vijakom, s katerim določimo silo vzmeti. Tlak merimo s pomočjo opne. Pogrešek je razlika med tlakom zaradi vzmeti in tlakom plina ali tekočine. Če je tlak plina ali tekočine manjši od referenčnega tlaka, se opna ukrivi navzdol. To povzroči povečanje pretoka plina ali tekočine in povečanje tlaka. Ko je tlak vzmeti enak tlaku plina ali tekočine, je ventil v ustaljeni legi, pretok je konstanten in tlak ustrezno reguliran. $\square$

Primer 10.2 Stopenjski regulator brez pomožne energije za uravnavanje temperature.

Slika 10.3 prikazuje tipični temperaturni regulator iz bimetala, ki se uporablja v likalnikih, električnih blazinah, pečicah in podobnih napravah.



Slika 10.2: Zvezni regulator brez pomožne energije za uravnavanje tlaka tekočine ali plina

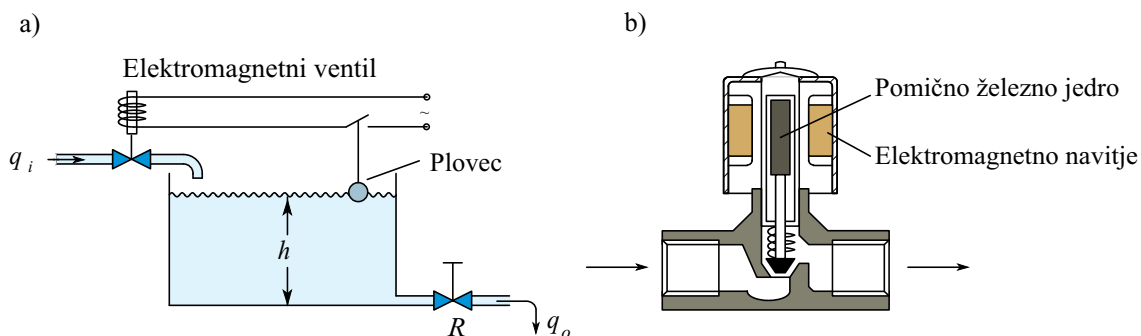


Slika 10.3: Bimetalni temperaturni regulator kot stopenjski regulator

Tipalo in istočasno regulator predstavljata zvarjeni kovini – medenina (velik toplotni razteznostni koeficient) in invar (majhen toplotni razteznostni koeficient). Različni raztezki obeh kovin preneseni preko vzvoda - vijaka, s katerim nastavljamo tudi referenčno vrednost, ustvarjajo stik v tokokrogu. Pri tej izvedbi, ki je primerna za manjše preklopne moči, nastopa iskrenje kontaktov, kar zmanjšuje življensko dobo. Problem omilimo, če kontaktne del (A,B) izvedemo z magnetom in železno ploščico ali če uporabimo vzmet. Slednjo rešitev uporabljamo pri preklonih večje moči. □

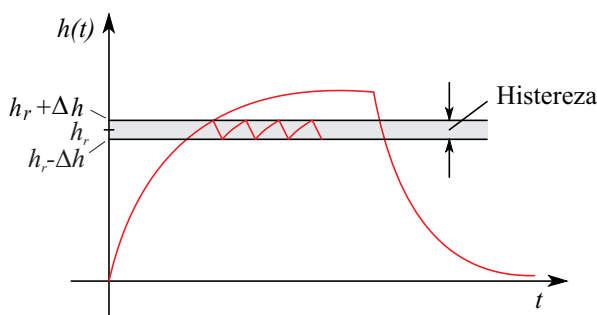
Primer 10.3 Stopenjski regulator s pomožno energijo za uravnavanje nivoja tekočine.

Slika 10.4a pa predstavlja dvopoložajni regulacijski sistem z uporabo pomožne, v tem primeru električne energije. Uporablja se elektromagnetni ON - OFF ventil, ki ga prikazuje slika 10.4b. Regulirana veličina je nivo v posodi, regulirna veličina pa vhodni pretok tekočine. Vhodni pretok je lahko nek pozitiven pretok pri



Slika 10.4: a) regulacijski sistem za regulacijo nivoja
b) elektromagnetni ventil

odprtem ventilu ali pa nič pri zaprtem ventilu. Izhodni pretok predstavlja porabo. Če predpostavimo, da je le-ta npr. 10 lmin^{-1} , potem mora biti pretok pri polnem odprtju ustrezno večji, npr. 20 lmin^{-1} , kar omogoča relativno hitro polnjenje posode. Želeni nivo se torej vzpostavi s periodičnim polnjenjem in praznjenjem. Potek nivoja prikazuje slika 10.5. Nihanje regulirane veličine je v glavnem posledica



Slika 10.5: časovni potek nivoja tekočine

dica preklopne karakteristike, ki vsebuje tudi histerezo. Z zmanjšanjem histereze je sicer možno zmanjšati tudi nihanja regulirane veličine, vendar se pri tem zelo poveča število preklopov, kar povzroča hitro izrabo preklopnih elementov. $\square$

10.2 Zvezno delujoči regulatorji

Zvezno delujoči regulatorji imajo lastnost, da regulirna veličina lahko v določenem območju, ki mu pravimo regulirno območje, zavzame katerokoli vrednost. Obravnavali bomo regulatorje, za katere velja linearna zveza med pogreškom in regulirano veličino. Večina industrijskih regulatorjev glede na vgrajene algoritme, ki določajo zvezo med pogreškom in regulirano veličino, pripada enemu naslednjih tipov ali kombinaciji tipov regulatorjev:

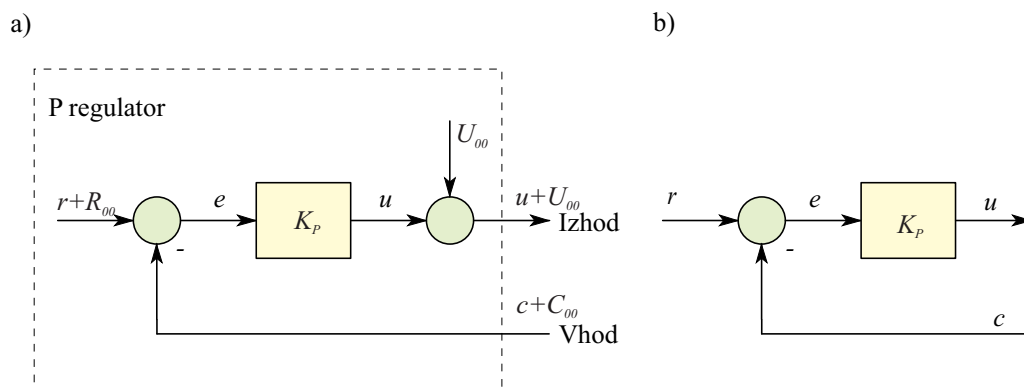
- proporcionalno delujoči regulatorji (P regulatorji),
- integrirno delujoči regulatorji (I regulatorji),
- diferencirno delujoči dodatki regulatorjev (D regulatorji).

V splošnem je diferencirni regulator kot samostojni regulator neuspešen, saj upošteva le spremembe pogreška. V kombinaciji s proporcionalnim ali integrirnim regulatorjem pa je lahko zelo učinkovit.

10.2.1 Proporcionalni regulatorji

Proporcionalni regulacijski algoritem je temeljni algoritem v industrijskih regulatorjih in uspešno rešuje velik del regulacijskih problemov. Njegova osnovna značilnost je, da je regulirna veličina premosorazmerna s pogreškom. Slika 10.6a prikazuje bločno shemo P regulatorja. Vhod v regulator je regulirana veličina. Le-ta se odšteje od referenčne veličine. Tako nastane signal pogreška, ki se pomnoži s proporcionalno konstanto K_P (ojačenje P regulatorja), kar daje regulirno veličino. Vse veličine so izražene v absolutnih iznosih. Enosmerne vrednosti imajo naslednji pomen: enosmerna komponenta regulirne veličine U_{00} omogoči tako regulirno veličino (npr. moč grelnika), da postane regulirana veličina enaka delovni (običajno želeni oz. referenčni) veličini $C_{00} = R_{00}$ (npr. temperatura v prostoru C_{00} postane v ustaljenem stanju enaka $R_{00} = 20^\circ C$). Vendar pa pri analizi in sintezi regulacijskih sistemov obravnavamo le odstopanja od delovne točke. Zato pri tem uporabljamo bločno shemo, kot jo prikazuje slika 10.6b. Tako velja za P regulator enačba

$$u(t) = K_P e(t) \quad (10.1)$$



Slika 10.6: Industrijski P regulator: a) absolutne veličine
b) odstopanja od delovne točke

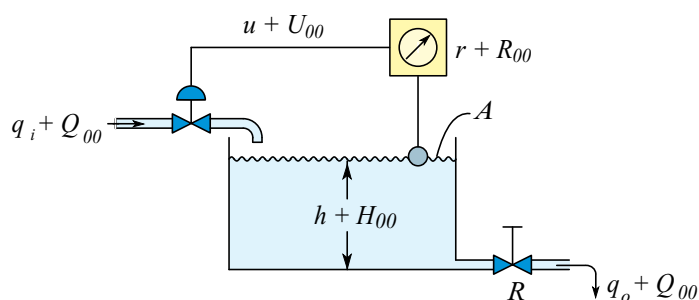
oz. prenosna funkcija

$$G_R(s) = \frac{U(s)}{E(s)} = K_P \quad (10.2)$$

Industrijski proporcionalni regulator vsebuje vsaj tri gumbе: gumb za nastavljanje želene vrednosti R_{00} , gumb za nastavljanje ojačenja K_P in gumb za nastavljanje enosmerne oz. delovne vrednosti regulirne veličine U_{00} .

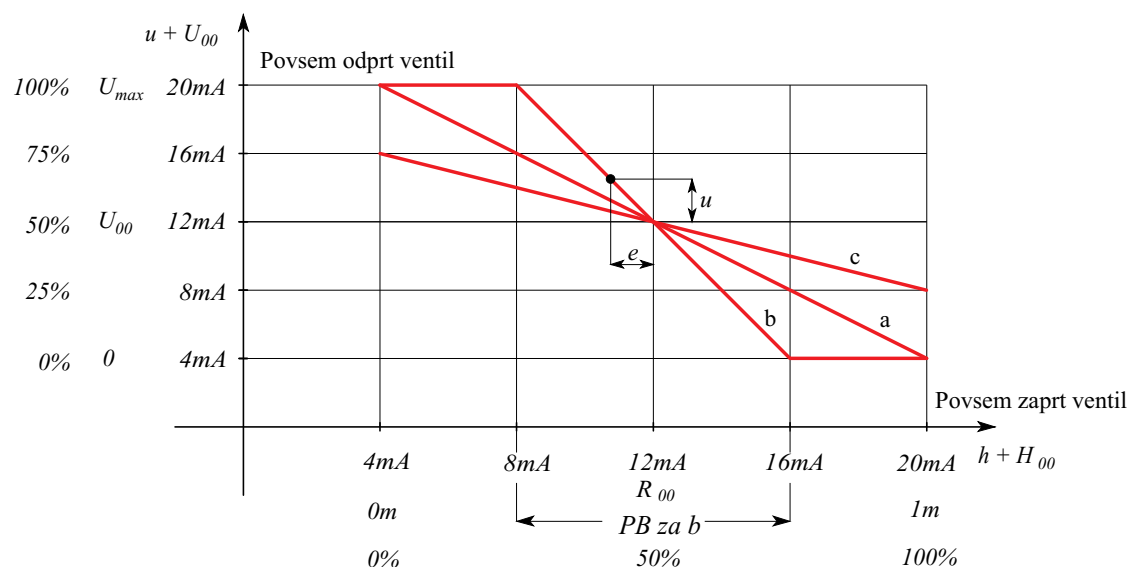
Vendar pa običajno parameter industrijskega regulatorja ni neposredno ojačenje ampak t.i. proporcionalno območje regulatorja. Za razumevanje proporcionalnega območja je potrebno poznati statično karakteristiko proporcionalnega regulatorja, to je odvisnost med pogreškom (ali med regulirano veličino) in regulirno veličino v ustaljenem stanju (to je pomembno le, če ima P regulator še kakšen dinamični dodatek, sicer je vseeno, kdaj določamo kvocient).

Za primer si oglejmo hidravlični sistem, ki ga prikazuje slika 10.7. Sistem omogoča regulacijo nivoja v nivojski posodi. Pri tem je A površina posode, R pa upornost iztočne poti. Merilni signal nivoja (regulirano veličino) daje merilnik na območju $4 - 20mA$ (npr. $4mA \dots$ posoda prazna $\dots 0m$, $20mA \dots$ posoda polna $\dots 1m$). Prav tako je regulirni signal, ki ga regulator posreduje procesu (tok ali kar odprtje ventila), na območju $4 - 20mA$ ($4mA \dots$ ventil povsem zaprt $\dots u + U_{00} = 0$, $20mA \dots$ ventil povsem odprt $\dots u + U_{00} = U_{max}$). V praksi zelo radi veličine izražamo v procentih. Tako regulirana veličina $4mA$ oz. $0m$ pomeni 0% , $20mA$ oz. $1m$ pa 100% . Pri regulirni veličini $4mA$ oz. popolna zaprtost ventila predstavlja 0% in $20mA$ oz. popolna odprtost ventila 100% .



Slika 10.7: Hidravlični regulacijski sistem

Statično karakteristiko regulatorja prikazuje slika 10.8. Na abscisni osi nanašamo



Slika 10.8: Statične karakteristike P regulatorja: a) prop. območje 100%
 b) prop. območje 50%
 c) prop. območje 200%

regulirano veličino (nivo v mA , m ali %), na ordinato pa regulirno veličino (v mA , v enotah odprtja ventila ali v %). Statično karakteristiko izmerimo v odprti zanki tako, da spreminjamo regulirano veličino (nivo) od 0 do 100% in merimo regulirno veličino oz. signal, ki ga daje regulator. Strmina dobljene karakteristike je sicer odvisna le od ojačenja regulatorja K_P , vendar je celotna oblika odvisna tudi od nastavitvev R_{00} in U_{00} na regulatorju (R_{00} je želena vrednost regulirane veličine, U_{00} pa regulirna veličina, ki zagotovi, da postane regulirana veličina enaka željeni). V načinu delovanja ročno izberimo $R_{00} = 50\%$ in $U_{00} = 50\%$, nato pa

preklopimo v način avtomatsko. Torej če bo $h + H_{00} = 50\%$, bo pogrešek $e = 0$ in regulirna veličina bo $u + U_{00} = 50\%$ (polovično odprtje ventila). To je le primer idealno dimenzioniranega regulacijskega sistema, saj 50% območja regulirane veličine povzroči negativni pogrešek, 50% pa pozitiven pogrešek. Regulirna veličina pa ima 50% območja za zmanjševanje in 50% območja za povečevanje. Če je regulirana veličina manjša od referenčne veličine, je pogrešek pozitiven in regulator bolj odpre ventil. Pri dovolj velikem pogrešku je ventil povsem odprt, regulirna vrednost se zato ne more več povečati. Če pa je regulirana veličina večja od referenčne (nivo je nad zelenim nivojem), regulator zaradi negativnega pogreška bolj zapre ventil. Tri različne krivulje v sliki 10.8 predstavljajo tri različne občutljivosti regulatorja na signal pogreška oz. tri različna ojačenja.

Proporcionalno območje regulatorja (PB - proportional band, včasih ima oznako x_P) je definirano kot območje regulirane veličine (ali pogreška), ki izkoristi 100% regulirno veličino. Lahko ga podajamo v enotah regulirane veličine ali v % celotnega merilnega območja regulirane veličine. Tako pri statični karakteristiki a velja $PB = 1m$ ali $PB = 16mA$ ali $PB = 100\%$. Pri karakteristiki b je $PB = 0.5m$ ali $PB = 8mA$ ali $PB = 50\%$, torej 50% območje regulirane veličine ($0.5m$) izkoristi 100% regulirno veličino. Pri karakteristiki c pa je proporcionalno območje $PB = 2m$ ali $PB = 32mA$ ali $PB = 200\%$. Torej šele 200% območje regulirane veličine ($2m$) izkoristi 100% regulirno veličino. Ker pa je največje možno območje regulirane veličine enako 100%, je zato območje regulirne veličine le 25% do 75% (ali $\frac{U_{max}}{4}$ do $\frac{3U_{max}}{4}$ ali $8mA$ do $16mA$).

Proporcionalno območje PB je edini parameter proporcionalnega regulacijskega algoritma. Pri analizi in sintezi regulacijskih sistemov pa raje računamo z ojačenjem regulatorja K_P . Le-to je odvisno od strmine statične karakteristike oz. od občutljivosti regulatorja

$$K_P = \frac{u}{e} = \frac{U_{max}}{PB} \quad (10.3)$$

Ponavadi izrazimo maksimalno vrednost regulirne veličine in proporcionalno območje v %. Zato velja

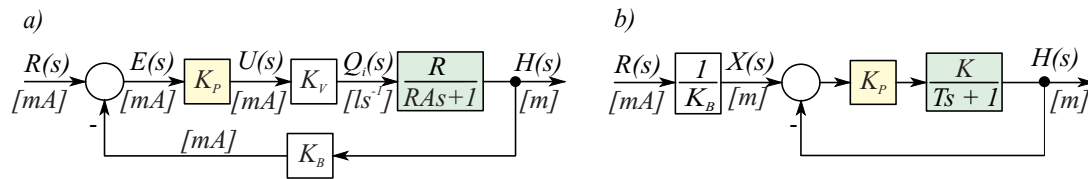
$$K_P = \frac{100\%}{PB[\%]} \quad (10.4)$$

V tem primeru lahko zapišemo proporcionalni algoritem v obliki

$$u(t) = \frac{100\%}{PB[\%]} e(t) \quad (10.5)$$

Primer 10.4 Proporcionalna regulacija nivoja hidravličnega proporcionalnega sistema 1. reda.

Za hidravlični sistem na sliki 10.7 velja bločni diagram, ki ga v originalni in poenostavljeni obliki prikazuje slika 10.9.



Slika 10.9: Bločni diagram hidravličnega regulacijskega sistema:

- a) originalni diagram
- b) poenostavljeni diagram

Ustrezno prenosno funkcijo bi dobili s postopkom modeliranja, ki smo ga v 4. poglavju prikazali za sistem z dvema nivojskima posodama. K_P je ojačenje proporcionalnega regulatorja, K_V je konstanta ventila (zveza med regulirnim signalom v $[mA]$ in vhodnim pretokom), K_B pa je konstanta merilnega sistema (zveza med regulirano veličino v $[m]$ in $[mA]$). Če nas zanima zveza med želenim nivojem $x(t)$ na sliki 10.9b (v $[m]$) in regulirano veličino $h(t)$, velja

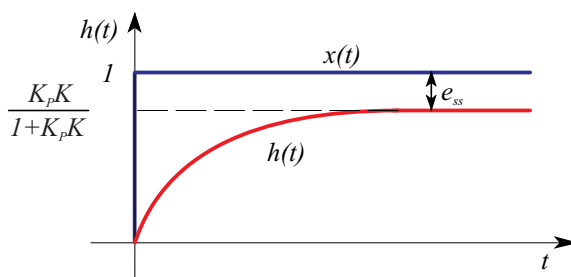
$$\frac{H(s)}{X(s)} = \frac{K_P K}{Ts + 1 + K_P K} \quad K = K_V R K_B \quad T = RA \quad (10.6)$$

Pri stopničastem poteku spremenljivke $x(t)$ ($X(s) = \frac{1}{s}$) je

$$h(t) = \frac{K_P K}{1 + K_P K} \left(1 - e^{-\frac{t}{T_1}} \right) \quad T_1 = \frac{T}{1 + K_P K} \quad (10.7)$$

Odziv prikazuje slika 10.10.

Vidimo, da je regulacijski sistem za faktor $\frac{1}{1+K_P K}$ zmanjšal časovno konstanto. Pogrešek v ustaljenem stanju pa je



Slika 10.10: Potek nivoja pri spremenjeni želeni vrednosti

$$e_{ss} = x_{ss} - h_{ss} = 1 - \frac{K_P K}{1 + K_P K} = \frac{1}{1 + K_P K} \quad (10.8)$$

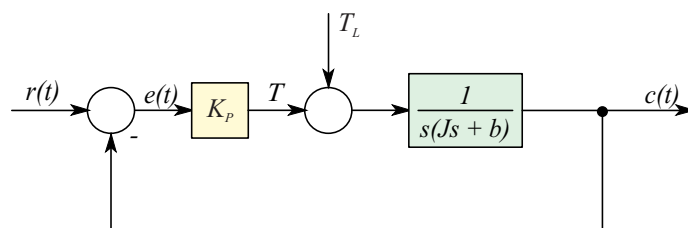
Enak rezultat dobimo tudi ob upoštevanju konstante pogreška. Ker je sistem proporcionalen, je konstanta pozicijskega pogreška $K'_P = K_P K$. Le-ta vpliva na velikost pogreška v ustaljenem stanju pri stopničasti referenci z enačbo

$$e_{ss} = \frac{1}{1 + K'_P} = \frac{1}{1 + K_P K} \quad (10.9)$$

Pogrešek v ustaljenem stanju bi dobili tudi pri analizi regulacijskega sistema v primeru motnje. Če je v regulacijskem sistemu potreben v ustaljenem stanju regulirni signal, ki je različen od nič (velja za proporcionalne procese), potem P regulator potrebuje nek pogrešek, da lahko generira tak signal. Pri tem imamo v mislih le odstopanje od delovnih vrednosti. S pomočjo U_{00} je namreč možno doseči, da je v eni in samo v eni delovni točki tudi pri P regulatorju pogrešek enak nič. $\square$

Primer 10.5 Proporcionalna regulacija zasuka rotacijskega mehanskega sistema ($I1$ sistem).

V tem primeru bomo proučevali regulacijo zasuka sistema 1. vrste pri stopničasti spremembi reference in motnje. Rotacijski mehanski sistem ima vztrajnostni moment J in viskozno dušenje b . Reguliramo ga s proporcionalnim regulatorjem. V tem primeru privzamemo, da je regulator idealni motor, ki iz signala pogreška e generira ustrezen moment T . Bločni diagram prikazuje slika 10.11. Pri tem je $r(t)$ želeni zasuk, $c(t)$ dejanski zasuk, $e(t)$ signal pogreška, T moment idealnega motorja in T_L motilni signal (npr. zavora pri rotiranju).



Slika 10.11: Regulacija zasuka mehanskega rotacijskega sistema

Pogrešek regulacijskega sistema pri podani referenčni veličini lahko določimo s pomočjo konstant pogreškov. Pri stopničasti referenci določa pogrešek konstanta pozicijskega pogreška. Ker je le-ta za sistem 1. vrste enaka ∞ , je pogrešek v ustaljenem stanju

$$e_{ss} = 0 \quad (10.10)$$

Torej proporcionalni regulator v primeru integrirnih procesov nima pogreška v ustaljenem stanju, saj je ustaljena vrednost regulirne veličine enaka 0. Ta zaključek pa ne velja za pogrešek, ki nastopi v primeru stopničaste motnje $T_L(s) = \frac{T_0}{s}$. V tem primeru si ne moremo neposredno pomagati s konstantami pogreškov, ampak je potrebno izračunati prenosno funkcijo

$$\frac{C(s)}{T_L(s)} = \frac{1}{Js^2 + bs + K_P} \quad (10.11)$$

Ker je $E(s) = -C(s)$ ($R = 0$), je pogrešek v ustaljenem stanju pri stopničasti motnji $T_L(s) = \frac{T_0}{s}$

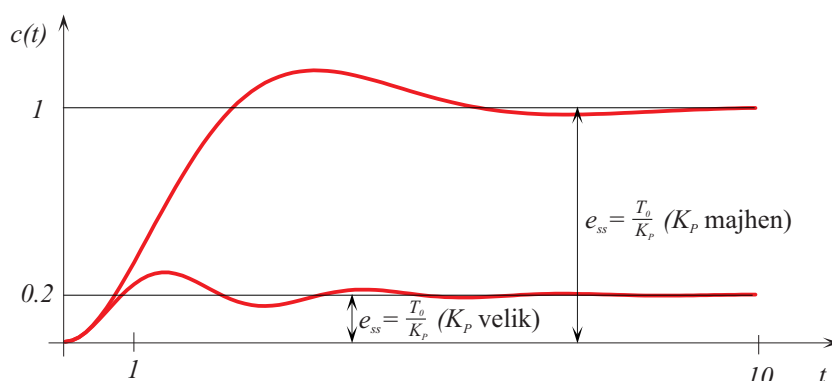
$$e_{ss} = \lim_{s \rightarrow 0} sE(s) = \lim_{s \rightarrow 0} \frac{-s}{Js^2 + bs + K_P} \frac{T_0}{s} = -\frac{T_0}{K_P} \quad (10.12)$$

V ustaljenem stanju regulator proizvaja moment $-T_0$, ki kompenzira konstantni motilni moment T_0 . Ostane pa pogrešek zaradi predhodne akumulacije. Regulirana veličina v ustaljenem stanju pa je podana z izrazom

$$c_{ss} = -e_{ss} = \frac{T_0}{K_P} \quad (10.13)$$

Pogrešek v ustaljenem stanju lahko zmanjšamo le s povečanjem ojačenja P regulatorja. Vendar večji K_P povzroči bolj nihajoč odziv. Slika 10.12 prikazuje

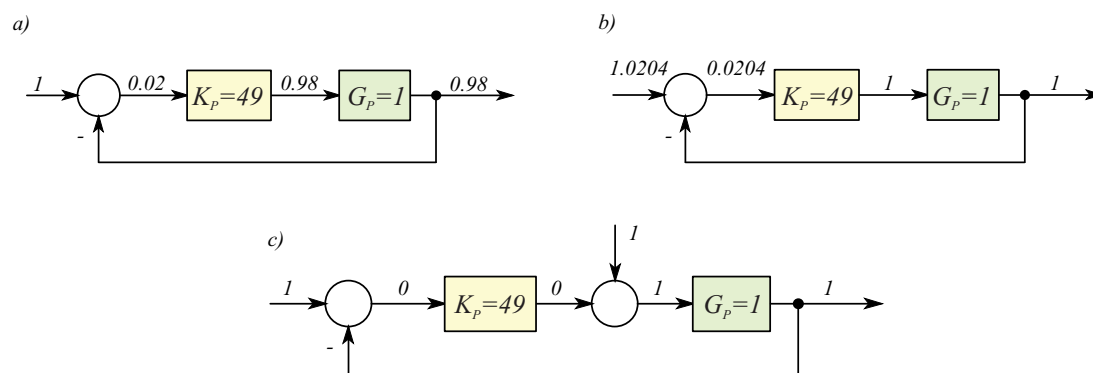
tipična odziva na motnjo za majhno ($K_P = 1$) in veliko vrednost ($K_P = 5$) ojačenja ($T_0 = 1$, $J = 1$, $b = 1$).



Slika 10.12: Tipična odziva pri stopničastem momentu

Vendar s še tako velikim ojačenjem ne moremo narediti sistema nestabilnega. Karakteristična enačba je namreč drugega reda in zagotavlja stabilen sistem, če le so vsi koeficienti karakteristične enačbe pozitivni. $\square$

Pogrešek v ustaljenem stanju je osnovna slabost proporcionalnih regulacijskih sistemov. Problematiko pojasnjuje slika 10.13a. Predpostavimo konstantno spremembo reference, proporcionalni proces pa modeliramo kar z njegovim ojačenjem, saj analiziramo le razmere v ustaljenem stanju.



Slika 10.13: Problem pogreška v ustaljenem stanju pri P regulatorju:

- a) pogrešek med želeno in regulirano vrednostjo
- b) namerno povišanje referenčne vrednosti
- c) prenavstavitve regulirne veličine

Slika 10.13a jasno prikazuje, da je za doseg končne vrednosti regulirane veličine (različne od nič) potreben končni pogrešek. Včasih lahko rešimo problem tako, da namerno nastavimo višjo referenco (slika 10.13b). Na ta način regulirana veličina postane enaka željeni. Lahko pa si pomagamo tudi tako, da spremenimo ročno prenavstavitvev regulirne veličine in ji damo tako vrednost, da bo povzročila želeno regulirano veličino (slika 10.13c). Vendar je tak način možno uporabiti samo, če imamo opravka z konstantnimi referenčnimi ali motilnimi signali, nikakor pa ne v primeru spremenljivih in nepredvidljivih signalov. Tako ročno prenavstavitvev regulirnega signala pa avtomatsko opravi I del regulatorja, zato so včasih integrirno regulacijo imenovali tudi prenavstavitveno regulacijo (reset control).

10.2.2 Integrirni regulatorji

Integrirne regulatorje uporabljamo predvsem tam, kjer ne želimo pogreška v ustaljenem stanju. Algoritem podaja enačba

$$u(t) = K_I \int e(t) dt \quad (10.14)$$

oz.

$$\frac{du(t)}{dt} = K_I e(t) \quad (10.15)$$

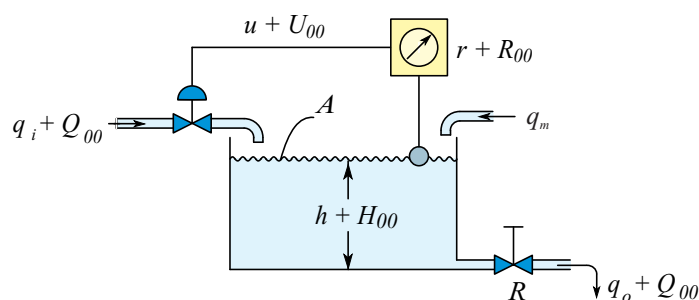
pri čemer je K_I ojačenje integrirnega regulatorja. Torej je hitrost regulirne veličine proporcionalna pogrešku. Prenosna funkcija pa je

$$G_R(s) = \frac{U(s)}{E(s)} = \frac{K_I}{s} \quad (10.16)$$

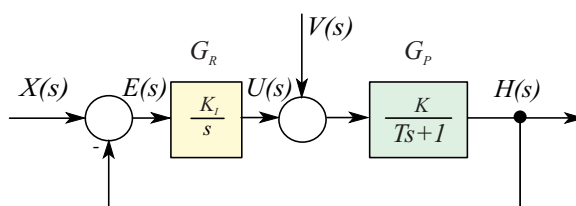
Bistvena lastnost integrirnega regulatorja je v tem, da lahko proizvaja regulirno veličino tudi, če je pogrešek enak nič. To pa zato, ker so pretekle vrednosti pogreška “napolnile” integrator. Pri konstantnih referenčnih ali motilnih signalih je pogrešek v ustaljenem stanju enak nič, saj bi v nasprotnem primeru povzročil linearno naraščajoči regulirni signal. Ker zasuk motorja linearno narašča pri stopničastem vhodnem signalu (približno $I0$ člen pri majhnem vztrajnostnem momentu), predstavlja motor eno od možnosti praktične izvedbe I regulatorja (regulator + aktuator).

Primer 10.6 Integrirna regulacija nivoja hidravličnega proporcionalnega sistema 1.reda

V primeru 10.4 smo obravnavali proporcionalno regulacijo hidravličnega sistema. Če P regulator $G_R = K_P$ zamenjamo z I regulatorjem $G_R = \frac{K_I}{s}$, dobimo za hidravlični sistem, ki ga prikazuje slika 10.14 ob upoštevanju slike 10.9b bločni diagram, ki ga prikazuje slika 10.15.



Slika 10.14: Hidravlični regulacijski sistem



Slika 10.15: Bločni diagram hidravličnega regulacijskega sistema

Pokazali bomo, da je v tem primeru pri stopničasti referenci $X(s) = \frac{1}{s}$ pogrešek v ustaljenem stanju enak nič, kar ni veljalo pri P regulatorju. Velja enačba

$$\frac{E(s)}{X(s)} = \frac{1}{1 + G_R G_P} = \frac{T s^2 + s}{T s^2 + s + K_I K} \quad (10.17)$$

in zato

$$e_{ss} = \lim_{s \rightarrow 0} s E(s) = \lim_{s \rightarrow 0} \frac{s(T s^2 + s)}{T s^2 + s + K_I K} \frac{1}{s} = 0 \quad (10.18)$$

Torej je I regulator izboljšal regulacijo glede na P regulator v tem smislu, da je odpravil pogrešek v ustaljenem stanju.

V primeru stopničaste motnje na vhodu procesa $V(s) = \frac{1}{s}$ (npr. motnja na signalu, ki odpira oz. zapira ventil ali motnja v obliki motilnega dotoka v posodo q_m) pa je prenosna funkcija

$$\frac{E(s)}{V(s)} = \frac{-G_P}{1 + G_R G_P} = -\frac{K s}{T s^2 + s + K K_I} \quad (10.19)$$

Torej je pogrešek v ustaljenem stanju

$$e_{ss} = \lim_{s \rightarrow 0} s E(s) = \lim_{s \rightarrow 0} -\frac{K s^2}{T s^2 + s + K K_I} \frac{1}{s} = 0 \quad (10.20)$$

I regulator odpravlja v ustaljenem stanju vplive vseh vrst konstantnih motenj.

□

10.2.3 Proporcionalno – integrirni regulatorji

Proporcionalno – integrirni (PI) regulator lahko ob ustrezni nastavitvi parametrov združi dobre lastnosti P in I regulatorja. Pri tem P regulator zelo hitro (trenutno) učinkuje na pogrešek, vendar ne odpravlja pogreška v ustaljenem stanju, medtem ko I regulator zaradi končne hitrosti učinkuje relativno počasi, a odpravlja pogrešek v ustaljenem stanju. Z I delom regulatorja povišamo vrsto odprtozančne prenosne funkcije in tako ugodno delujemo na ustaljeni pogrešek (glej podpoglavje 9.6 in tabelo 9.1. PI regulacijski algoritem podaja enačba

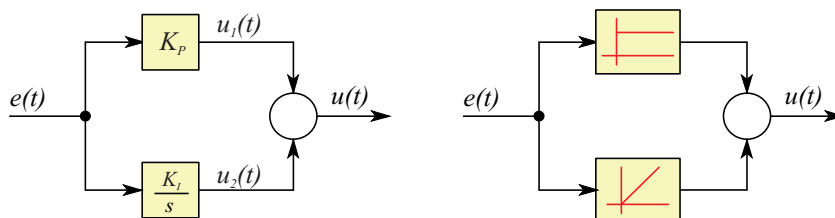
$$u(t) = K_P e(t) + K_I \int e(t) dt = K_P \left(e(t) + \frac{1}{T_I} \int e(t) dt \right) \quad (10.21)$$

oz. prenosna funkcija

$$G_R(s) = K_P \left(1 + \frac{1}{T_I s} \right) \quad (10.22)$$

K_P je ojačenje regulatorja, T_I pa je integrirni čas. PI regulator realiziramo s

paralelno kombinacijo P in I dela. Prikažemo ga z enim od bločnih diagramov na sliki 10.16.



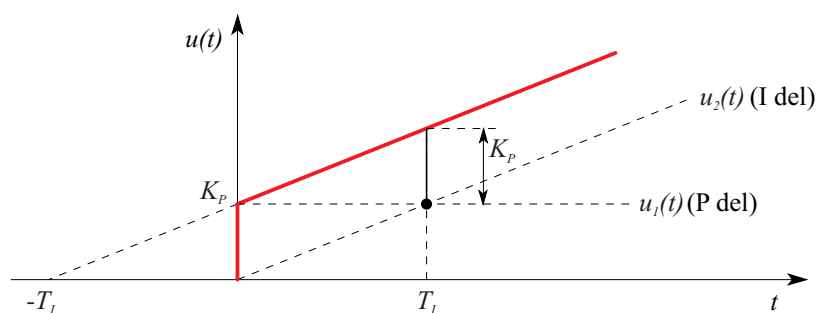
Slika 10.16: Možna bločna diagrama PI regulatorja

V bloka vpišemo ustrezno prenosno funkcijo ali pa vrišemo ikono, ki predstavlja odziv dela regulatorja na stopničasti signal pogreška.

Vpliv parametrov PI regulatorja običajno predočimo s pomočjo odziva regulatorja na stopničasti signal pogreška. Če je $e(t) = 1$, velja

$$u(t) = u_1(t) + u_2(t) = K_P(e(t) + \frac{1}{T_I} \int e(t)dt) = K_P(1 + \frac{1}{T_I}t) \quad (10.23)$$

Ustrezen odziv prikazuje slika 10.17.

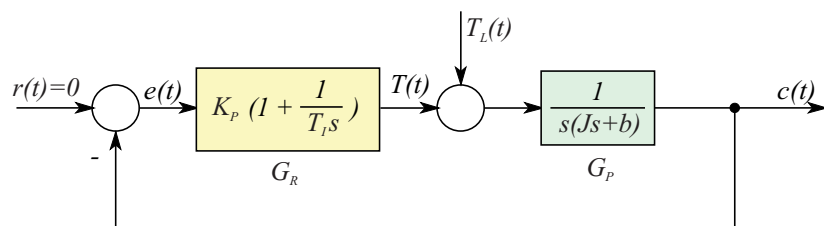


Slika 10.17: Odziv PI regulatorja na enotino stopnico

V času integrirnega časa T_I regulirna veličina podvoji začetno vrednost, ki je določena z ojačenjem K_P . Zato se uporablja za T_I tudi izraz prenestavitveni čas (reset time) ali njegova obratna vrednost - prenestavitvena frekvenca (reset rate). Prenastavitvena frekvenca pove, s koliko je treba pomnožiti del regulirne veličine, ki pripada P delu regulatorja, da dobimo vrednost regulirne veličine po 1 minuti.

Primer 10.7 Proporcionalno – integrirna regulacija zasuka rotacijskega integrirnega mehanskega sistema.

V primeru 10.5. smo si ogledali proporcionalno regulacijo integrirnega rotacijskega mehanskega sistema, ki je omogočila ničelni pogrešek pri referenčnem signalu, kar pa ni veljalo za primer motnje na vходу v proces. Zato je v tem primeru potrebno uporabiti *PI* regulator. Ustreden bločni diagram prikazuje slika 10.18.



Slika 10.18: *PI* regulacija zasuka rotacijskega mehanskega sistema

Velja ($R(s) = 0$)

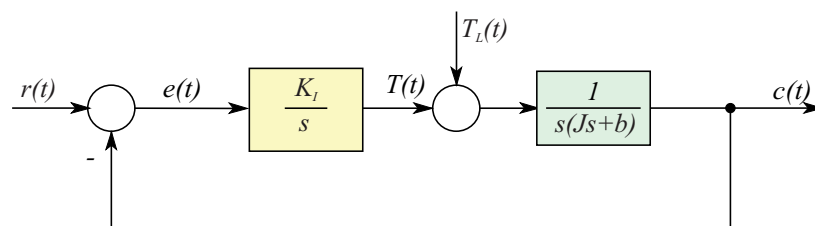
$$\frac{E(s)}{T_L(s)} = \frac{-G_P}{1 + G_R G_P} = \frac{-s}{Js^3 + bs^2 + K_P s + \frac{K_P}{T_I}} \quad (10.24)$$

Če je sistem stabilen (koreni polinoma $Js^3 + bs^2 + K_P s + \frac{K_P}{T_I}$ imajo negativne realne dele), potem izračunamo pogrešek v ustaljenem stanju s pomočjo teorema končne vrednosti. Ker je T_L stopničasta momentna motnja z velikostjo T_0 , velja

$$e_{ss} = \lim_{s \rightarrow 0} sE(s) = \lim_{s \rightarrow 0} \frac{-s^2}{Js^3 + bs^2 + K_P s + \frac{K_P}{T_I}} \frac{T_0}{s} = 0 \quad (10.25)$$

Torej smo s *PI* regulatorjem zagotovili ničelni pogrešek v ustaljenem stanju tudi pri stopničasti motnji na vходу. Zaradi I dela regulatorja pa smo dobili karakteristični polinom 3. reda. V tem primeru moramo biti zelo pazljivi pri izbiri parametrov regulatorja K_P in T_I . Velik K_P in majhen T_I namreč lahko povzročita korene karakterističnega polinoma v desnem delu ravnine s .

Omeniti je še potrebno, da v tem primeru ne moremo uporabiti čistega integrirnega regulatorja kot v primeru hidravličnega sistema (primer 10.6). Tak regulacijski sistem prikazuje slika 10.19. Karakteristična enačba regulacijskega sistema na

Slika 10.19: I regulacija zasuka rotacijskega mehanskega sistema

sliki 10.19 je namreč

$$Js^3 + bs^2 + K_I = 0 \quad (10.26)$$

Enačba (10.26) ima korene s pozitivnim realnim delom, kar je razvidno iz naslednje Routh-ove sheme:

$$\begin{array}{ccc} s^3 & J & 0 \\ s^2 & b & K_I \\ s^1 & -\frac{JK_I}{b} & \\ s^0 & K_I & \end{array}$$

Regulacijski sistem je torej nestabilen in praktično neuporaben. □

10.2.4 Regulatorji z diferencirnim dodatkom

Če proporcionalnemu ali proporcionalno – integrirnemu regulatorju dodamo člen, katerega velikost je proporcionalna odvodu pogreška, dobimo proporcionalno – diferencirni (PD) ali proporcionalno – integrirno – diferencirni (PID) regulator. PD regulacijski algoritem podaja enačba

$$u(t) = K_P e(t) + K_D \frac{de(t)}{dt} = K_P \left(e(t) + T_D \frac{de(t)}{dt} \right) \quad (10.27)$$

oz. prenosna funkcija

$$G_R(s) = K_P(1 + T_D s) \quad (10.28)$$

K_P je ojačenje regulatorja, T_D pa diferencirni čas. PID regulacijski algoritem pa podaja enačba

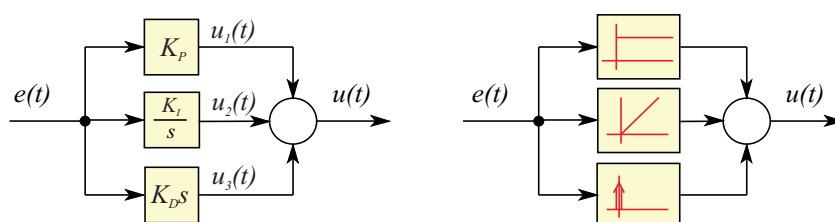
$$u(t) = K_P e(t) + K_I \int e(t) dt + K_D \frac{de(t)}{dt} = K_P \left(e(t) + \frac{1}{T_I} \int e(t) dt + T_D \frac{de(t)}{dt} \right)$$

$$T_I = \frac{K_P}{K_I} \quad T_D = \frac{K_D}{K_P} \quad (10.29)$$

oz. prenosna funkcija

$$G_R(s) = K_P \left(1 + \frac{1}{T_I s} + T_D s \right) \quad (10.30)$$

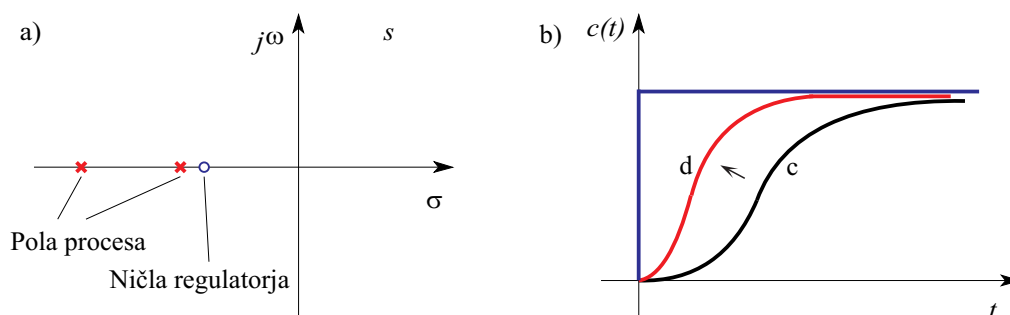
PID regulator realiziramo s paralelno kombinacijo P , I in D člena. Prikažemo ga z enim od bločnih diagramov na sliki 10.20. V bloke vpišemo ustrezne prenosne funkcije ali pa ikone, ki prikazujejo odziv posameznih delov regulatorja na enotino stopnico.



Slika 10.20: Možna bločna diagrama PID regulatorja

Razen vzporedne se uporabljajo še druge izvedbe. Včasih se uporablja zaporedna vezava PD in PI členov.

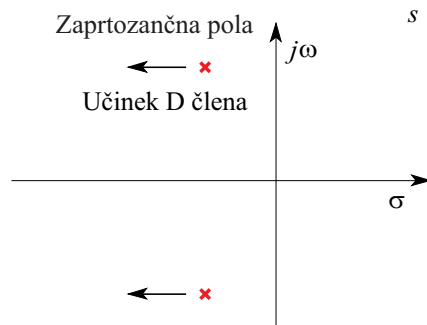
Diferencirni del regulatorja se redko uporablja samostojno (razen pri regulaciji procesov višje vrste). V povezavi s P ali PI regulatorjem pa povzroči veliko občutljivost na spremembo signala pogreška. Ker učinkuje v skladu s spremembo pogreška, deluje prediktivno in izvaja hitre popravke v regulacijski zanki, še preden pogrešek znatneje naraste. Če je proces nadkritično dušen (dominantni poli so realni), lahko ob pravilni nastavitvi PD člen v regulatorju s svojo ničlo (enačba (10.28)) delno kompenzira dominantni oz. počasni pol procesa, kar prikazuje slika 10.21. Ker sistem v odprti zanki postane hitrejši, postane hitrejši tudi povratnozančni sistem.



Slika 10.21: Kompenzacija počasnega pola procesa s PD členom regulatorja:

- a) poli in ničle v s ravnini
- b) odziv procesa na stopničasto vzbujanje (krivulja c) in odziv kompenziranega sistema (v odprti zanki) (krivulja d)

Primerno nastavljen D člen v PD ali PID regulatorju lahko poveča dušenje regulacijskega sistema. Če je zaprtozankni regulacijski sistem podkritično dušen (odziv je določen s konjugirano kompleksnim parom dominantnih polov), potem D člen pomakne pola nekoliko proti levi. Sistem zato postane bolj dušen in stabilnejši. Učinek prikazuje slika 10.22.



Slika 10.22: Učinek D člena v PD ali PID regulatorju

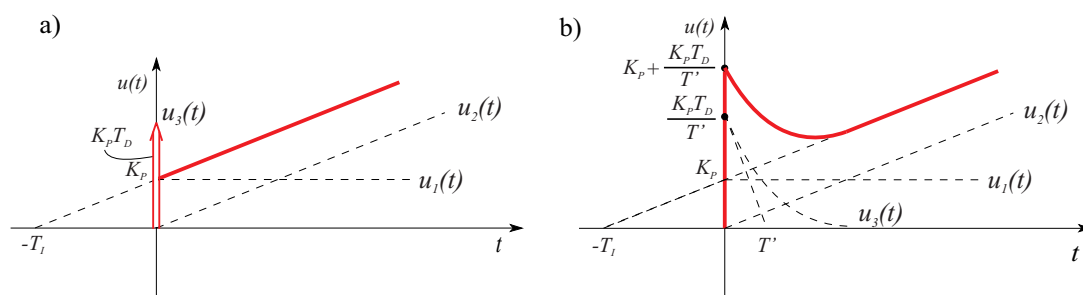
Čeprav D člen sam po sebi nima vpliva na pogrešek v ustaljenem stanju, pa zaradi povečanja stabilnosti omogoči, da lahko povečamo ojačenje P regulatorja, kar zmanjša pogrešek v ustaljenem stanju.

Do sedaj obravnavana PD in PID regulatorja sta vsebovala idealni diferenciator (člen $D0$). Zato ju imenujemo idealna PD oz. PID regulatorja. Idealne regulatorje pa v praksi ne moremo realizirati, saj bi pri stopničastem signalu povzročili v trenutku nastopa spremembe neskončno vrednost regulirne veličine. Tak regulator bi bil tudi praktično neuporaben, saj bi povzročil nasičenje izvršnih členov

in veliko ojačenje visokofrekvenčnih motenj v regulacijskem sistemu. Zato se v praksi uporablja *PID* regulator, ki vsebuje *D1* člen. Prenosna funkcija je v tem primeru

$$G_R(s) = K_P \left(1 + \frac{1}{T_I s} + \frac{T_D s}{T' s + 1} \right) \quad (10.31)$$

Učinek karakterističnih parametrov lahko predstavimo s pomočjo odziva regulatorja na stopničasti signal pogreška. Odzivu *PI* regulatorja dodamo učinek *D0* ali *D1* člena. Slika 10.23a prikazuje odziv idealnega *PID* regulatorja, slika 10.23b pa odziv realnega *PID* regulatorja na stopničasti signal pogreška.



Slika 10.23: Odziv *PID* regulatorja na stopničasti signal pogreška:

- a) idealni primer
- b) realni primer

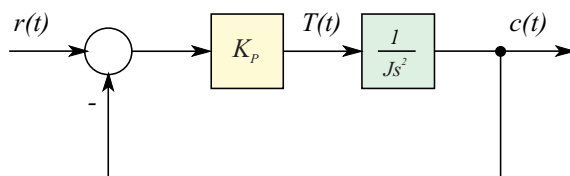
Začetno vrednost odziva lahko izračunamo z uporabo teorema začetne vrednosti ($\lim_{s \rightarrow \infty} s G_R(s) \frac{1}{s}$). Slika nazorno prikazuje hitro reagiranje *D* člena regulatorja. V ustaljenem stanju le-ta nima nikakršnega vpliva.

Primer 10.8 Proporcionalno – diferencirna regulacija zasuka rotacijskega mehanskega sistema brez dušenja.

Če ima rotacijski mehanski sistem zanemarljivo dušenje ($b \doteq 0$), ga lahko opišemo z diferencialno enačbo

$$J \frac{d^2 c(t)}{dt^2} = T(t) \quad (10.32)$$

Pri tem je $c(t)$ zasuk sistema, $T(t)$ pa vzbujalni moment. Slika 10.24 prikazuje bločno regulacijsko shemo v primeru, če bi zasuk regulirali s P regulatorjem. P regulator si lahko predstavljamo kot kombinacijo regulatorja in motorja, ki generira vzbujalni moment $T(t)$ proporcionalen pogrešku (oz. napetosti na rotorskem vzbujalnem krogu).

Slika 10.24: P regulacija zasuka

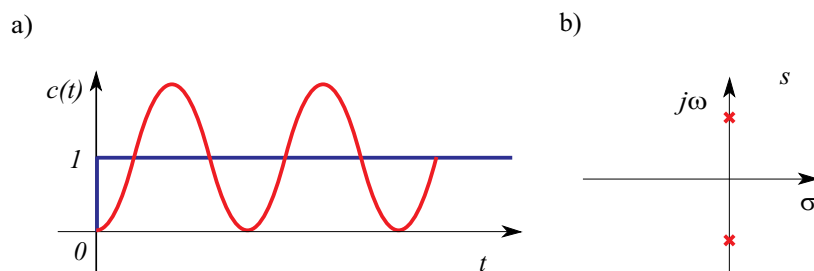
Ker je v tem primeru prenosna funkcija zaprtozančnega sistema

$$\frac{C(s)}{R(s)} = \frac{K_P}{Js^2 + K_P} \quad (10.33)$$

oz. karakteristična enačba

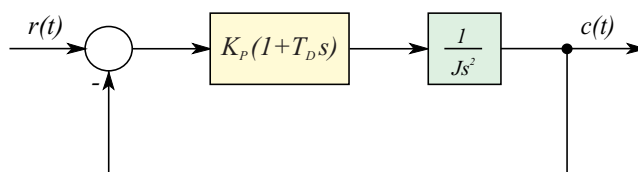
$$s^2 + \frac{K_P}{J} = 0 \quad (10.34)$$

ima sistem dva povsem imaginarna konjugirano kompleksna korena (slika 10.25b). Zato regulirana veličina niha s konstantno amplitudo okoli referenčne vrednosti pri stopničasti spremembi reference. Odziv na enotino stopnico prikazuje slika 10.25a.



Slika 10.25: a) Časovni potek zasuka pri P regulatorju
b) Lega korenov karakteristične enačbe

Z ojačenjem P regulatorja lahko vplivamo le na frekvenco nihanja (gibanje korenov po imaginarni osi), ne pa tudi na stabilnost.

Slika 10.26: PD regulacija zasuka

Spremenimo P regulator v PD regulator, kot prikazuje bločni diagram na sliki 10.26.

Prenosna funkcija zaprtozančnega sistema je v tem primeru

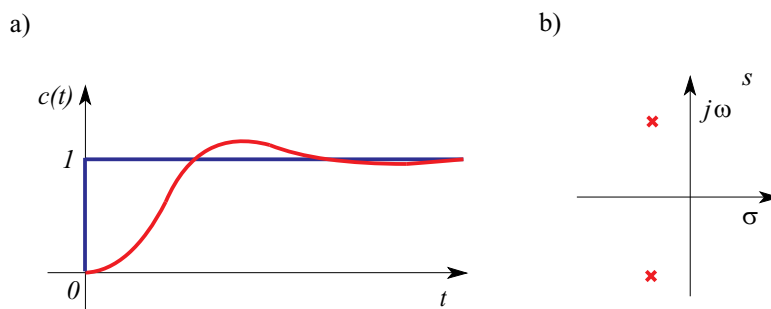
$$\frac{C(s)}{R(s)} = \frac{K_P(1 + T_D s)}{J s^2 + K_P T_D s + K_P} \quad (10.35)$$

karakteristična enačba pa je

$$s^2 + \frac{K_P T_D}{J} s + \frac{K_P}{J} = 0 \quad (10.36)$$

Zaradi D člena regulatorja se je v karakteristični enačbi pojavilo dušenje (člen $\frac{K_P T_D}{J} s$ v karakteristični enačbi). Lega korenov se je premaknila v levo polravnino (slika 10.27b). Odziv regulacijskega sistema za ta primer prikazuje slika 10.27a.

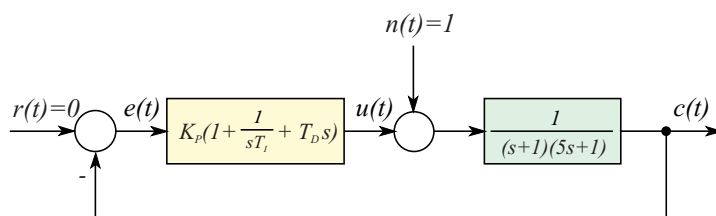
□



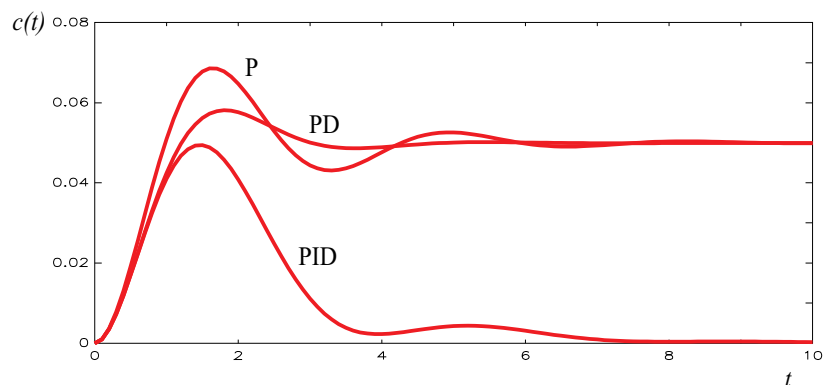
Slika 10.27: a) časovni potek zasuka pri PD regulatorju
b) Lega korenov karakteristične enačbe

Primer 10.9 Proporcionalno – integrirno – diferencirna regulacija sistema 2. reda pri stopničasti motnji na regulirnem signalu.

Slika 10.28 prikazuje bločno shemo PID regulacije sistema 2. reda pri stopničasti motnji na regulirnem signalu pri parametrih regulatorja $K_P = 19, T_I = 2, T_D = \frac{4}{19}$. Odziv oz. potek regulirane veličine za P , PD in PID regulator prikazuje slika 10.29.



Slika 10.28: Bločna shema PID regulacijskega sistema



Slika 10.29: Odziv regulacijskega sistema pri P , PD in PID regulatorju

Ker velja $e(t) = -c(t)$ vidimo, da ima P regulator pogrešek v ustaljenem stanju (ne odpravi vpliva motnje). Z D členom povečamo dušenje, tako da je odziv manj nihajoč, z I členom pa odpravimo pogrešek v ustaljenem stanju. Na ta način tudi v praksi uglašujemo PID regulatorje. $\square$

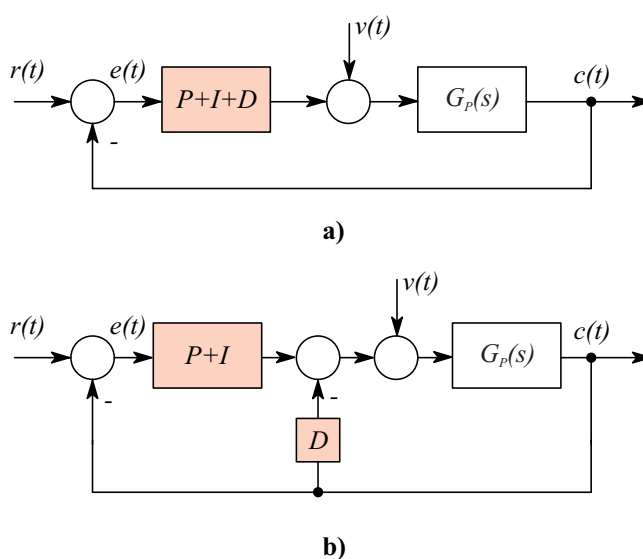
10.2.5 Modificirani PID regulatorji

Enačbi 10.29 in 10.30 predstavljata t.i. vzporedno izvedbo, ki je najpogosteje predmet teoretične obravnave. Zlasti zaradi slabosti pri sledilnem delovanju pa se v industrijskih regulatorjih taka izvedba redko uporablja, saj hitre spremembe referenčne veličine povzročijo velike spremembe regulirne veličine, kar lahko pripelje

izvršni sistem v nasičenje. Če označimo z oznakami P , I in D prenosne funkcije proporcionalnega, integrirnega in diferencirnega člena

$$P = K_P \quad I = \frac{K_P}{T_I s} \quad D = \frac{K_P T_D s}{T' s + 1} \quad (10.37)$$

potem najpogosteje uporabljeno modifikacijo vzporedne vezave (slika 10.30a) prikazuje slika 10.30b.



Slika 10.30: a) Vzporedni PID regulator
b) $PI-D$ regulator

Regulator imenujemo tudi $PI - D$. Stem ob stopničasti spremembi reference preprečimo hitro reagiranje D člena.

V regulacijskem delovanju ($r(t) = 0$, nastopajo motnje) pa sta obe izvedbi ekvivalentni.

10.3 Načrtovanje zveznih PID regulatorjev

Ker imajo PID regulatorji vnaprej določeno strukturo, se postopek sinteze regulatorjev poenostavi na določevanje njegovih parametrov: ojačenja, integrirnega

časa in diferencirnega časa (enačbi (10.30) in (10.31)). Zato pravimo, da so *PID* regulatorji parametrski regulatorji in v primeru optimalne nastavitve parametrsko optimalni regulatorji. Postopek določanja parametrov regulatorja je predvsem odvisen od tega ali imamo na voljo model procesa ali ne. Če model procesa ne obstaja, smo vezani predvsem na t.i. eksperimentalne metode, ki so znane tudi kot nastavitvena pravila. Če pa je model procesa znan, imamo večjo izbiro možnosti. Lahko uporabljamo nekatere analitične metode, ki temeljijo na doseganju želenih zaprtozančnih polov ali pa na t.i. kompenzaciji v frekvenčnem prostoru. Lahko uporabljamo nastavitvena pravila na modelu procesa, lahko uporabljamo nekatere korelacijske metode, lahko poizkušamo s simulacijo, lahko pa uporabimo tudi optimizacijo. Le-ta se skoraj vedno izvaja na računalniku s pomočjo posebnih numeričnih algoritmov. Le zelo enostavne primere je možno optimirati z analitičnim pristopom.

Kompenzacijske metode bomo spoznali pri frekvenčnih metodah. Na tem mestu pa bomo obravnavali predvsem analitične metode, nastavitvena pravila in optimizacijo. Jasno je, da mora vsak načrtovalni postopek zagotoviti stabilno delovanje. Tega pri nadaljnji obravnavi sinteze *PID* regulatorjev ne bomo več ponavljali.

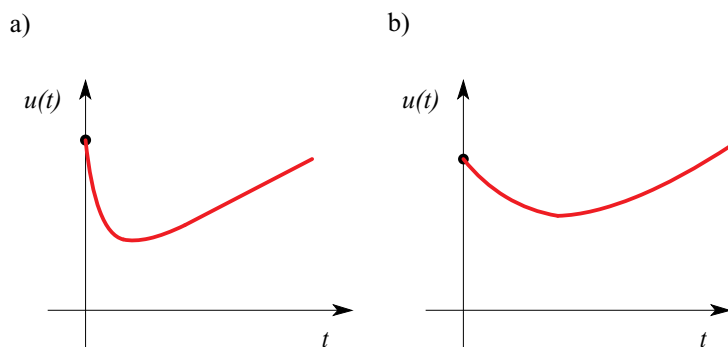
Pri večini načrtovalnih postopkov bomo upoštevali idealni *PID* regulator, ki pa ga ni možno realizirati. V praksi običajno realiziramo prenosno funkcijo

$$G_R(s) = K_P \left(1 + \frac{1}{T_I s} + \frac{T_D s}{T' s + 1} \right) \quad (10.38)$$

v kateri nastopa še parameter T' . Tega ponavadi nastavimo tako, da velja

$$T' \doteq (0.1 \text{ do } 0.3) T_D \quad (10.39)$$

Slika 10.31 prikazuje odziv *PID* regulatorja na stopničasti pogrešek pri majhni vrednosti T' (slika 10.31a) in pri večji vrednosti T' (slika 10.31b). Da se dokazati, da je ploščina pod obema krivuljama enaka. To pomeni, da je skupni učinek regulatorja na proces relativno neodvisen od časovne konstante T' . Razlikuje se le hitrost učinkovanja. Če zaradi D člena med delovanjem izvršni sistem prihaja v nasičenje, je potrebno T' povečati.



Slika 10.31: Odziv *PID* regulatorja na stopničasti pogrešek: a) majhen T'
b) velik T'

10.3.1 Analitični postopki

Regulator za premikanje polov

Če lahko proces zapišemo z enostavnim matematičnim modelom, včasih izpolnimo določene zahteve regulacijskega sistema s pomočjo analitično izračunanih parametrov. Zlasti je postopek enostaven, če so zahteve podane s pomočjo želene lege polov zaprtozančnega sistema.

Pogosto lahko proces zadovoljivo natančno opišemo z modelom 2. reda

$$G_P(s) = \frac{K}{(T_1s + 1)(T_2s + 1)} \quad (10.40)$$

Model ima tri parametre (K , T_1 , T_2). Z uporabo *PID* regulatorja, ki ima prav tako tri parametre (K_P , T_I , T_D), lahko dosežemo želeno lego treh zaprtozančnih polov.¹ Običajno izberemo en realni pol ($s = -\alpha\omega_n$) in konjugirano kompleksni par (ζ , ω_n), torej je želena karakteristična enačba

$$(s + \alpha\omega_n)(s^2 + 2\zeta\omega_ns + \omega_n^2) = 0 \quad (10.41)$$

Iz primerjave dejanskega in zelenega karakterističnega polinoma izračunamo naslednje vrednosti parametrov regulatorja:

¹Zaprtozančni sistem je tretjega reda.

$$\begin{aligned}
K_P &= \frac{T_1 T_2 \omega_n^2 (1 + 2\zeta\alpha) - 1}{K} \\
T_I &= \frac{T_1 T_2 \omega_n^2 (1 + 2\zeta\alpha) - 1}{T_1 T_2 \alpha \omega_n^3} \\
T_D &= \frac{T_1 T_2 \omega_n (\alpha + 2\zeta) - T_1 - T_2}{T_1 T_2 (1 + 2\zeta\alpha) \omega_n^2 - 1}
\end{aligned} \tag{10.42}$$

Če izberemo lastno frekvenco

$$\omega_n = \frac{T_1 + T_2}{T_1 T_2 (\alpha + 2\zeta)} \tag{10.43}$$

je parameter $T_D = 0$, torej dobimo PI regulator. Manjša lastna frekvenca pa bi zahtevala negativni parameter T_D , zato enačba (10.43) podaja minimalno vrednost oziroma spodnjo frekvenčno mejo. Želena velika vrednost ω_n pa zelo poveča ojačenje regulatorja in razširi frekvenčno področje zaprtozančnega sistema. Zgornja meja pa je vezana tudi s frekvenčno širino, v kateri je uporaben poenostavljeni model.

Kompenzacija polov procesa

Prenosna funkcija idealnega PID regulatorja

$$\begin{aligned}
G_R(s) &= K_P \left(1 + \frac{1}{T_I s} + T_D s \right) \\
&= K_P T_D \frac{\left[s + \left(\frac{1}{2T_D} + \sqrt{\frac{1}{4T_D^2} - \frac{1}{T_D T_I}} \right) \right] \left[s + \left(\frac{1}{2T_D} - \sqrt{\frac{1}{4T_D^2} - \frac{1}{T_D T_I}} \right) \right]}{s}
\end{aligned} \tag{10.44}$$

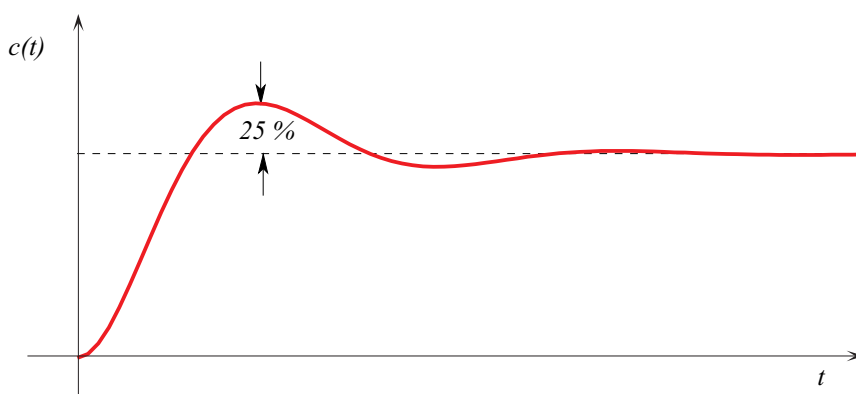
ima dve ničli in en pol v koordinatnem izhodišču. Ničli ležita na negativni realni osi za $T_I \geq 4T_D$ in sta kompleksni za $T_I < 4T_D$. Ker sta prenosni funkciji regulatorja in procesa vezani zaporedno, lahko s pomočjo ustrezne lege ničel regulatorja krajšamo dva dominantna (počasna) pola procesa. Taka kompenzacija

običajno daje dobre rezultate pri sledilnem, slabše pa pri regulacijskem delovanju. Uporablja pa se tudi nepopolna kompenzacija. Vendar tak pristop zahteva dobro poznavanje frekvenčnih metod, zato ga na tem mestu ne bomo obravnavali.

10.3.2 Nastavitvena pravila

Pri uporabi nastavitvenih pravil (Zupančič, 2010a, Ogata, 2010, Franklin in ostali, 2009) uporabljamo pokazatelje kvalitete regulacije oz. cenilke, ki smo jih spoznali pri obravnavanju sistema 2. reda. To so: čas vzpona, maksimalni prevzpon in umiritevni čas. Mnogokrat pa je pomembno predvsem hitro izreguliranje motenj.

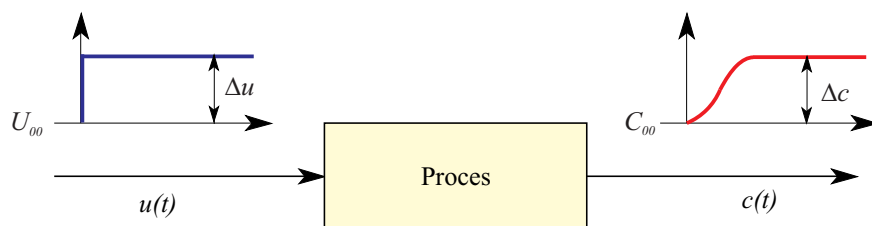
Ziegler in Nichols priporočata ustrezno nastavitev parametrov s pomočjo določenih eksperimentov na realnem objektu ali na modelu procesa. Znana sta dva postopka: določitev s pomočjo odziva procesa na stopnico in postopek z nihajnim preizkusom. Obe metodi dajeta odziv na stopničasto spremembo reference s približno 25% prevzponom, kar prikazuje slika 10.32.



Slika 10.32: Odziv regulacijskega sistema s približno 25% prevzponom

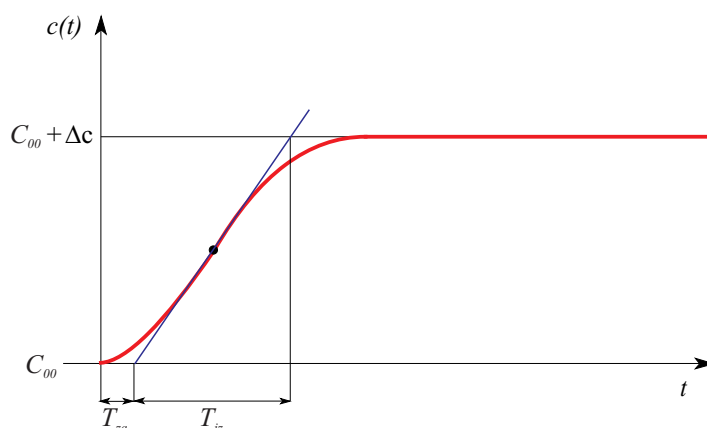
Metoda Ziegler - Nichols s pomočjo odziva na stopnico

Metoda spada med t.i. odprtozančne metode, ker zahteva eksperiment na odprtozančnem procesu. Primerna je za proporcionalne procese, ki pa morajo biti nekoliko nadkritično dušeni. Proces je potrebno vzbuditi s stopničasto spremembo in meriti odziv. Ustreden postopek prikazuje slika 10.33.



Slika 10.33: Merjenje odziva procesa na stopničasto spremembo

Tak posnetek je možno izvesti neposredno na objektu kar med obratovanjem. Regulator postavimo v položaj ročno, nakar vzbudimo proces s stopničasto spremembo vhodnega signala in posnamemo odziv. Običajno so dopustne majhne spremembe. Eksperiment lahko izvedemo tudi s simulacijo na modelu realnega procesa, če le-ta seveda obstaja. Karakteristični odziv, ki je primeren za uporabo tega kriterija, prikazuje slika 10.34.



Slika 10.34: Odziv proporcionalnega procesa

Ojačenje procesa izračunamo kot kvocient sprememb izhodnega in vhodnega signala v ustaljenem stanju

$$K = \frac{\Delta c}{\Delta u} \quad (10.45)$$

Nato narišemo tangento v prevojni točki odziva. S pomočjo presečišč tangente z absciso in premico $c(t) = C_{00} + \Delta c$ določimo čas zaostajanja T_{za} in čas izravnave T_{iz} . Na ta način dobimo tudi zelo poenostavljen model

$$\frac{C(s)}{U(s)} = \frac{K e^{-T_{za}s}}{T_{iz}s + 1} \quad (10.46)$$

Ziegler in Nichols sta na osnovi teh parametrov priporočala nastavitve parametrov *PID* regulatorja, kot prikazuje tabela 10.1.

Tabela 10.1: Nastavitev parametrov *PID* regulatorja po metodi Ziegler-Nichols s pomočjo odziva procesa na stopnico

Vrsta regulatorja	K_P	T_I	T_D
P	$\frac{1}{K} \frac{T_{iz}}{T_{za}}$	∞	0
PI	$\frac{0.9}{K} \frac{T_{iz}}{T_{za}}$	$3.3T_{za}$	0
PID	$\frac{1.2}{K} \frac{T_{iz}}{T_{za}}$	$2T_{za}$	$0.5T_{za}$

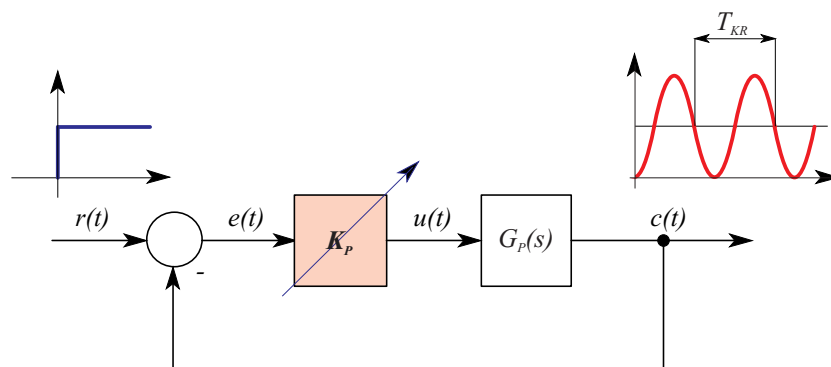
Prenosna funkcija tako nastavljenega *PID* regulatorja je

$$\begin{aligned} G_R(s) &= K_P \left(1 + \frac{1}{T_I s} + T_D s \right) \\ &= \frac{1.2}{K} \frac{T_{iz}}{T_{za}} \left(1 + \frac{1}{2T_{za}s} + 0.5T_{za}s \right) = \\ &= \frac{0.6}{K} T_{iz} \frac{(s + \frac{1}{T_{za}})^2}{s} \end{aligned} \quad (10.47)$$

Tako nastavljeni *PID* regulator ima pol v koordinatnem izhodišču in dvojno ničlo pri $s = -\frac{1}{T_{za}}$.

Metoda Ziegler - Nichols s pomočjo nihajnega preizkusa

Nihajni preizkus je metoda, ki zahteva poizkus v zaprti zanki. Pri metodi nihajnega preizkusa nastavimo $T_I = \infty$ in $T_D = 0$, torej uporabimo v zanki le proporcionalni regulator. Regulacijskemu sistemu pri poljubnem vzburjanju (običajno konstantne spremembe reference ali motnje) povečujemo ojačenje K_P toliko časa, da regulirana veličina nedušeno zaniha. Ustrezen postopek prikazuje slika 10.35.



Slika 10.35: Nihajni preizkus

Nihajni preizkus je možno izvesti le v primeru, če je proces vsaj tretjega reda (oz. razlika med številom polov in ničel je vsaj tri) ali če vsebuje mrtvi čas. Tako je npr. proces $G_P(s) = \frac{(s+2)(s+3)}{s(s+1)(s+5)}$ stabilen za katerokoli ojačenje (to lahko ugotovimo z Routh-ovim kriterijem). Ojačenju, pri katerem regulacijski sistem nedušen zaniha, pravimo kritično ojačenje K_{KR} . Iz posnetega odziva ali s pomočjo časovnega merjenja dobimo periodo nihanja, ki jo imenujemo kritična perioda T_{KR} . Na osnovi kritičnega ojačenja in kritične periode sta Ziegler in Nichols predlagala parametre PID regulatorja, ki jih prikazuje tabela 10.2.

Tabela 10.2: Nastavitev parametrov z nihajnim preizkusom po metodi Ziegler - Nichols

Vrsta regulatorja	K_P	T_I	T_D
P	$0.5K_{KR}$	∞	0
PI	$0.45K_{KR}$	$0.83T_{KR}$	0
PID	$0.6K_{KR}$	$0.5T_{KR}$	$0.125T_{KR}$

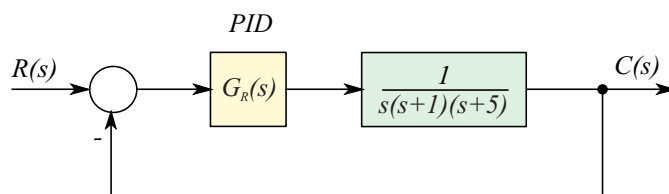
Prenosna funkcija tako uglašnega regulatorja ima obliko

$$\begin{aligned}
 G_R(s) &= K_P \left(1 + \frac{1}{T_I s} + T_D s \right) = \\
 &= 0.6K_{KR} \left(1 + \frac{1}{0.5T_{KR}s} + 0.125T_{KR}s \right) = \\
 &= 0.075K_{KR}T_{KR} \frac{\left(s + \frac{4}{T_{KR}} \right)^2}{s}
 \end{aligned} \tag{10.48}$$

Tako nastavljen *PID* regulator ima en pol v koordinatnem izhodišču in dvojno ničlo pri $s = -\frac{4}{T_{KR}}$.

Pravila Zieglerja in Nicholisa so narejena za cel spekter sorodnih procesov. Pri stopničasti referenci dajo 10% - 60% prevzpon. Povprečni prevzpon, ugotovljen pri uporabi na različnih procesih, je 25%. Zato je razumljivo, da nobeno nastavitveno pravilo ne da optimalnih rezultatov v konkretnem primeru. Njihov pomen pa je v tem, da dobimo okvirne vrednosti parametrov. Le-te potem eksperimentalno na objektu ali simulacijsko (če imamo model) še izboljšamo. Slabost metod Zieglerja in Nicholisa pa je v tem, da ni definirano, za kakšne vhodne signale naj regulator dobro deluje. Optimalno delovanje za stopničasto spremembo reference zahteva namreč drugačno nastavitve parametrov kot optimalno delovanje pri pojavu motnje.

Primer 10.10 Potrebno je načrtati *PID* regulator za sistem, ki ga prikazuje slika 10.36.



Slika 10.36: *PID* regulacijski sistem

Ker je proces 1. vrste, pride v poštev le nihajni preizkus. Ker je model procesa znan, lahko kritično ojačenje in kritično periodo določimo z Routh-ovim stabilnostnim kriterijem. Pri proporcionalnem regulatorju $G_R(s) = K_P$ je zaprtozančna prenosna funkcija

$$\frac{C(s)}{R(s)} = \frac{K_P}{s(s+1)(s+5) + K_P} \quad (10.49)$$

in karakteristična enačba

$$s^3 + 6s^2 + 5s + K_P = 0 \quad (10.50)$$

Routh-ova shema je naslednja:

$$\begin{array}{ccc}
s^3 & 1 & 5 \\
s^2 & 6 & K_P \\
s^1 & \frac{30-K_P}{6} & \\
s^0 & K_P &
\end{array}$$

Iz pogoja, da morajo biti pri stabilnem sistemu vsi elementi v prvi koloni pozitivni, dobimo, da je kritično ojačenje

$$K_{KR} = 30$$

Ob upoštevanju $K_P = K_{KR}$ določimo iz kolone pri s^2 kritično periodo na naslednji način:

$$\begin{aligned}
6s^2 + K_P &= 0 \\
6s^2 + 30 &= 0 \\
s^2 &= -5 \\
s &= \pm j\sqrt{5} \implies \omega_{KR} = \sqrt{5} \implies T_{KR} = \frac{2\pi}{\sqrt{5}} = 2.81
\end{aligned} \tag{10.51}$$

Z uporabo omenjenih pravil pridemo do parametrov regulatorja

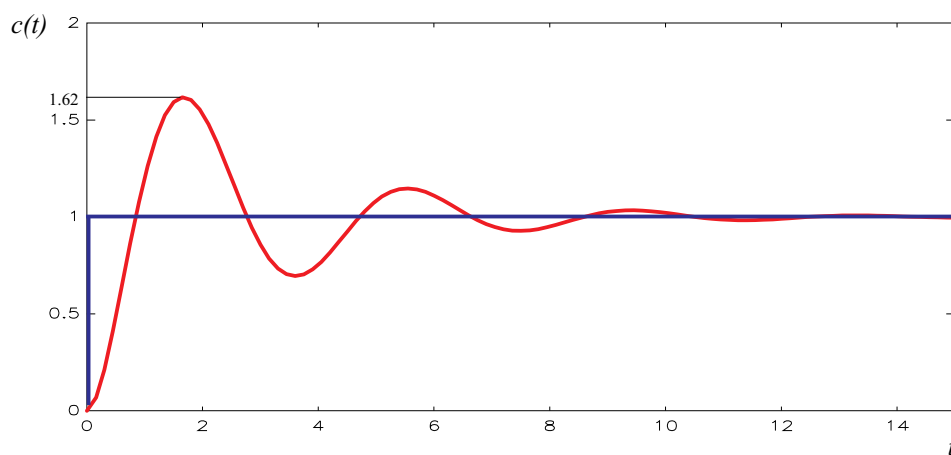
$$\begin{aligned}
K_P &= 0.6K_{KR} = 18 \\
T_I &= 0.5T_{KR} = 1.405 \\
T_D &= 0.125T_{KR} = 0.351
\end{aligned} \tag{10.52}$$

Torej je prenosna funkcija regulatorja

$$\begin{aligned}
G_R(s) &= 18\left(1 + \frac{1}{1.405s} + 0.351s\right) = \\
&= \frac{6.322(s + 1.4235)^2}{s}
\end{aligned} \tag{10.53}$$

Odziv regulacijskega sistema dobimo najlaže s simulacijo. Prikazuje ga slika 10.37 pri stopničastem referenčnem signalu. Vidimo, da ima sistem kar 62% prevzpon.

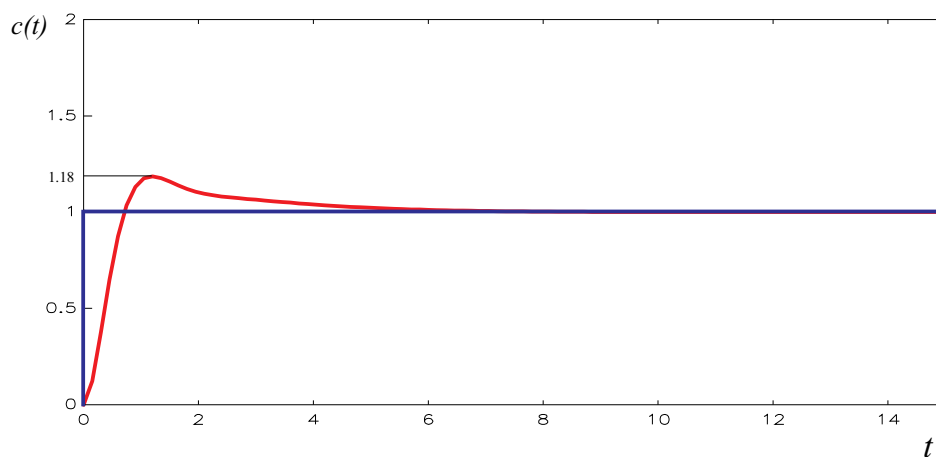
Z računalniško simulacijo lahko kar s poizkušanjem hitro pridemo do ustrežnejših parametrov regulatorja oz. do ustrežnejšega odziva. Če pri istem ojačenju regulatorja premaknemo ničlo proti koordinatnemu izhodišču, se nekoliko zmanjša učinek pola zaradi integrirnega člena in regulacijski sistem postane bolj dušen,

Slika 10.37: Potek regulirane veličine v PID regulacijskem sistemu

poli zaprtozančnega sistema se pomaknejo v levo. Če prestavimo dvojno ničlo v -0.65 je prenosna funkcija regulatorja

$$G_R(s) = 13.846 \frac{(s + 0.65)^2}{s} = 18 \left(1 + \frac{1}{3.077s} + 0.7692s \right) \quad (10.54)$$

Torej se je res zmanjšal učinek I člena (manjši prevzpon) in povečal učinek D dela (večje dušenje). Ustrezni odziv prikazuje slika 10.38. Prevzpon se je zmanjšal na cca. 18%, kar je zadovoljivo.



Slika 10.38: Potek regulirane veličine pri spremenjenih parametrih

□

Metoda Chien - Hrones - Reswick

Priporočila, ki so jih izdelali Chien, Hrones in Reswick s pomočjo simulacijskih študij, so primerna za načrtovanje *PID* regulatorja za proporcionalne sisteme višjega reda. Pred priporočili Zieglerja in Nicholisa imajo to prednost, da upoštevajo ali regulacijski sistem deluje v regulacijskem ali sledilnem načinu delovanja (ali je vhodni signal referenca ali motnja na vходу v proces). Nastavitve tudi upoštevajo zahtevo po odzivu z 20% prevzponom ali pa zahtevo po aperiodičnem odzivu s čim krajšim umiritvenim časom. Tako kot pri prvi metodi Zieglerja in Nicholisa, potrebujemo odziv procesa na stopničasto spremembo, iz katerega določimo ojačenje procesa K , čas zaostajanja T_{za} in čas izravnave T_{iz} . Priporočila za nastavitve pa podaja tabela 10.3.

Tabela 10.3: Uglasčevanje *PID* regulatorja s pravili Chien-Hrones-Reswick

Regulator		Aperiodični odziv z najkrajšim umiritvenim časom		Najkrajši umiritveni čas z 20% prevzponom	
		motnja	referenca	motnja	referenca
P	K_P	$\frac{0.3}{K} \frac{T_{iz}}{T_{za}}$	$\frac{0.3}{K} \frac{T_{iz}}{T_{za}}$	$\frac{0.7}{K} \frac{T_{iz}}{T_{za}}$	$\frac{0.7}{K} \frac{T_{iz}}{T_{za}}$
PI	K_P	$\frac{0.6}{K} \frac{T_{iz}}{T_{za}}$	$\frac{0.35}{K} \frac{T_{iz}}{T_{za}}$	$\frac{0.7}{K} \frac{T_{iz}}{T_{za}}$	$\frac{0.6}{K} \frac{T_{iz}}{T_{za}}$
	T_I	$4T_{za}$	$1.2T_{iz}$	$2.3T_{za}$	T_{iz}
PID	K_P	$\frac{0.95}{K} \frac{T_{iz}}{T_{za}}$	$\frac{0.6}{K} \frac{T_{iz}}{T_{za}}$	$\frac{1.2}{K} \frac{T_{iz}}{T_{za}}$	$\frac{0.95}{K} \frac{T_{iz}}{T_{za}}$
	T_I	$2.4T_{za}$	T_{iz}	$2T_{za}$	$1.35T_{iz}$
	T_D	$0.42T_{za}$	$0.5T_{za}$	$0.42T_{za}$	$0.47T_{za}$

10.3.3 Optimizacija s pomočjo integralnih cenilk

Ker se zaradi izredno sposobnih digitalnih računalnikov pri načrtovanju parametrov *PID* regulatorjev vse bolj uporablja računalniška optimizacija, bomo podali njene osnovne značilnosti. Čeprav jo je v principu možno izvajati tudi neposredno na

realnem objektu, jo v praksi skoraj vedno uporabljamo na modelu realnega objekta. Potrebujemo torej model realnega procesa.

Optimizacija regulacijskega sistema je postopek, s pomočjo katerega načrtamo regulator tako, da le-ta minimizira ali maksimizira (odvisno od situacije) določeno cenilko (kriterijsko funkcijo, performančni indeks) v nekem prehodnem pojavu. Le-ta je posledica spremenjene referenčne veličine ali motnje. Cenilka je odvisna od signala pogreška in od regulirne veličine in je običajno integralskega tipa. Razen cenilke lahko v optimizacijske postopke vključimo tudi omejitve. Stem upoštevamo nekatere realne lastnosti regulacijskih sistemov, ki jih sicer pri teoretični obravnavi radi zanemarimo (na primer fizične omejitve izvršnih sistemov, merilnih sistemov in regulatorjev 4-20ma). Ker se v primeru *PID* regulatorja problem poenostavi tako, da je potrebno določiti le parametre regulatorja, ne pa tudi strukture, pravimo taki optimizaciji parametrska optimizacija.

Samo najenostavnejše probleme, ki so ponavadi zgolj teoretičnega pomena, je možno rešiti analitično. V tem primeru cenilko, ki jo določimo v nekem prehodnem pojavu (npr. $C = \int e^2(t)dt$ pri stopničasti spremembi reference), zapišemo kot funkcijo parametrov *PID* regulatorja

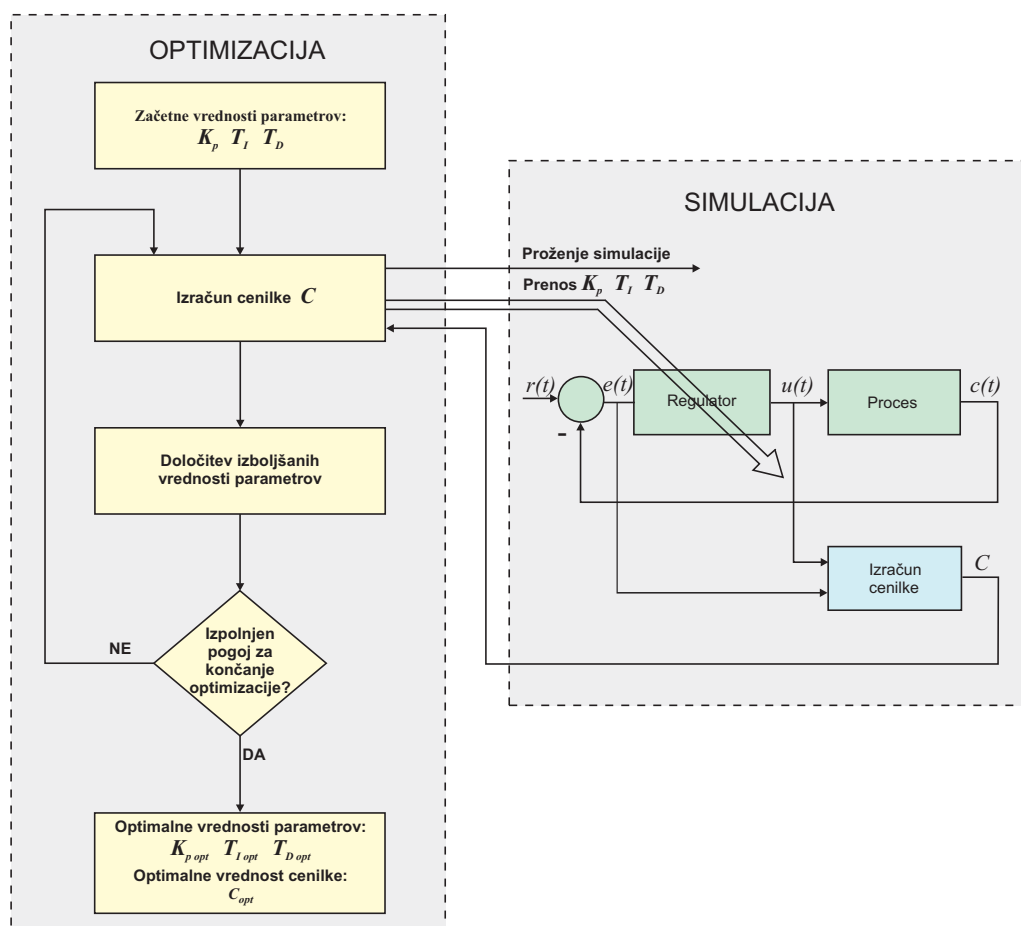
$$C = f(K_P, T_I, T_D) \quad (10.55)$$

Če želimo določiti npr. optimalno ojačenje regulatorja, je potrebno cenilko odvajati po ojačenju in odvod izenačiti z nič

$$\frac{\partial C}{\partial K_P} = 0 \quad \implies \quad K_{Popt} \quad (10.56)$$

Za realne probleme pa danes v glavnem uporabljamo računalniško parametrsko optimizacijo. To je poseben algoritem, ki pri ponavljajočih prehodnih pojavih toliko časa spreminja parametre regulatorja, da na koncu zagotovi minimum cenilke. Glavna slabost postopka je velika potratnost računalniškega časa, saj vsaka nova kombinacija parametrov regulatorja zahteva nov simulacijski tek oz. nov prehodni pojav, v katerem se ovrednoti vrednost cenilke. Torej je pri reševanju realnih problemov potrebno regulacijski sistem simulirati, saj je nemogoče analitično izraziti vrednost cenilke v odvisnosti od parametrov regulatorja. Ker je pri reševanju kompleksnejših problemov potrebno izvesti tudi nekaj sto poizkusov - simulacijskih tekov, je razumljivo, kako neobhoden je sposoben računalnik. Z razvojem računalnikov pa ta slabost postaja vse manj pomembna. Glavna prednost takega pristopa pa je, da je uporaben tudi za nelinearne sisteme.

Koncept računalniške optimizacije, ki uporablja simulacijo, prikazuje slika 10.39. Tak koncept je zlasti možno učinkovito uporabiti v okolju Matlab-Simulink. Optimizacijo izvaja okolje Optimization toolbox (npr. funkcija `fminsearch`), simulacijo pa okolje Simulink.

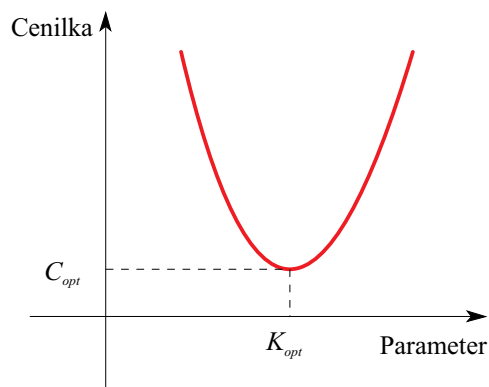


Slika 10.39: Optimizacija s pomočjo simulacije

Cenilke

Cenilka je število, ki pove, kako dobro deluje regulacijski sistem v nekem prehodnem pojavu. V splošnem lahko tudi že znani pokazatelji (maksimalni prevzpon, čas vzpona in čas umiritve) nastopajo v cenilki, vendar v zvezi z računalniško optimizacijo največkrat uporabljamo integralske cenilke. Za cenilko je izjemno pomembno, da kaže določeno selektivnost, kar pomeni, da ima pri optimalnih

vrednostih parametrov jasno izražen optimum. Ustrezno obliko v odvisnosti od enega parametra prikazuje slika 10.40.



Slika 10.40: Vrednost cenilke v odvisnosti od parametra

Literatura priporoča naslednje integralske cenilke pri optimizaciji regulacijskih sistemov:

$$\begin{aligned}
 C_1 = ISE &= \int_0^\infty e^2(t) dt && \text{integral square error} \\
 C_2 = ITSE &= \int_0^\infty t e^2(t) dt && \text{integral time square error} \\
 C_3 = IAE &= \int_0^\infty |e(t)| dt && \text{integral absolute error} \\
 C_4 = ITAE &= \int_0^\infty t |e(t)| dt && \text{integral time absolute error} \\
 C_5 = S_{eu} &= \int_0^\infty [e^2(t) + R(u(t) - u(\infty))^2] dt && (10.57)
 \end{aligned}$$

Pri tem je pogrešek $e(t)$ razlika med referenčno in regulirano veličino, $u(t)$ je regulirna veličina. Izbira ustrezne cenilke je seveda ključna pri optimiranju regulacijskega sistema. Neskončni integracijski interval pa zamenjamo s tako končno dolžino, da prehodno pojav izzveni in da ustaljeni pogrešek postane nič ali zelo majhen. Ker je težko ugotoviti, katera cenilka je primernejša, lahko uporabimo več cenilk in izberemo najprimernejšo rešitev. Kvadriranje v cenilki ali računanje absolutne vrednosti se uporablja zlasti pri podkritično dušenih prehodnih pojavih, ko so pogreški pozitivni in negativni. Množenje s časom se uporablja, v kolikor želimo bolj utežiti del po prehodnem pojavu, t.j. ustaljeno stanje. V enačbi 10.57 je R utežni faktor regulirne veličine, $u(\infty)$ vrednost regulirne veličine v ustaljenem stanju. Ker v cenilki upoštevamo tudi regulirno veličino, s tem zmanjšamo porabo energije, ki jo je potrebno v določenem prehodnem pojavu dovajati

procesu. Zato postane regulacijski sistem nekoliko počasnejši oz. bolj dušen v prehodnem pojavu. Na regulirni veličini se to odraža tako, da so njene vrednosti v trenutkih hitrih sprememb reference ali nastopa motnje tem manjše, čim večji je utežni faktor R .

Omejitve

Razen cenilke lahko v postopkih računalniške optimizacije vpeljemo tudi omejitve. Tipična omejitev je omejitev regulirne veličine zaradi končnega območja izvršnega sistema. Tudi maksimalni dopustni prevzpon lahko vključimo kot omejitev. Optimizacijski algoritem mora izračunati take optimalne vrednosti parametrov, da niso kršeni omejitveni pogoji.

Pogoj za končanje optimizacije

Računalniški optimizacijski algoritmi omogočajo več različnih pogojev za končanje optimizacije. Ker optimizacija poteka v iteracijah (ena iteracija je del algoritma, ki določi izboljšane vrednosti parametrov), je možno podati število iteracij. Nadalje je možno optimizacijo zaključiti, če je med dvema iteracijama sprememba cenilke manjša od predpisanega praga ali če je normirana vrednost spremembe parametrov pod predpisanim pragom.

Postopek računalniške optimizacije bomo prikazali s pomočjo primera.

Primer 10.11 Potrebno je načrtati P regulator za proces

$$G_P(s) = \frac{\omega_n^2}{s(s^2 + 2\zeta\omega_n s + \omega_n^2)} \quad \omega_n = 1, \zeta = 0.7 \quad (10.58)$$

Ker je kritično ojačenje regulacijskega sistema $K_{KR} = 1.4$, je po nastavitvenem Ziegler-Nichols-ovem pravilu ojačenje proporcionalnega regulatorja $K_P = \frac{K_{KR}}{2} = 0.7$. Vendar je potek regulirane veličine pri stopničasti spremembi reference zelo nihajoč, prevzpon pa je kar 40% (slika 10.41, krivulja a). Zato smo uporabili računalniško optimizacijo v okolju Matlab-Simulink (slika 10.39). Prva izbrana cenilka je bila

$$C_1 = \text{IAE} = \int_0^{30} |e(t)| dt \quad (10.59)$$

Začetno vrednost ojačenja regulatorja smo izbrali kot v prejšnjem primeru, t.j. $K_{P_{zač}} = 0.7$. Optimizacijski algoritem je modificiral ojačenje $K_{P_{zač}} = 0.7$ v ojačenje $K_{P_{opt}} = 0.39$, cenilka IAE pa se je pri tem spremenila od 4.16 na 3.27. Prevezpon se je zmanjšal na 12% (slika 10.41, krivulja b).

Druga izbrana cenilka pa je bila

$$C_2 = ITAE = \int_0^{30} t |e(t)| dt \quad (10.60)$$

Ta cenilka je še povečala dušenje v regulacijskem sistemu in zmanjšala ojačenje na $K_{P_{opt}} = 0.32$. Njena vrednost se je pri tem od 21.3 pri $K_{P_{zač}} = 0.7$ spremenila na 7.39 pri $K_{P_{opt}} = 0.32$, prevzpon pa se je nadalje zmanjšal na 4% (slika 10.41, krivulja c). Ustrezno primerjavo vseh teh postopkov omogočata tabela 10.4 in slika 10.41. Obe cenilki imata dobro selektivnost, druga pa v splošnem daje bolj dušen potek regulirane veličine.

Tabela 10.4: Primerjava treh postopkov za določitev P regulatorja

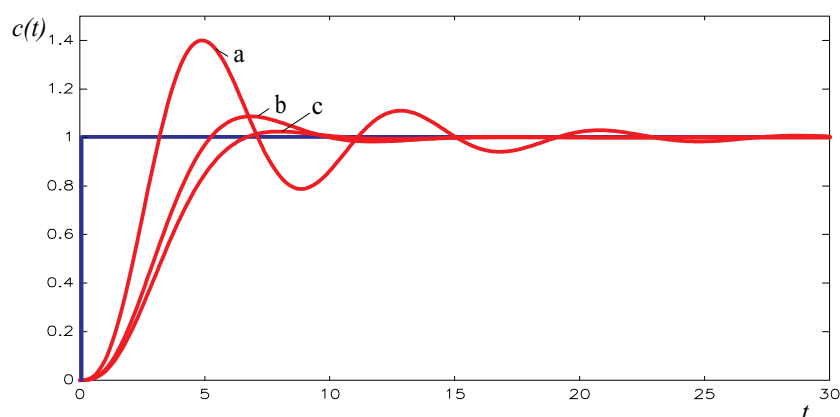
		Nihajni preizkus	Optimizacija po IAE= $\int e(t) dt$		Optimizacija po ITAE= $\int t e(t) dt$	
$K_{P_{zač}}$	$K_{P_{opt}}$	0.7	0.7	0.39	0.7	0.32
$C_{1zač}$	C_{1opt}	4.16	4.16	3.27		
$C_{2zač}$	C_{2opt}	21.3			21.3	7.39
$M_p [\%]$		40	12		4	
$t_r(0 - 100\%) [s]$		3.18	4.8		6.1	
$t_s(2\%) [s]$		21.9	12.4		9.25	

□

10.4 Izvedbe PID regulatorjev

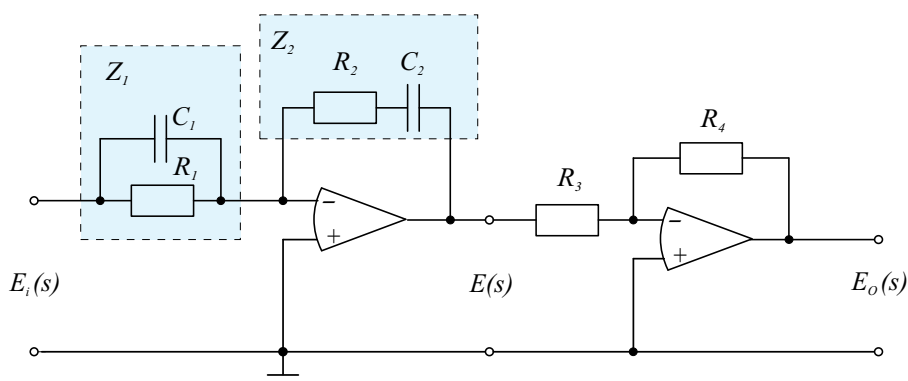
Elektronska izvedba

Kot smo omenili se največ uporabljajo električni regulatorji, ki pa so lahko elektronski ali računalniški (digitalni). Elektronski PID regulator realiziramo s po-



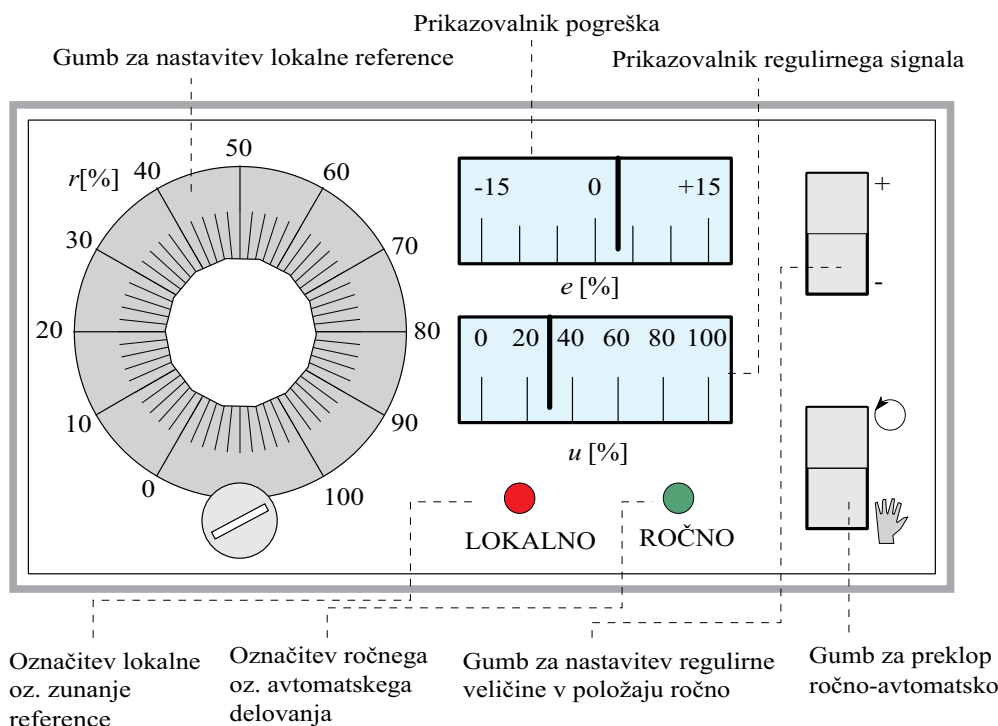
Slika 10.41: Poteki regulirane veličine: a) $K_P = 0.7$ b) $K_P = 0.39$ c) $K_P = 0.32$

močjo operacijskih ojačevalnikov z ustreznimi elementi v direktni in povratni veji. Eno od izvedb prikazuje slika 10.42.



Slika 10.42: Izvedba elektronskega PID regulatorja

Slika 10.43 prikazuje izgled čelne plošče enostavnega elektronskega PID regulatorja, ki je namenjen vgradnji v komandne pulte za vodenje in nadzor industrijskih procesov. Na čelni plošči se nahaja gumb za nastavljanje reference, prikazovalnik pogreška in regulirne veličine, gumb za preklop ročno - avtomatsko ter gumb za nastavljanje regulirne veličine v položaju ročno. Nastavitev parametrov regulatorja pa ni naloga operaterja, zato se ustrezni gumbi nahajajo na zadnji strani, kjer so tudi priključnice za pričvrstitev žic.

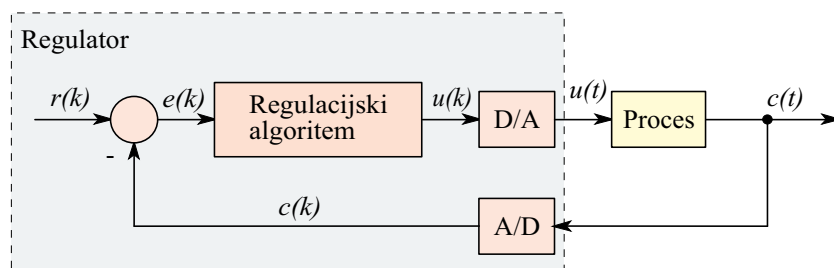


Slika 10.43: Izgled čelne plošče enostavnega elektronskega *PID* regulatorja

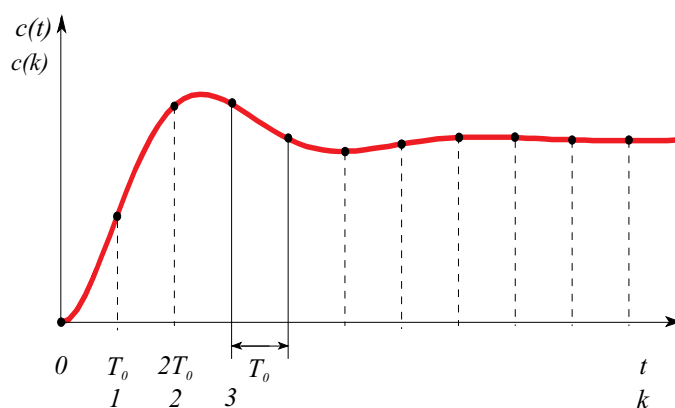
Računalniška (digitalna) izvedba

Klasične hidravlične, pnevmatske in analogne elektronske regulatorje danes vse bolj zamenjujejo digitalno (računalniško) izvedeni regulacijski algoritmi. Taki regulatorji so izjemno fleksibilni, saj je možno s spreminjanjem podatkov ali programa enostavno spreminjati njihove parametre ali celo njihovo strukturo. Zato pa potrebujemo dodatne pretvorniške elemente, ki omogočajo prenos vzorcev regulirane veličine v računalnik in prenos izračunane regulirne veličine na izvršni sistem. To so A/D (analogno-digitalni) in D/A (digitalno-analogni) pretvorniki. Ustrezno bločno shemo prikazuje slika 10.44, povezavo med zvezno ($c(t)$) in vzorčeno regulirano veličino ($c(k)$) pa slika 10.45.

Ker A/D pretvornik jemlje vzorce s časom vzorčenja T_0 , so $r(k)$, $e(k)$, $u(k)$ in $c(k)$ ustrezni vzorci v trenutkih $0, T_0, 2T_0, \dots$. Da lahko programsko realiziramo regulator, je potrebno enačbo zveznega *PID* algoritma



Slika 10.44: Bločna shema regulacijskega sistema z digitalno izvedenim regulatorjem



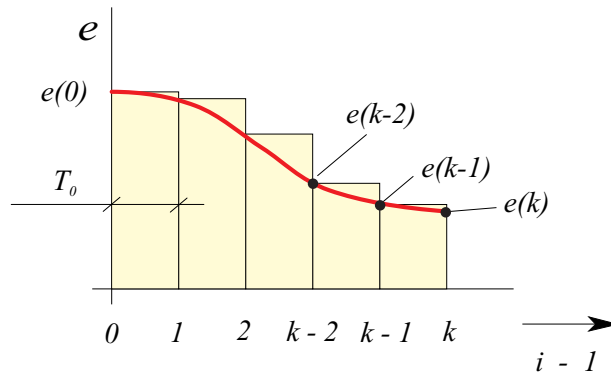
Slika 10.45: Povezava med zvezno in vzorčeno veličino

$$u(t) = K_P \left(e(t) + \frac{1}{T_I} \int e(t) dt + T_D \frac{de(t)}{dt} \right) \quad (10.61)$$

diskretizirati. Če zamenjamo integral z vsoto (slika 10.46), odvod pa z diferenco, dobimo enačbo

$$u(k) = K_P \left\{ e(k) + \frac{T_0}{T_I} \sum_{i=1}^k e(i-1) + \frac{T_D}{T_0} [e(k) - e(k-1)] \right\} \quad (10.62)$$

Če zapišemo enačbo (10.62) še za $(k-1)$ in odštejemo obe enačbi, dobimo diskretni *PID* algoritem v obliki



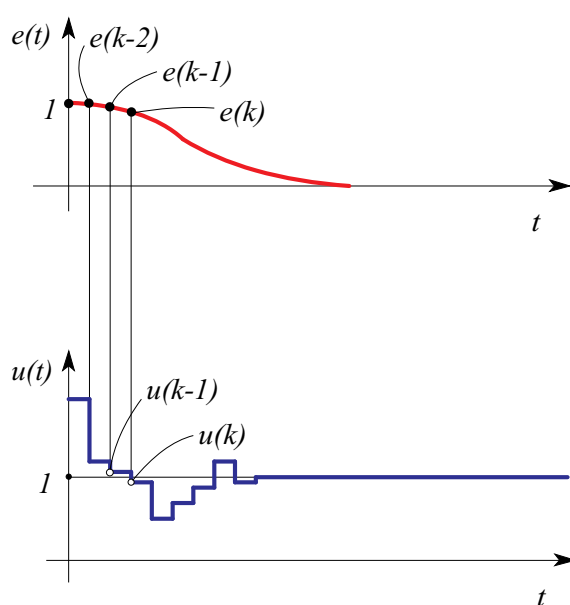
Slika 10.46: Numerično integriranje pogreška

$$u(k) = u(k-1) + q_0 e(k) + q_1 e(k-1) + q_2 e(k-2) \quad (10.63)$$

kjer so

$$\begin{aligned} q_0 &= K \left(1 + \frac{T_D}{T_0} \right) \\ q_1 &= -K \left(1 + 2 \frac{T_D}{T_0} - \frac{T_0}{T_I} \right) \\ q_2 &= K \frac{T_D}{T_0} \end{aligned} \quad (10.64)$$

Enačba (10.63) jasno kaže, kako se programsko realizira *PID* regulator. Za izračun sedanje vrednosti regulirne veličine $u(k)$ potrebujemo preteklo vrednost $u(k-1)$ ter sedanjo vrednost signala pogreška $e(k)$ in dve pretekli vrednosti $e(k-1), e(k-2)$. Slika 10.47 prikazuje tipični signal pogreška in regulirni signal pri računalniškem vodenju procesov pri spremembi referenčnega signala za ena pri proporcionalnem procesu z ojačenjem 1.



Slika 10.47: Signal pogreška in regulirni signal pri vodenju z diskretnim *PID* regulatorjem

Literatura

Aburdene, M.F. (1988), *Digital Continuous System Simulation*, Wm.C. Brown Publishers, Dubuque, Iowa, USA.

ANSI/ISA-S88.01 (1995), *Batch Control*, Part 1: Models and Terminology, ISA. Research Triangle Park, (glej tudi IEC 61512-01).

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.

Sawyer, P. (1993), *Computer-Controlled Batch Processing*. Institution of Chemical Engineers, Rugby, Warwickshire.

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.

Stenerson, J. (1993), *Fundamentals of Programmable Logic Controllers, Sensors, and Communications*. Regents/Prentice-Hall, Englewood Cliffs.

Strmčnik, S., R. Hanus, D. Juričič, 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. (1989), *Sinteza simulacijskega jezika pri računalniško podprtem načrtovanju sistemov avtomatskega vodenja*, Doktorska disertacija, Fakulteta za elektrotehniko in računalništvo, Univerza v Ljubljani, Ljubljana, Slovenija.

Zupančič B. (1992), *SIMCOS- jezik za simulacijo zveznih in diskretnih dinamičnih sistemov*, Fakulteta za elektrotehniko in računalništvo, Univerza v Ljubljani, Slovenija.

Zupančič, B. (2010): *Simulacija dinamičnih sistemov*, Založba Fakultete za elektrotehniko in Fakultete za računalništvo in informatiko, Univerza v Ljubljani, SLO

Zupančič, B. (2010a), *Zvezni regulacijski sistemi, I.del*, Založba Fakultete za elektrotehniko in Fakultete za računalništvo in informatiko, Univerza v Ljubljani, SLO

Zupančič, B. (2010b), *Zvezni regulacijski sistemi, II.del*, Založba Fakultete za elektrotehniko in Fakultete za računalništvo in informatiko, Univerza v Ljubljani, SLO