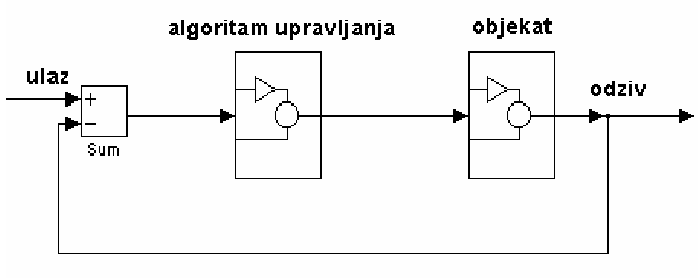
IV
Uticaj parametara PID regulatora i vremenskog kašnjenja na odziv i amplitudno-faznu karakteristiku sistema Simulink

Uvod

Prilikom automatiziranja nekog objekta često koristimo upravljanje sa negativnom povratnom spregom kao na slijedećoj slici:

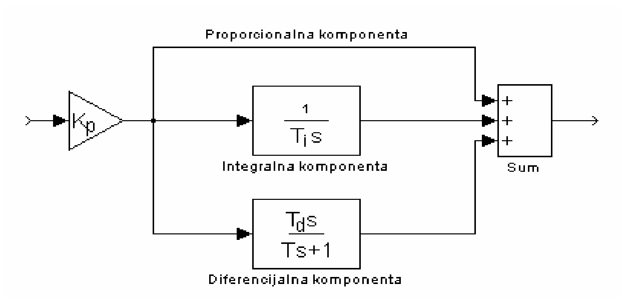


Slika 4.1

Blok *algoritma upravljanja* ima za cilj da natjera sistem da generira zadovoljavajući odziv. Najbitniji elementi koji utiču na kvalitet odziva su: *stabilnost* sistema, *preskok (najčešće prvi)*, *vrijeme smirivanja* i *greška u stacionarnom stanju* odziva sistema. Jedan od prvih algoritama upravaljanja, koji se i danas najviše koristi u regulaciji, je **PID** algoritam. Ovaj algoritam, izražen u domenu Laplasove kompleksne varijable *s*, ima oblik :

$$W_r(s) = K_p(1 + \frac{1}{T_i s} + \frac{T_d s}{Ts + 1})$$

ili prikazan u obliku modela:



Slika 4.2

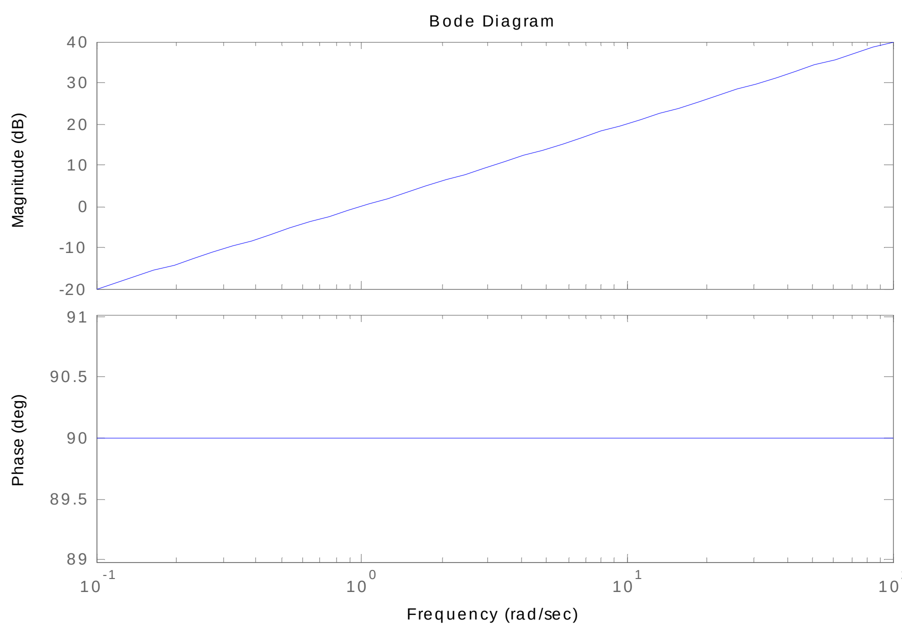
Vidimo da ovaj regulator posjeduje tri komponente: *proporcionalnu* K_p, *integralnu* T_i i *diferencijalnu* T_d, gdje svaka od ovih komponenti ima svoju zadaću.

Treba istaći da se u praksi ne može realizirati idealni diferencijator $T_d s$ zato što u ovom obliku predstavlja nekauzalni sistem. Iz ovog razloga se koristi realni diferencijator $T_d s/(Ts+1)$, gdje je T dovoljno maleno da bi nas ovakav oblik diferencijatora zadovoljio u smislu izlaznog signala. Ovakav oblik diferencijatora ujedno prigušuje visoke frekvencije (sve frekvencije iznad $1/T$) koje se javaljaju u šumovima, za razliku od idealnog diferencijatora koji ovakve signale dodatno pojačava.

Povećavanjem parametra K_p smanjujemo grešku stacionarnog stanja, odnosno stanja poslije završetka svih prelaznih procesa, s jedne strane, i ekspandiramo amplitudno-faznu karakteristiku sistema, što bi značilo da bi za sisteme trećeg i višeg reda (sisteme koji obuhvataju tri ili više kvadranata) moglo doći do neželjenog obuhvatanja kritične tačke (-1, j0), odnosno odvođenja sistema u nestabilnost, s druge strane. Očigledno je da nam ova dva limita i određuju interval optimalnog K_p.

Integralna komponenta služi za potpuno anuliranje greške stacionarnog stanja. Naime, ako postoji razlika izlaza sistema i zadane vrijednosti, tada će se ova greška vremenom dovoljno akumulirati preko integratora tako da će se pojaviti izlaz iz regulatora koji će djelovati na izvršni organ sistema sve dok ne dođe do anuliranja greške. Integralna komponenta je opasna za stabilnost sistema jer ona zakreće amplitudno-faznu karakteristiku sistema za $\pi/2$ u negativnom smjeru, tako da se može desiti da onaj dio karakteristike koji se nalazio u 1. i 2. kvadrantu i koji je najispupčeniji zakretanjem za $\pi/2$, može obuhvatiti kritičnu tačku. Iz ovog razloga treba biti oprezan prilikom izbora veličine integralne komponente. Treba napomenuti da velik broj procesa u prirodi ima integralni karakter, tako da kod ovakvih procesa ova komponenta i nije potrebna za sintezu algoritma upravljanja.

Diferencijalna komponenta je komponenta koja može poslužiti da stabilizira sistem prethodno odveden u nestabilnost sa integralnom komponentom, jer ona, suprotno njoj, vraća amplitudno-faznu karakteristiku sistema za $\pi/2$. Takođe, ova komponenta je idealna za brze procese kod kojih nastojimo da pratimo ulaz, jer ona po svojoj matematičkoj formi, formi diferencijala, predviđa ponašanje (trend) ulaza i daje adekvatan izlaz izvršnom organu sistema u cilju što bržeg formiranja adekvatnog odziva. Treba napomenuti da ova komponenta nije preporučljiva za sisteme kod kojih se pojavljuju šumovi na izlazu. Naime, bodeovi dijagrami diferencijalne komponente imaju slijedeći oblik:



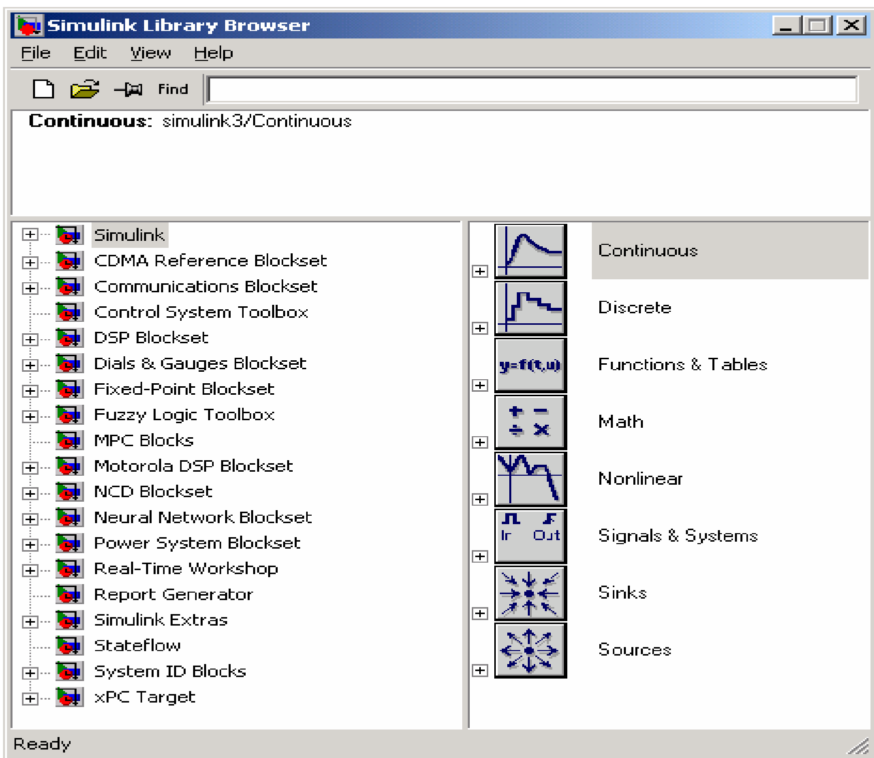
Slika 4.3

Vidimo da pri frekvencijama šumova, koji imaju karakter visokih frekvencija, imamo neželjena pojačanja sistema.

Pored *PID regulatora* postoje tzv. *adaptivni*, *robusni*, *fuzzy regulatori*, kao i regulatori sposobni da uče iz prethodnih iskustava – regulatori na bazi *neuronskih mreža*.

Cilj procedure automatizacije nekog procesa jeste zadovoljenje realnih zahtjeva onog ko je naručio automatizaciju. Ovi zahtjevi su najčešće izraženi kroz granicu *preskoka* odziva sistema i željeno *vrijeme smirivanja* prelaznih procesa u sistemu. Nažalost, ova dva zahtjeva su oprečne prirode, tako da se sva filozofija *PID regulatora* svodi na određivanje optimalnih parametara u cilju ispunjenja pomenutih zahtjeva.

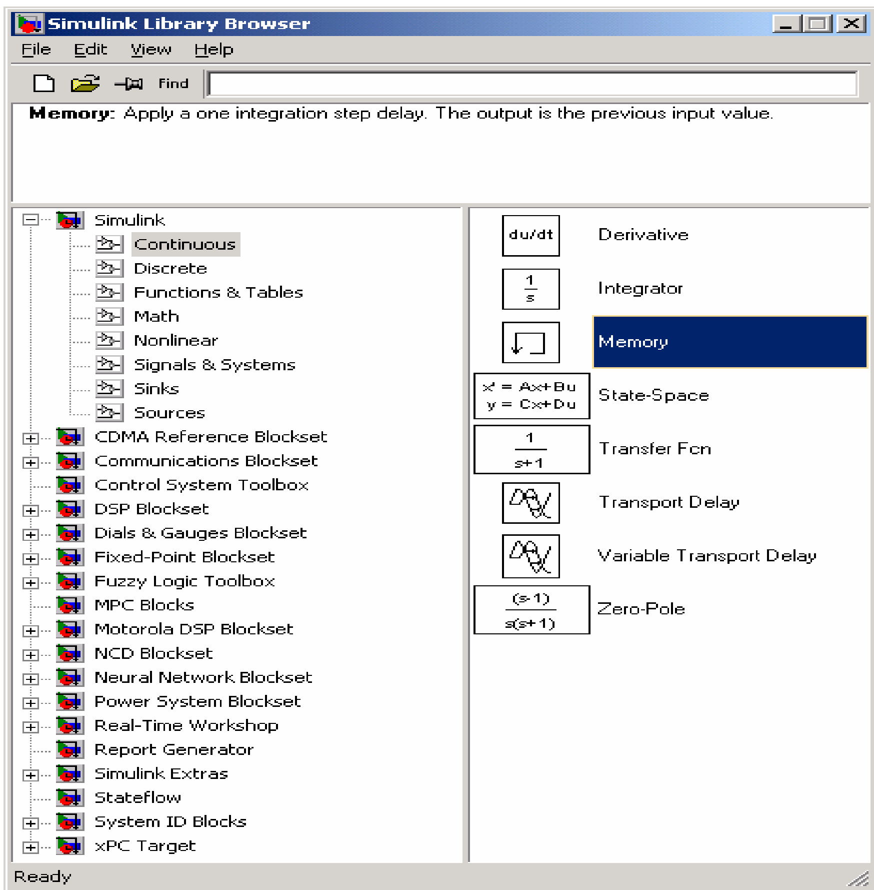
Da bi mogli uvidjeti uticaj parametara *PID* regulatora na odziv sistema, kao i na amplitudno-faznu karakteristiku, koristićemo MATLAB-ov **Simulink** paket.
Simulink predstavlja dio MATLAB-a za modeliranje i simulaciju dinamičkih sistema i posjeduje grafički *interface* (sučelje) koji omogućava modeliranje sistema dodavanjem već gotovih elementarnih blok dijagrama na ekran i dinamičko manipuliranje sa njima. Ovim programom možemo da simuliramo linearne, nelinearne, vremenski kontinulane, vremenski diskretne, kao i multivarijabline sisteme.
Ukucajmo sada u komandnom prozoru MATLAB-a naredbu **simulink** ili kliknimo mišom na **New Simulink Model** ikonicu MATLAB-ove palete. Dobijamo slijedeće:



Slika 4.4

U svakom od ovih blokova *Simulink*-ove biblioteke se nalaze odgovarajući elementarni blokovi.
Sources blok nam daje paletu najčešće korištenih funkcijskih izvora.
Sinks blok sadrži razne mogućnosti prikazivanja i čuvanja rezultata.

Discrete, **Continues** i **Nonlinear** blokovi, kako sama imena govore, sadrže elementarne blokove diskretnih, kontinualnih i nelinearnih sistema, respektivno.
Math blok nam daje spektar funkcija koje se često koriste u elementarnim proračunima.
I na kraju, paleta blokova s lijeve strane sadrži korespondirajuće blokove nekih od toolbox-ova MATLAB sistema.
Kako u našem predmetu pretežno operiramo sa vremenski kontinulanim i linearnim sistemima, slika elementarnih blokova koje koriste ti sistemi se dobija klikanjem na **Continues** blok i ista ima slijedeći oblik:



Slika 4.5

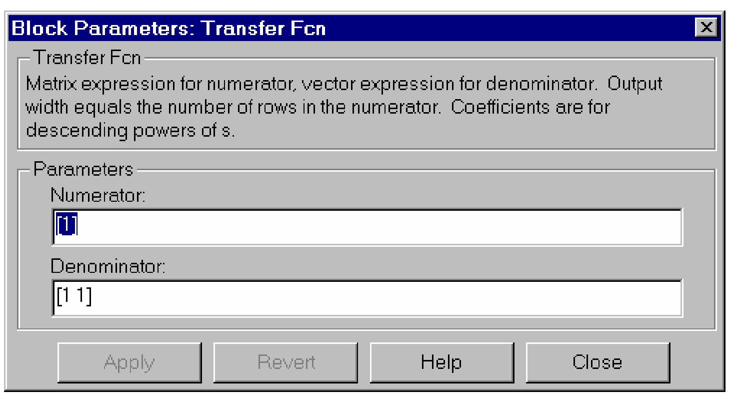
Pored **Continues** bloka, a radi prikazivanja rezultata i generiranja ulaza sistema, koristićemo **Sinks**, kao i **Sources** blok.

Primjer 1

Pronaći metodom isprobavanja parametre *PID regulatora* tako da čitav sistem sa negativnom povratnom spregom bude *stabilan*, ima *relativno brz odziv* i da mu *greška u stacionarnom stanju* bude anulirana. Za svaki set isprobanih parametara gledati odziv sistema i amplitudno-faznu karakteristiku otvorenog kruga.
Sistem je 3. reda i ima oblik:

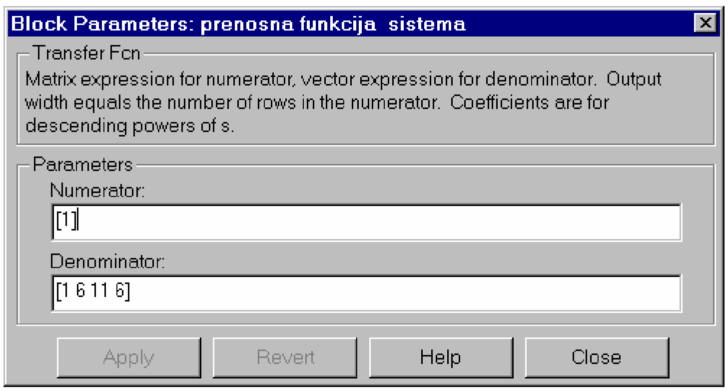
$$G(s) = \frac{1}{s^3 + 6s^2 + 11s + 6}$$

Napomena: Na predmetu '***Projektovanje sistema automatskog upravljanja***', koji studenti na odsjeku za automatiku i elektroniku slušaju na 4. godini studija, rade se konkretne metode pronalaženja optimalnih parametara *PID regulatora*. Ovaj primjer ima za cilj da student uvidi ulogu svakog od parametara pojedinačno u odzivu sistema i njihov uticaj na amplitudno-faznu karakteristiku otvorenog kruga, odnosno našeg procesa zajedno sa *PID regulatorom*.
Prilikom otvaranja *Simulink*-a automatski nam se otvara i prozor za simulaciju privremeno nazvan **untitled** u koji ćemo ubacivati blokove za simulaciju. U slučaju da se ovaj prozor ne pojavi, treba kliknuti na **New model** ikonicu u prozoru *Simulink*-a.
Otvorimo prvo **Continues** blok i iz njega mišom prenesimo **Transfer Fcn** blok u prozor za simulaciju. Kada smo to uradili, kliknimo dva puta na ovaj blok. Dobijamo slijedeću tzv. *parametarsku karticu* **Transfer Fcn** bloka:

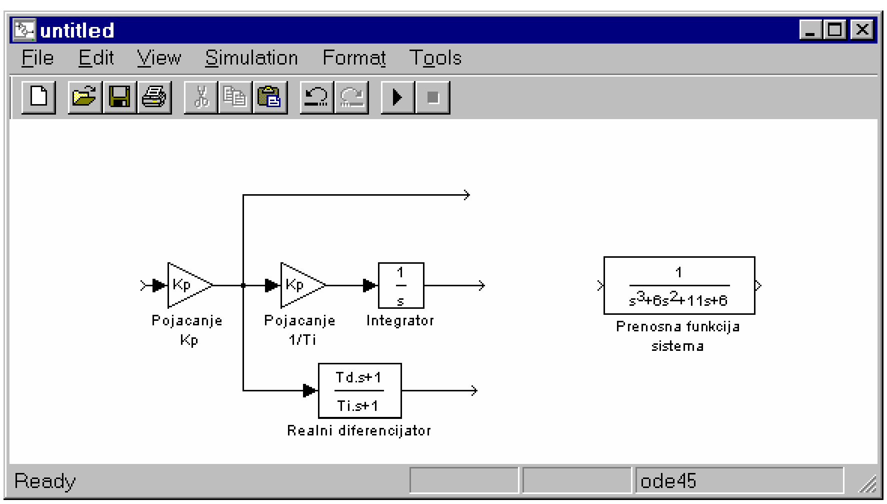


Slika 4.6

Sada unesimo našu prenosnu funkciju kao na *slici 4.7* i kliknimo na *Apply*, a zatim na *Close*.
Ovom procedurom smo kompletirali unošenje modela procesa u prozoru za simulaciju.
Sada trebamo dodati *PID regulator*.
Proporcionalnu, integralnu i realno-diferencijalnu komponentu dobijamo prenosom **Gain**, **Integrator** i **Transfer Fcn** blokova, respektivno, u prozor za simulaciju. Takođe, ispred integratora će nam trebati, zbog parametra T_i, još jedan **Gain** blok sa pojačanjem $K_i = 1/T_i$.

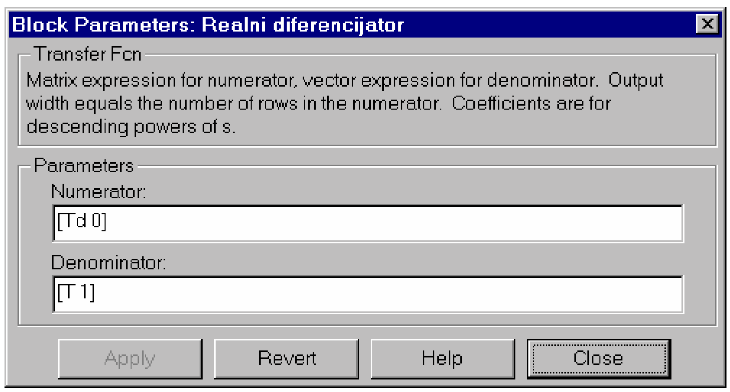


Slika 4.7



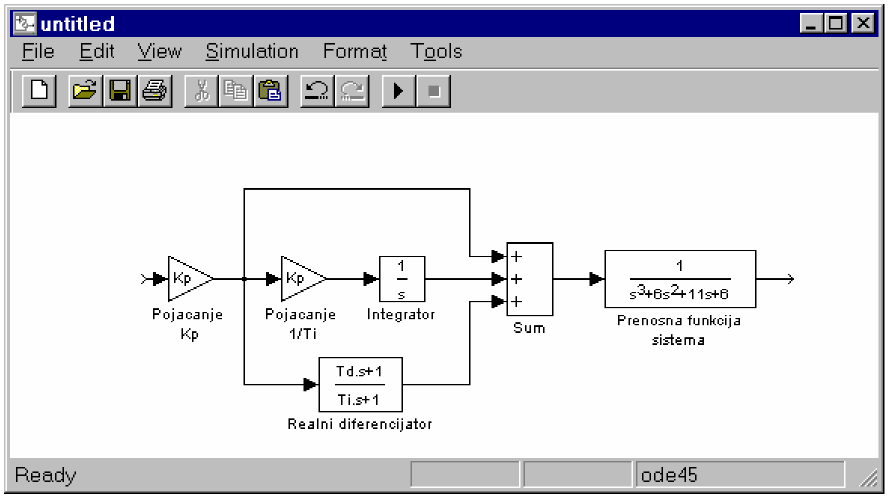
Slika 4.8

Poredajmo blokove kao na *slici 4.8*.
Preporučuje se promjena imena blokova radi kasnijeg lakšeg snalaženja u složenijim simulacijama. Promjenu imena blokova možemo realizovati nakon dvostrukog klikanja na staro ime bloka i unošenja novih imena (*Pojačanje* K_p, *Pojačanje* $1/T_i$, *Realni diferencijator* i *Prenosna funkcija sistema*). Takođe, na blok realnog diferencijatora treba kliknuti dva puta da bi promjenili parametre prenosne funkcije, kako bi dobili funkciju realnog diferencijatora. Za početak stavimo varijable T_d i T kao na *slici 4.9*.



Slika 4.9

Sada, pošto ova tri izlaza iz različitih komponenata *PID regulatora* treba sabrati u jedan, prenosimo blok **Sum,** koji se nalazi **Math** paleti blokova, u prozor za simulaciju. Kako je podrazumijevani broj ulaza ovog bloka 2, to ćemo upisati na njegovoj parametarskoj kartici tri plusa. Spojimo sada, korištenjem miša, izlaze sa ulazima kao na *slici 4.10*.

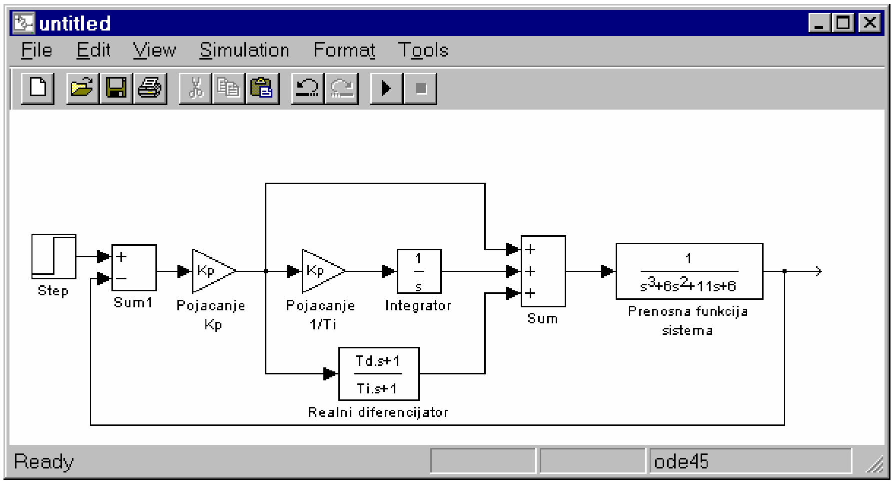


Slika 4.10

Da bi generirali jedan od ulaza, npr. step, trebamo iz **Sources** bloka prebaciti **Step** blok u prozor za simulaciju. I ovaj blok, naravno, ima svoju karticu sa parametrima, tako da možemo mijenjati veličinu amplitude, kao i vrijeme pojave step signala.

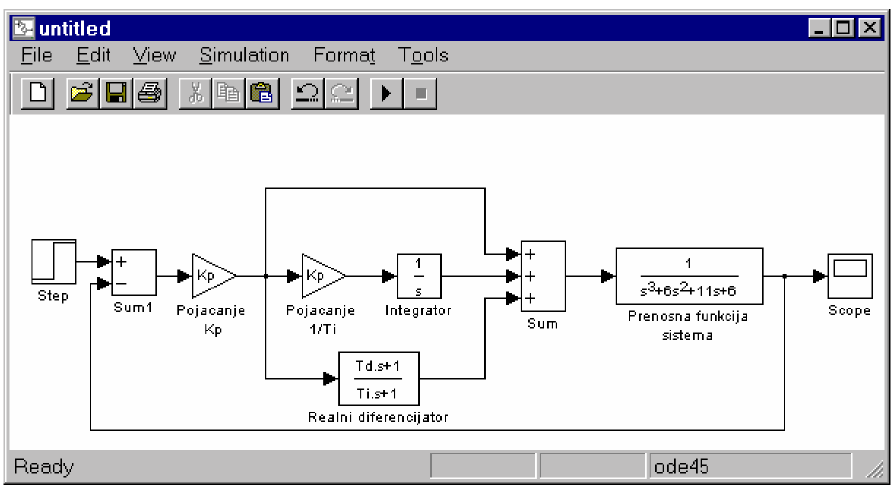
Dalje, da bi kreirali upravljanje sa negativnom povratnom spregom, trebamo dodati blok **Sum** i u parametarskoj kartici tog bloka obavezno staviti +- znak.

Slika sada ima izgled:



Slika 4.11

Konačno, radi prikazivanja odziva, iz **Sinks** bloka prebacimo **Scope** blok. Finalna simulacija treba izgledati ovako:



Slika 4.12

Vrijednosti komponenenata *PID regulatora* (K_p, T_i, T_d) izaberimo proizvoljno i stavimo da je $T = 0.001$.

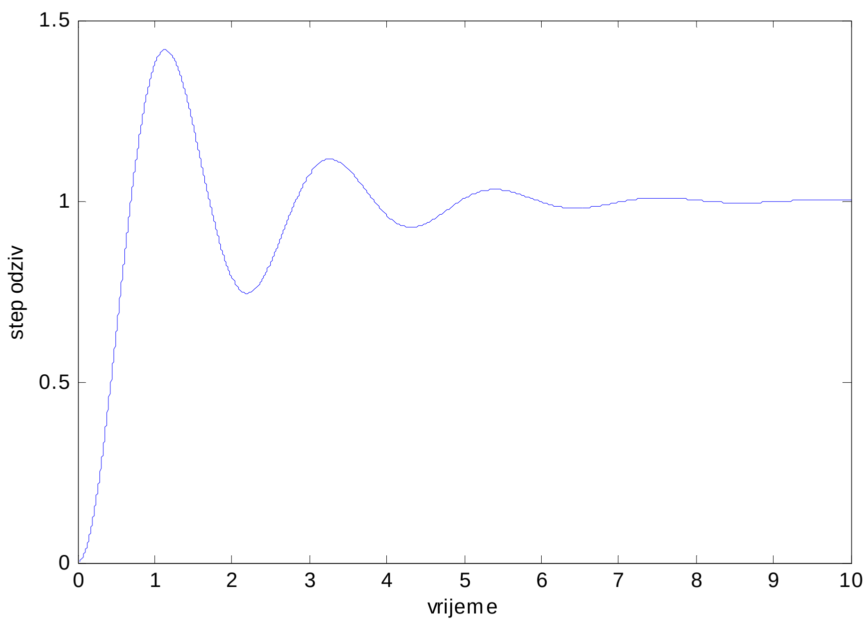
Pokretanje simulacije ide preko menija **Simulation→Start** ili klikom miša na predzadnju ikonicu palete ikonica prozora za simulaciju.

Ako uvidimo da prelazni proces nije mogao stati na grafik, koji dobijamo dvostrukim klikanjem na **Scope** blok nakon završene simulacije, tada treba kliknuti na **Simulation →Parameters** i povećati **Stop time**. Tok i završetak simulacije se indicira u donjem desnom uglu sa vremenom trajanja.

Najbolji način isprobavanja parametara regulatora je da isključimo integralnu i diferencijalnu komponentu, odnosno da pojačanje integralne komponente $K_i = 1/ T_i$ i parametar T_d diferencijalne komponente stavimo na 0, te da povećavamo proporcionalnu komponentu K_p. Ovom procedurom spoznajemo ulogu ovog parametra. Lahko ćemo uvidjeti da za dovoljno malene K_p greška stacionarnog stanja postaje uočljiva, dok za dovoljno velike K_p čitav sistem sa negativnom povratnom spregom postaje nestabilan.

Nakon ovoga, trebamo izabrati jedan K_p u intervalu ova dva limita, te mijenjati T_d i $1/T_i$ i uvidjeti njihov uticaj na odziv sistema.

Za vrijednosti parametara $K_p=20$, $1/T_i=1$, $T_d=0.3$, imamo slijedeći odziv:



Slika 4.13

Za svaku isprobanu kombinaciju parametara *PID regulatora* treba pogledati i amplitudno-faznu karakteristiku otvorenog kruga. Prenosna funkcija otovorenog kruga, na osnovu koje se dobija amplitudno-fazna karakteristika, ima oblik:

$$G(s) = Kp \cdot (1 + \frac{1}{T_i s} + \frac{T_d s}{Ts + 1}) \cdot \frac{1}{s^3 + 5s^2 + 11s + 6}$$

Najlakši način unošenja ove prenosne funkcije u MATLAB-ov radni prostor je razbijanje ove funkcije na više funkcija na slijedeći način:

$$G_1 = K_p$$

$$G_2 = 1$$

$$G_3 = \frac{1}{T_i s}$$

$$G_4 = \frac{T_d s}{Ts + 1}$$

$$G_5 = \frac{1}{s^3 + 5s^2 + 11s + 6}$$

Sekvence unošenja za, npr., gore navedene izabrane vrijednosti parametara *PID regulatora* će biti:

```
» G1=tf(20);
» G2=tf(1);
» G3=tf(1,[1 0]);
» G4=tf([0.3 0],[0.001 1]);
» G5=tf(1,[1 5 11 6]);
» G=G1*(G2+G3+G4)*G5;
» nyquist(G)
```

Napomena: Za svaku promjenu parametara regulatora treba mijenjati samo funkcije G_1, G_3 ili G_4. Radi jednostavnosti koristiti MATLAB-ov program **ltiview**.

Zadatak 1

Pronaći parametre P, PI i PID regulatora *Zigler-Nichols* metodom sistema sa prenosnom funkcijom:

$$G(s) = \frac{1}{(s+0.1)(s+0.3)(s+0.7)}$$

i uvidjeti razlike između ovih regulatora.

Zigler-Nichols metoda:

Ovom metodom prvo isključujemo rad integralne i diferencijalne komponente. Nakon toga povećavamo pojačanje K_p dok sistem ne počne da oscilira neprigušenim i nedivergentnim oscilacijama (granični slučaj između nestabilnosti i stabilnosti). Pojačanje regulatora pri kome se to dešava naziva se kritično pojačanje K_{pkrit}. Period dobijenih oscilacija označavamo sa T_{krit}. Optimalna podešenja parametara za *P*, *PI* i *PID regulator* u odnosu na pronađene vrijednosti kritičnog pojačanja i perioda oscilacija data su u tabeli.

Regulator	Optimalno podešenje		
	K_p	T_i	T_d
P	$0.55K_{pkrit}$	-	-
PI	$0.35K_{pkrit}$	$1.25T_{krit}$	-
PID	$0.6K_{pkrit}$	$0.8T_{krit}$	$0.2T_{krit}$

Primjer 2

Uvidjeti uticaj čistog vremenskog kašnjenja na amplitudno-faznu karakteristiku sistema datog slijedećom prenosnom funkcijom:

$$G(s) = \frac{1}{s^3 + 5s^2 + 11s + 6} \cdot e^{-\tau s}$$

MATLAB ne sadrži naredbu koja bi direktno unosila prenosnu funkciju koja posjeduje trasportno kašnjenje. Da bi ovo uradili moramo prvo unijeti dio prenosne funkcije bez kašnjenja. To radimo na standardan način:

```
» G=tf(1,[1 5 11 6])

Transfer function:
          1
----------------------
s^3 + 5 s^2 + 11 s + 6
```

Većinu elemenata (blokova, varijabli, grafika, itd.) MATLAB posmatra kao objekat iza koga stoje atributi koji u potpunosti određuju objekat. Da bi došli do liste ovih atributa, u komandnom prozoru MATLAB-a treba ukucati naredbu **get**(*handle*), gdje je *handle* ili puni naziv objekta ili interni broj koji MATLAB pripiše svakom korištenom objektu. U našem slučaju imamo:

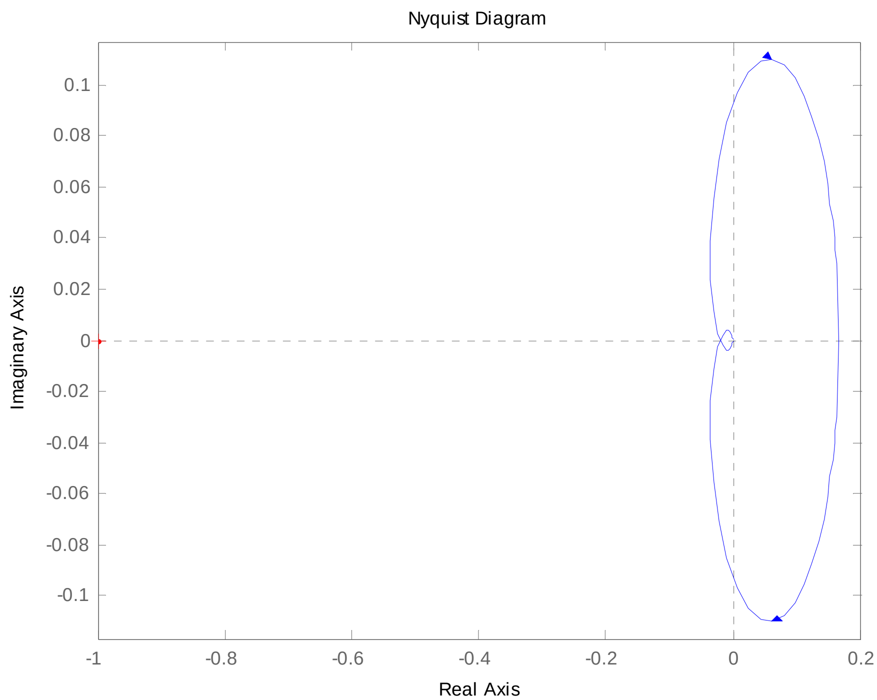
```
>> get(g)
          num: {[0 0 0 1]}
          den: {[1 5 11 6]}
         Variable: 's'
          Ts: 0
          ioDelay: 0
          InputDelay: 0
         OutputDelay: 0
          InputName: {''}
         OutputName: {''}
          InputGroup: {0x2 cell}
         OutputGroup: {0x2 cell}
         Notes: {}
         UserData: []
```

Ovo predstavlja listu atributa objekta prenosne funkcije nad kojom možemo da vršimo manipulaciju. Nama je interesantan atribut *OutputDelay* koji predstavlja vrijeme kašnjenja signala.

Mijenjanje parametara objekta se obavlja naredbom **set**(*handle*, ’ime_parametra’, nova_vrijednost_parametra). Uzećemo nekoliko vrijednosti parametara T_d: 0.1; 0.5; 1; 2 i uvidjeti uticaj kašnjenja na amplitudno-faznu karakteristiku.

Prvo nacrtajmo amplitudno-faznu karakteristiku funkcije bez kašnjenja.

```
» nyquist(G)
```



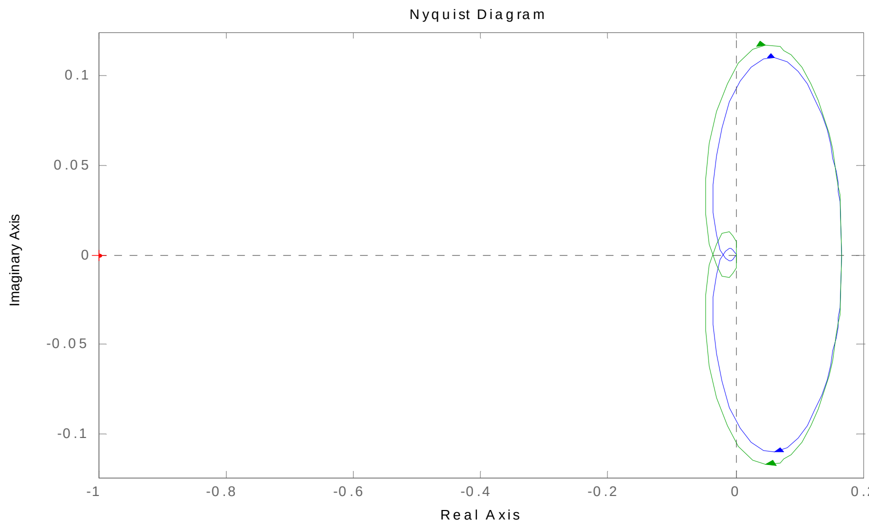
Slika 4.14

Da bi mogli crtati jedan grafik preko drugog grafika koristimo naredbu **hold on** koja čuva prethodno nacrtani grafik.

```
» hold on;
```

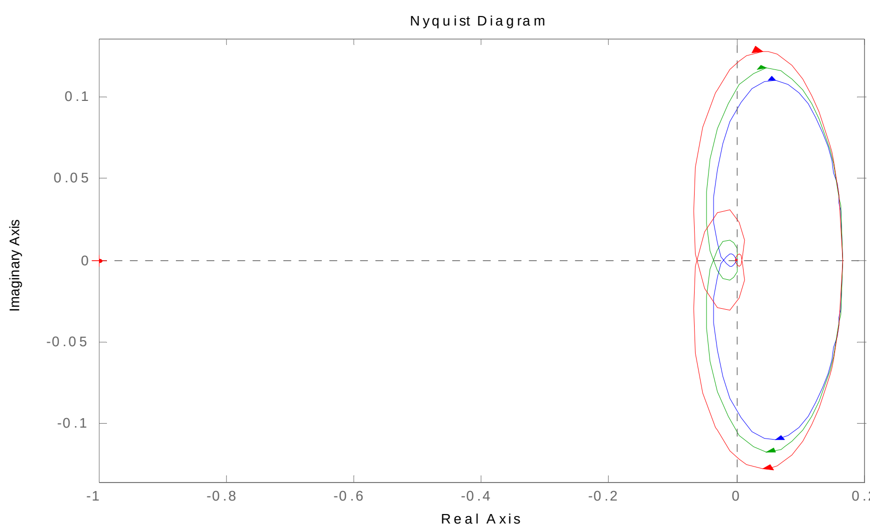
Sada ubacujemo vremensko kašnjenje i odmah crtamo amplitudno-faznu karakteristiku:

```
» set(G,'OutputDelay',0.1);
» nyquist(G)
```

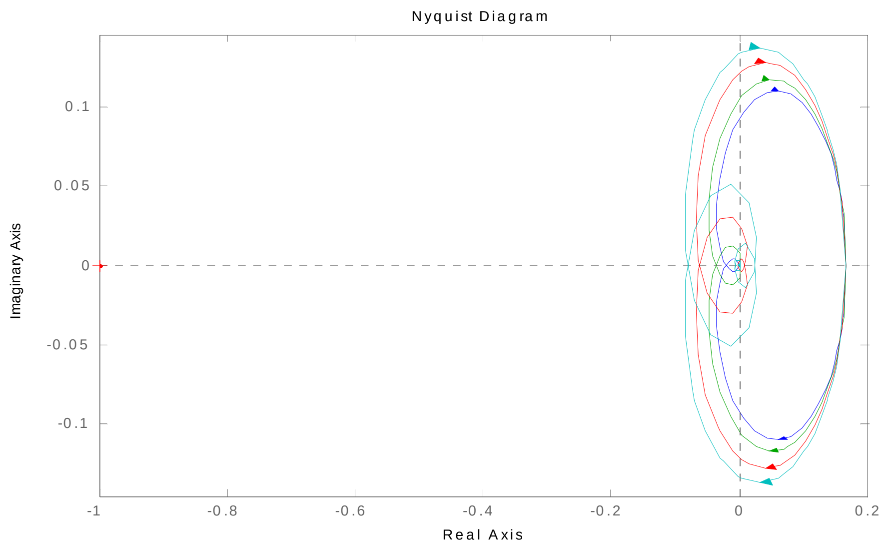


Slika 4.15

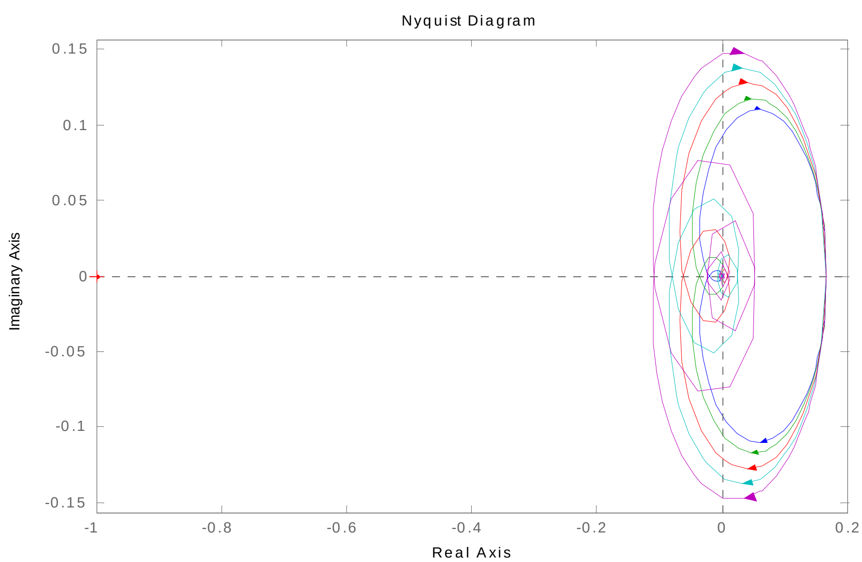
Za ostale date vrijednosti čistog kašnjenja imamo slijedeće korespodente dijagrame:



Slika 4.16



Slika 4.17



Slika 4.18

Nije teško uočiti neželjeno ekspandiranje amplitudno-fazne karakteristike sa povećanjem čistog transportnog kašnjenja.

Zadatak 2

Naći približnu vrijednost vremenskog kašnjenja τ za koju sistem sa jediničnom povratnom spregom i dat sa prenosnom funkcijom:

$$G(s) = \frac{2}{s+1} \cdot e^{-\tau s}$$

odlazi u nestabilnost.
Napomena: Zadatak raditi u kompleksnom domenu.

...

Zadatak 3

Za pronađene optimalne parametre PID regulatora sistema iz **zadatka 1** ispitati ponašanje odziva datog sistema sa trasportnim kašnjenjem od 0.5s.
Za ovakav sistem ponovo naći optimalne parametre PID regulatora **Zigler-Nichols** metodom.

...

Zadatak 4

Za proces čija je dinamika opisana prenosnom funkcijom:

$$G(s) = \frac{3}{(s+1)^3} \cdot e^{-\tau s}$$

odrediti vrijednost transportnog kašnjenja τ tako da sistem bude na granici stabilnosti.
Napomena: Zadatak raditi u vremenskom domenu, a rezultat provjeriti u kompleksnom domenu.

...

Zadatak 5

Neka je dat sistem opisan prenosnom funkcijom:

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

Na ulaz sistema dovesti nesinusoidalne periodične funkcije različitih frekvencija, te na osnovu izlaza iz sistema zaključiti koju vrstu filtera predstavlja dati sistem.
Dobijeni zaključak uporediti sa zaključkom dobijenim iz dijagrama amplitudno-frekventne karakteristike.

...