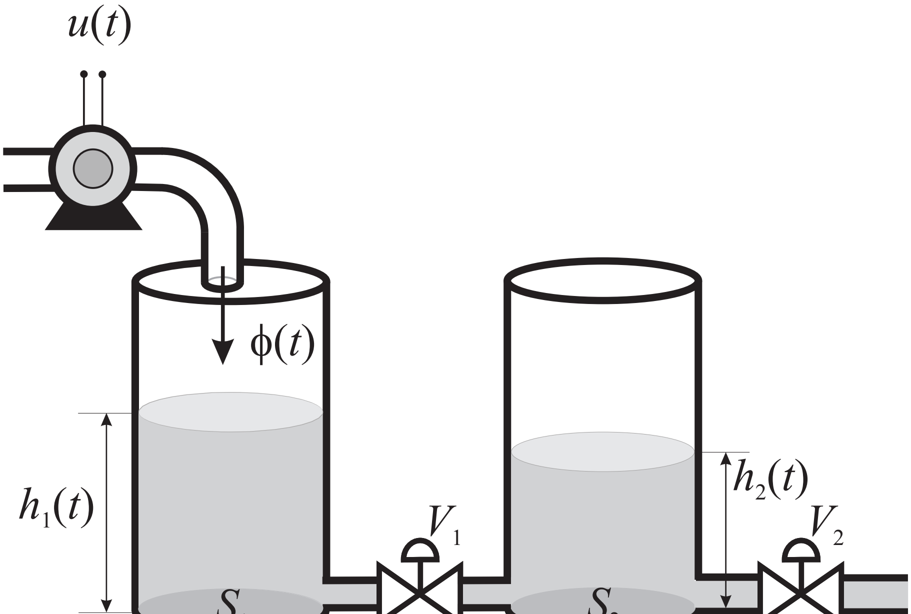
Laboratorijske vaje
Računalniška simulacija
2012/13

1. laboratorijska vaja

Simulacija dinamičnih sistemov s pomočjo osnovnih funkcij orodij MATLAB in Simulink

Pri tej laboratorijski vaji boste spoznali osnovne možnosti za simulacijo dinamičnih sistemov v orodjih MATLAB in Simulink. Obravnavani sistem dveh shranjevalnikov je prikazan na sliki 1.



Slika 1: Shematski prikaz obravnavanega sistema dveh shranjevalnikov

Spremenljivke v sistemu so:

- napetost na črpalki $u(t)$,
- vhodni masni pretok $\phi(t)$,
- masni pretok $\phi_1(t)$,

- masni pretok $\phi_2(t)$,
- višina vode v prvem shranjevalniku $h_1(t)$ in
- višina vode v drugem shranjevalniku $h_2(t)$.

Prečna preseka obeh cilindričnih shranjevalnikov sta S_1 in S_2. Do matematičnega modela sistema pridemo z upoštevanjem masnega ravnovesja v obeh shranjevalnikih:

$$\begin{aligned} \rho S_1 \dot{h}_1 &= \phi - \phi_1 \\ \rho S_2 \dot{h}_2 &= \phi_1 - \phi_2 \end{aligned} \tag{1}$$

pri čemer je ρ gostota kapljevine. Pretok skozi ventil je odvisen od tlačne razlike na obeh straneh ventila. Če predpostavimo, da je pretok skozi ventil V_1 premo sorazmeren razliki višin h_1 in h_2, pretok skozi ventil V_2 pa premo sorazmeren višini h_2, dobimo naslednji model sistema:

$$\begin{aligned} C_{H1} \dot{h}_1 &= \phi - \tfrac{h_1 - h_2}{R_{H1}} \\ C_{H2} \dot{h}_2 &= \tfrac{h_1 - h_2}{R_{H1}} - \tfrac{h_2}{R_{H2}} \end{aligned} \tag{2}$$

pri čemer sta R_{H1} in R_{H2} hidravlični upornosti ventilov V_1 in V_2, po analogiji z električnimi sistemi pa smo uvedli tudi hidravlični kapacitivnosti $C_{H1} = \rho S_1$ in $C_{H2} = \rho S_2$.

Če izvedemo Laplaceovo transformacijo enačb (2), pridemo do prenosnih funkcij med vhodnim pretokom in obema višinama:

$$\begin{aligned} G_1(s) = \frac{h_1(s)}{\phi(s)} &= \frac{sR_{H1}R_{H2}C_{H2} + R_{H1} + R_{H2}}{s^2 R_{H1}R_{H2}C_{H1}C_{H2} + s(R_{H1}C_{H1} + R_{H2}C_{H1} + R_{H2}C_{H2}) + 1} \\ G_2(s) = \frac{h_2(s)}{\phi(s)} &= \frac{R_{H2}}{s^2 R_{H1}R_{H2}C_{H1}C_{H2} + s(R_{H1}C_{H1} + R_{H2}C_{H1} + R_{H2}C_{H2}) + 1} \end{aligned} \tag{3}$$

Številčne vrednosti konstant so:

- $\rho = 1000$ kg/m^3,
- $S_1 = S_2 = 0{,}01$ m^2,
- $R_{H1} = 5$ ms/kg in
- $R_{H2} = 10$ ms/kg.

Splošni napotki

Delo bo potekalo znotraj programskega paketa MATLAB, delno pa v njegovem orodju za simulacijo Simulink. Kratek opis funkcij, ki jih boste uporabljali, najdete v učbeniku Računalniška simulacija (avtor prof. dr. Borut Zupančič), ki je prosto dostopen tudi na spletu (poglavji 5.4 Simulacijsko okolje Matlab-Simulink in 5.5 Simulacija s pomočjo Matlabovih funkcij). Posamezne funkcije, ki jih boste uporabljali, so večinoma eksplicitno navedene, pomoč pa lahko dobite tudi z uporabo funkcijskih klicev `help` in `doc` in navedbo imena želene funkcije.

Svetujem, da na začetku vseh programov zbrišete prostor spremenljivk in slike:

```
clear all
close all
```

Poleg tega svetujem, da pri klicu funkcij uporabite procesne parametre (hidravlične upornosti, gostoto ipd.) kot spremenljivke, ki pa jih je prej seveda potrebno definirati. S tem je koda veliko bolj pregledna, hkrati pa je lažje prilagodljiva, ker boste tudi pri prihodnjih vajah delali na podobnih sistemih.

Naloge

Vsako od spodaj naštetih nalogo rešite in rezultate predstavite na posebnem grafu (novo okno z grafom odprete z ukazom `figure`):

1. Najprej boste simulirali sistem v programu MATLAB. Realizirajte prenosni funkciji $G_1(s)$ in $G_2(s)$ kot objekta tipa `tf` (klic funkcije `tf`). Nato izvedite simulacijo obeh sistemov na stopničasto spremembo vhodnega pretoka, pri čemer stopnica ni enotina, pač pa ima amplitudo 0,02 kg/s. Za simulacijo uporabite funkcijo `step`, pri čemer je potrebno rezultat funkcije pomnožiti z ustrezno konstanto (0,02). Simulacijo izvedite dvakrat z različnima časoma vzorčenja: rezultat prve simulacije s časom vzorčenja 10 s prikažite z dvema izvlečenima krivuljama (ena prikazuje $h_1(t)$, druga pa $h_2(t)$), rezultat simulacije s časom vzorčenja 100 s (za $h_1(t)$ in $h_2(t)$) pa prikažite s križci. Ali križci ležijo na krivulji? Kaj lahko iz tega sklepamo?

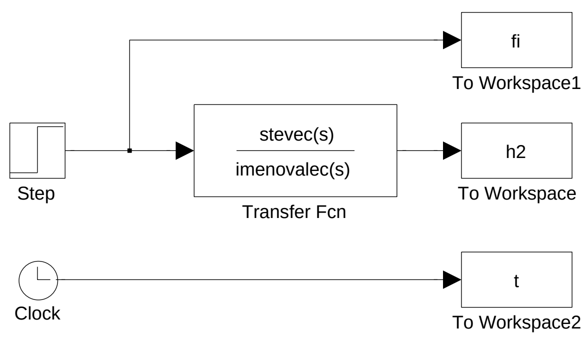
2. Sistem $G_2(s)$ simulirajte še z uporabo funkcije `lsim`, ki omogoča simulacijo na vzbujalni signal poljubne oblike. Vhodni signal naj bo spet stopničasti signal z amplitudo 0,02 kg/s. Znova izvedite dve simulaciji s korakoma 10 s in 100 s ter rezultate prikažite na podoben način ko pri prejšnji nalogi. Ali je rezultat simulacije enak kot pri prejšnji nalogi, če ste v časovnem vektorju uporabili iste vrednosti?

3. Sistem $G_2(s)$ simulirajte še z uporabo funkcije `lsim`, pri čemer naj bo vhodni signal

$$\phi(t) = \phi_0 + \phi_a \sin(\omega t)$$

kjer so vrednosti konstant naslednje: $\phi_0 = 0{,}02$ kg/s, $\phi_a = 0{,}01$ kg/s in $\omega = 0{,}01$ rad/s. Simulacijo izvedite dvakrat z različnima časoma vzorčenja: rezultat prve simulacije s časom vzorčenja 10 s prikažite z izvlečeno krivuljo, rezultat simulacije s časom vzorčenja 100 s pa prikažite s križci. Ali križci ležijo na krivulji? Kaj lahko iz tega sklepamo?

4. Sistem, ki ga opisuje prenosna funkcija $G_2(s)$, realizirajte še v orodju Simulink. Uporabite blok za predstavitev prenosne funkcije (`Transfer Fcn`), pri čemer naj parametri bloka ne vsebujejo številskih vrednosti konstant procesa, pač pa ustrezne spremenljivke, katerih vrednosti podajte v delovnem prostoru Matlaba. Simulacijska shema naj bo podobna kot tista na sliki 2. Simulacijo lahko poženete tudi iz Matlaba ali vaše skripte z ukazom `sim('ime_sheme')`. Ker so v simulacijski shemi uporabljeni bloki `To Workspace`, se rezultati simulacije prenesejo v ustrezne matrike v delovnem prostoru Matlaba, kjer jih lahko naprej obdelujete (format shranjevanja podatkov naj bo `Array`). Sistem naj vzbuja stopničasti vhodni signal – stopnica naj nastopi ob času $t = 0$, njena amplituda pa naj bo 0,02 kg/s. Najprej izvedite simulacijo s privzetimi vrednostmi simulacijskih parametrov in odziv prikažite na grafu z izvlečeno krivuljo. Meni `Simulation | Configuration parameters` vsebuje parametre, s katerimi lahko vplivamo na potek simulacije. V našem primeru bomo spremenili vrsto uporabljene integracijske metode. Naslednja dva simulacijska teka boste tako izvedli z Eulerjevo integracijsko metodo (koraka naj bosta 10 s in 100 s) in odziva prikazali na istem grafu kot rezultat prejšnje simulacije (križci – korak 10 s in krožci – korak 100 s). Ali križci in krožci ležijo na krivulji? Kaj lahko iz tega sklepamo?



Slika 2: Simulacijska shema pri 4. nalogi

5. Sedaj bomo predpostavili, da imata ventila korenski karakteristiki, torej velja:

$$\begin{aligned} \phi_1 &= k_{V1}\,\mathrm{sign}(h_1 - h_2)\sqrt{|h_1 - h_2|} \\ \phi_2 &= k_{V2}\,\mathrm{sign}(h_2)\sqrt{|h_2|} \end{aligned} \tag{4}$$

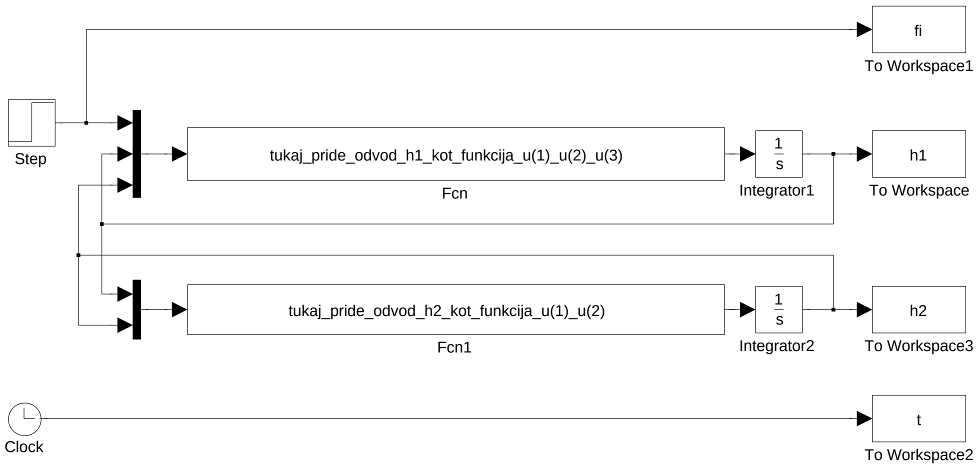
kjer sta vrednosti konstant obeh ventilov enaki $k_{V1} = 0{,}1$ kgm^{-1}s$^{-1/2}$ in $k_{V2} = 0{,}05$ kgm^{-1}s$^{-1/2}$. Funkcija sign$(\cdot)$ vrne 1, če je operand pozitiven, 0, če je operand enak 0, in -1, če je operand negativen. Funkcijo v Matlabu realiziramo s funkcijskim klicem `sign`, korensko funkcijo s `sqrt`, izračun absolutne vrednosti pa z `abs`. V praksi sicer ni mogoče, da bi postala višina h_2 negativna, v simulaciji pa je to zaradi numeričnih napak mogoče, zato v drugi enačbi v (4) dopuščamo, da je višina h_2 negativna.
V Matlabu boste simulirali ta nelinearni sistem na način, ki je opisan v poglavju 5.5.3 (Simulacija s pomočjo vgrajenih funkcij za numerično integracijo). Uporabili boste solver ode45 (klic funkcije `ode45`). Ključen je prvi argument te funkcije, ki podaja ročico (angl. *handle*) na funkcijo (imenujmo jo `odvoda`), ki na osnovi trenutnih stanj (v našem primeru sta to obe višini) izračuna njuna odvoda. Funkcija `odvoda` torej izračuna $\dot{h}_1$ in $\dot{h}_2$ z upoštevanjem enačb (1) in (4). V funkciji `odvoda` je potrebno definirati nekaj konstant ter na osnovi vhodnega pretoka ϕ in obeh stanj h_1 in h_2 izračunati vrednosti obeh odvodov $\dot{h}_1$ in $\dot{h}_2$. Seveda moramo poznati obliko vhodnega signala $\phi(t)$. V našem primeru bomo privzeli, da je vzbujanje spet stopničasto z amplitudo 0,02 kg/s, kar pomeni, da ga lahko v funkciji `odvoda` obravnavamo kot konstanto (simulacija namreč teče od časa $t = 0$ naprej). Spodaj je v pomoč podan del funkcije `odvoda`, ki jo shranite v datoteko `odvoda.m`:

```
function dx=odvoda(t,x)

h1 = x(1);
h2 = x(2);


...

dx=[dh1 dh2]';
```

Oba odziva (h_1 in h_2) prikažite na istem grafu.

6. Enak sistem kot v prejšnji točki boste realizirali tudi v Simulinku, pri čemer boste izračun $\dot{h}_1$ in $\dot{h}_2$ opravili z uporabo bloka `Fcn`. Oba odvoda boste integrirali in dobili višini vode. Tudi v tem primeru je ključen način izračuna $\dot{h}_1$ in $\dot{h}_2$. Že prej smo ugotovili, da je $\dot{h}_1$ odvisen od spremenljivk ϕ, h_1 in h_2 ter nekaj konstant. Zato je potrebno za izračun $\dot{h}_1$ v blok `Fcn` pripeljati omenjene tri signale. Ker ima blok `Fcn` le en vhod, si pomagajte z multiplekserjem (blok `Mux`), s katerim omenjene tri signale združite in jih pripeljete na vhod bloka `Fcn`. V simulacijski shemi v Simulinku torej na opisani način izvedite simulacijo obeh višin vode, kjer spet uporabite vzbujalni signal iz prejšnje naloge

(blok `Step`). Simulacijska shema bo podobna kot tista na sliki 3. Odziva (h_1 in h_2) spet prikažite na istem grafu. POMEMBNO: Funkcija sign($\cdot$) se znotraj bloka `Fcn` kliče kot `sgn` in ne `sign`!



Slika 3: Simulacijska shema pri 6. nalogi