

Harjoitus 7: Dynaamisten systeemien säätö (Simulink)

SCI-C0200 Fysiikan ja matematiikan menetelmien studio 2023



Harjoituksen aiheita

- Dynaamisten (=ajassa kehittyvien) systeemien mallintaminen ja säätö
 - Takaisinkytkentä, avoin vs. suljettu ohjaus, PID-säädin
- Dynaamisten systeemien mallintaminen Simulinkilla

Oppimistavoitteet

- Osaat luoda Simulinkillä yksinkertaisen dynaamisen systeemin mallin, tuottaa numeerisia ratkaisuja sen avulla sekä ohjata systeemin toimintaa PID-säätimellä
- Ymmärrät avoimen ja suljetun silmukan ohjauksen eron

Dynaamiset systeemit

Dynaamisia, eli ajassa kehittyviä, systeemejä kaikkialla

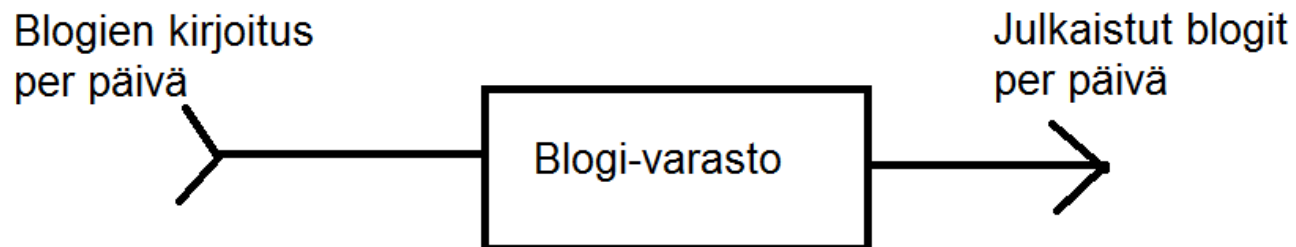
Takaisinkytkentä: Systeemin osa X vaikuttaa ympäristöönsä, joka vaikuttaa takaisin osaan X

Joitain dynaamisia ilmiöitä:

- Oskillaatiot. Esim. sydämessä, taloussyklit, jouset,...
- Kertyminen (säiliöt). Esim. raha pankkitilille, CO₂ ilmakehään,...

Esimerkki:

- Läpivirtaus-säiliöt stabiloivat dynaamisia prosesseja



Dynaamisten systeemien mallintaminen

- Simulointimalli:
Systeemin osien kuvaus muuttujilla.
Muuttujien keskenäiset riippuvuudet.
Mallia (tai malleja) voidaan ajaa erilaisilla asetuksilla ja vertailla tuloksia.
- Malli rakennetaan sopivaksi suhteessa sen käytötarkoitukseen
Esim. ennustaminen, optimointi, säätö, ongelmakohtien tunnistaminen, maailmankuvan rakentaminen
- Kokeilujen tekeminen voi olla kallista tai vaarallista
Esim. lentosimulaatiot, sotasimulaatiot, suuret teolliset prosessit
- Esimerkiksi dynaaminen systeemi tuotantoketjusta. Kysymys: Missä pullonkaulat?

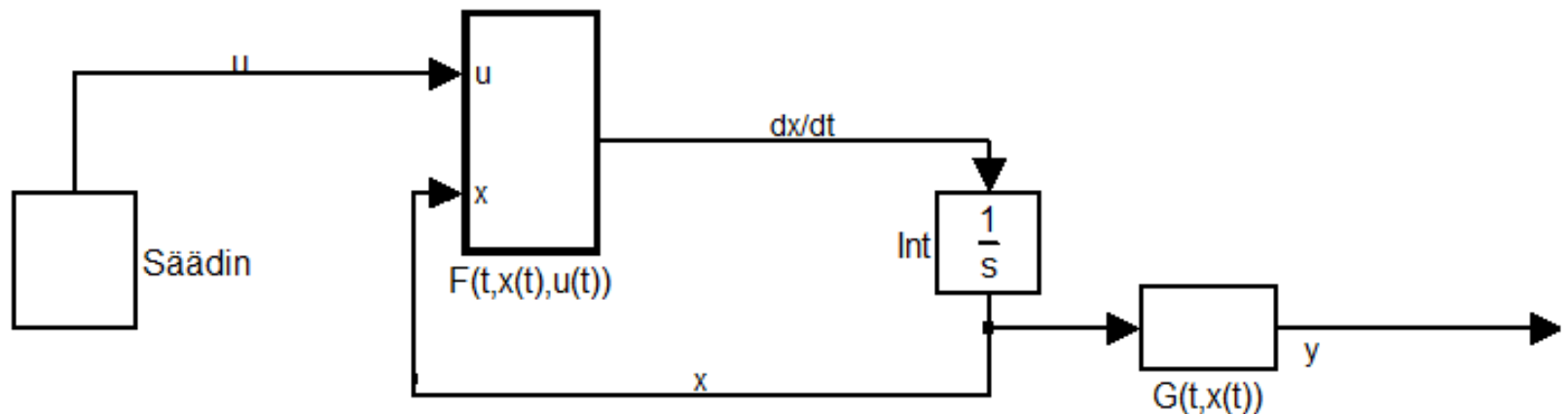
Dynaamisen systeemin säätö

- Dynaamista systeemiä usein halutaan ohjata toimimaan tietyllä tavalla
- Sovelluksia:
 - Teollisten prosessien automatisaatio: Esimerkiksi kaapelin valmistus hyödyntäen (dynaamista) kemiallista prosessia
 - Polttoaineen käytön säätely voimalaitoksessa
 - Robottiikka
- **Stabilointiongelma**: ulostulon oltava vakio systeemiin vaikuttavista häiriöstä riippumatta, esim. säiliön pinnankorkeuden säätö, vaihtovirtageneraattorin kierrosluku.
- **Servo-ongelma**: ulostulon seurattava referenssisignaalia mahdollisimman tarkasti, esim. teollisuusrobotin asennonsäätö.

Dynaamisen systeemin säätö

$$\frac{d}{dt}x(t) = F(t, x(t), u(t)) \quad F: \text{Malli systeemin riippuvuussuhteista}$$

$$y(t) = G(t, x(t)) \quad G: \text{ulostulon riippuvuus tilasta.}$$



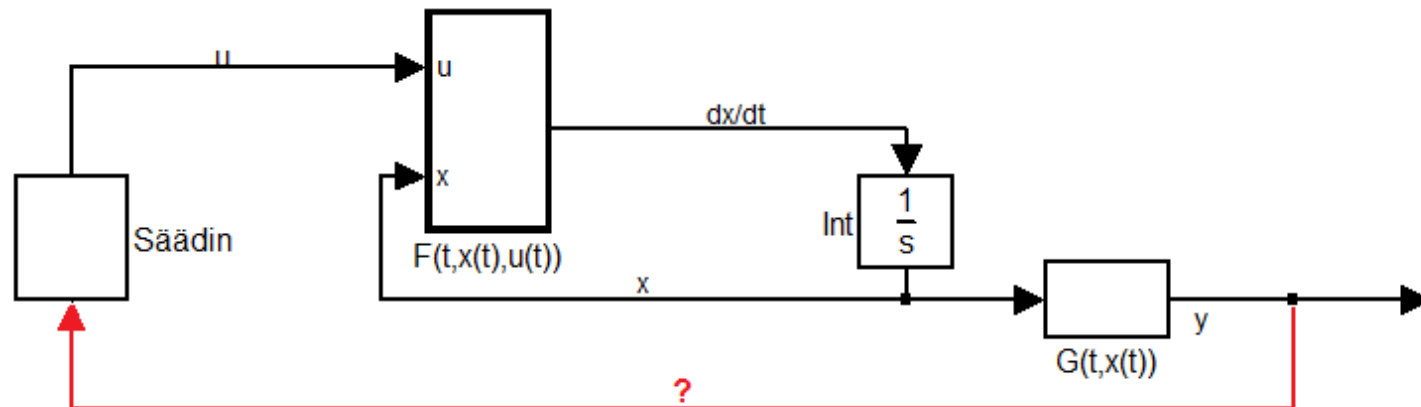
$x(t)$: Systeemin **tila**. Ei voida määritellä tai havainnoida suoraan.

$y(t)$: Systeemin **ulostulo**. Tämän voimme havaita ja mitata.

$u(t)$: Systeemin **ohjaus**. Tähän voimme vaikuttaa.

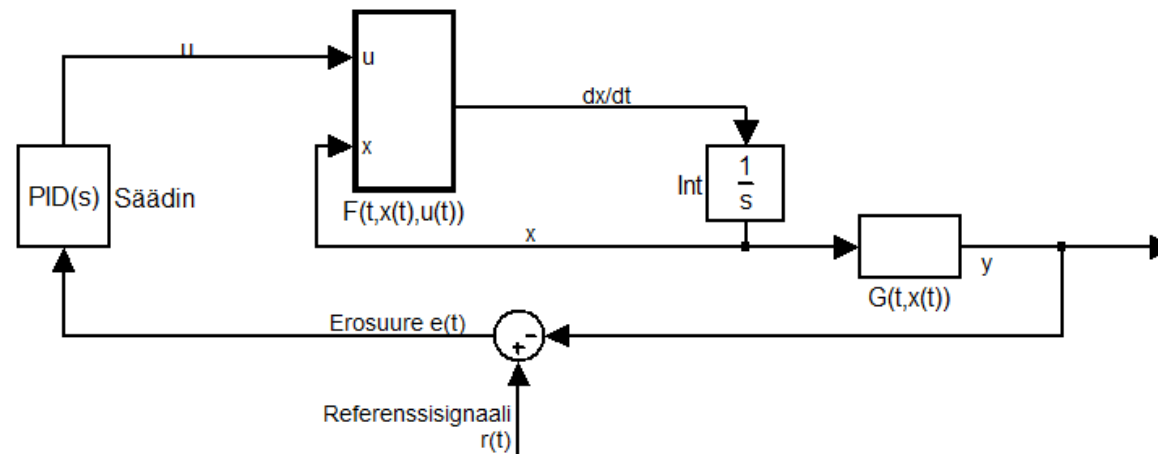
Avoim vs. takaisinkytketty ohjaus

- **Avoim silmukka** (open loop): ei takaisinkytkentää.
 - Ohjaus muodostetaan ilman ulostuloa (Esim. autolla ajo pelkän kellon avulla)
 - Häiriöitä (esim. muu liikennettä) ei voida kompensoida.
- **Suljettu silmukka** (closed loop): takaisinkytkentä.
 - Hyödynnetään ulostuloa ohjauksen muodostamisessa (esim. autolla ajaminen näköhavaintojen perusteella).



Säätöongelmia

- **Stabilointiongelma:** ulostulon oltava vakio systeemiin vaikuttavista häiriöstä riippumatta, esim. säiliön pinnankorkeuden säätö, vaihtovirtageneraattorin kierrosluku.
- **Servo-ongelma:** ulostulon seurattava referenssisignaalia mahdollisimman tarkasti, esim. teollisuusrobotin asennonsäätö.



- Valitse säädin (esim. PID, tilatakaisinkytkentä) ja parametrit s.e. Jos $e(t) = 0 \Rightarrow$ OK, muuten korjaa $u(t)$:tä kunnes $e(t) = 0$.

PID-säädin 1/3

Yksinkertainen laajasti käytetty säätötekniikka. Käytännön toteutus esim. pneumaattisesti (paineeseen perustuen) tai elektronisesti.

- P=proportionaalinen, I=integroiva, D=derivoiva
- P-säädin: $u(t) = K_P e(t)$, missä K_P on säätimen parametri.
 - Suhteellinen takaisinkytkentä
 - Yksinkertaisin mahdollinen säädin.
- Ongelma: P-säädin ei osaa kompensoida askelmaista häiriötä.
 - Syntyy pysyvä poikkeama ulostuloon
- Idea: kasvatetaan ohjausta kunnes $e(t) = 0$.
 - Asetetaan $u(t)$ riippumaan $e(t)$:n integraalista.
- PI-säädin: $u(t) = K_P e(t) + K_I \int_0^t e(\tau) d\tau$

PID-säädin 2/3

- Kun K_I kasvaa, vaste (eli ulostulo) nopeutuu MUTTA:
 - Suljetun silmukan systeemi muuttuu eräällä parametriyhdistelmällä epästabiiliksi.
 - Syy: luotetaan liian vanhaan informaatioon (integrointi).
- (Eräs) ratkaisu: derioiva takaisinkytkentä; perustetaan $u(t)$ $e(t)$:n derivaatalle (vrt. ennustaminen).
- PID-säädin:

$$u(t) = K_P e(t) + K_I \int_0^t e(\tau) d\tau + K_D \frac{d}{dt} e(t)$$

PID-säädin 3/3

- Tehtävä: Valitse sopivat parametrit K_P , K_I , K_D .
 - K_P suuri \Rightarrow nopea vaste, mutta epästabiilisuus vaanii.
 - K_I suuri \Rightarrow nopea vaste, pysyvät poikkeamat kompensoituvat (epästabiilisuus!)
 - K_D : käyttö esim. stabilointi (ongelma: kohina)
- PID-säädin on yleisesti käytetty säädintyyppi teollisuudessa.
- PID-säätimen virittämiseen eli parametrien määrittämiseen on olemassa erilaisia tekniikoita.

Matlab Simulinkin käyttö

- **Graafinen** ympäristö dynaamisten systeemien mallintamiseen
- Laajasti käytetty työkalu

Käyttäjiä: Puhelinvalmistajat, laivateollisuus, autoteollisuus, energiateollisuus,...

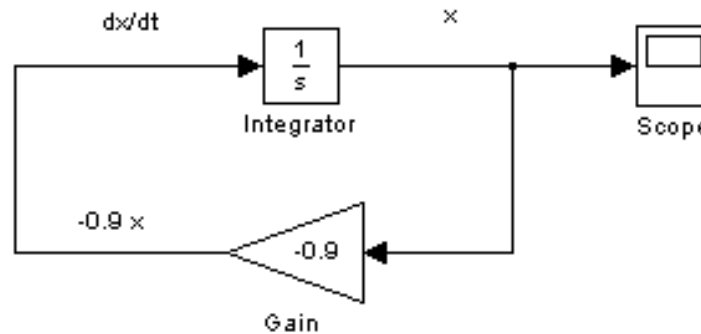
- Monipuolinen työkalu: Kukaan ei osaa täydellisesti!
- Samat mallit voidaan ainakin periaatteessa tehdä millä tahansa työkalulla (Matlab, C, kynä ja paperi)
- Simulinkin graafisen käyttöliittymän edut parhaimmillaan:
Mallinnus nopeutuu, koodi bugivapaata, mallin rakenteen esittely muille helpompaa
- Käyttö aluksi haastavaa: Tänään harjoitellaan perusjuttuja.
Tavoitteena jonkinlainen muistikuva ja kokemus Simulinkistä!

Lisää säätöteoriaa, simulinkia ja dynamiikkaa

- Säätöteoriaa käsitellään kursseilla
 - MS-E2148 Dynamic Optimization (optimiohjaus/-säätö)
 - MS-E2129 Systemien identifiointi
 - MS-E2133 Systems Analysis Laboratory II
- Näistä kahdella jälkimmäisellä käytetään Simulinkia.
- Dynaamisia systeemejä käsitellään osin myös kursseilla
 - MS-C2111 Stokastiset prosessit (satunnaisesti ajassa kehittyvät prosessit)
 - MS-E2170 Simulation (stokastisia prosesseja ja systeemidynaamisia simulaatioita riippuen vuodesta)
 - MS-E2114 Investment Science (mm. osakekurssien kehitys)
 - MS-C2128 Ennustaminen ja aikasarja-analyysi (aikasarjat voivat syntyä tietyn dynaamisen prosessin kautta)

Matlab Simulink

- Systemi kuvataan lohkoina (blocks) ja niitä yhdistävinä nuolina
- Helpottaa monimutkaisten systeemien mallintamisessa
- Esim. dif.yhtälö $\frac{d}{dt}x(t) = -0.9x(t)$ simulinkissä:

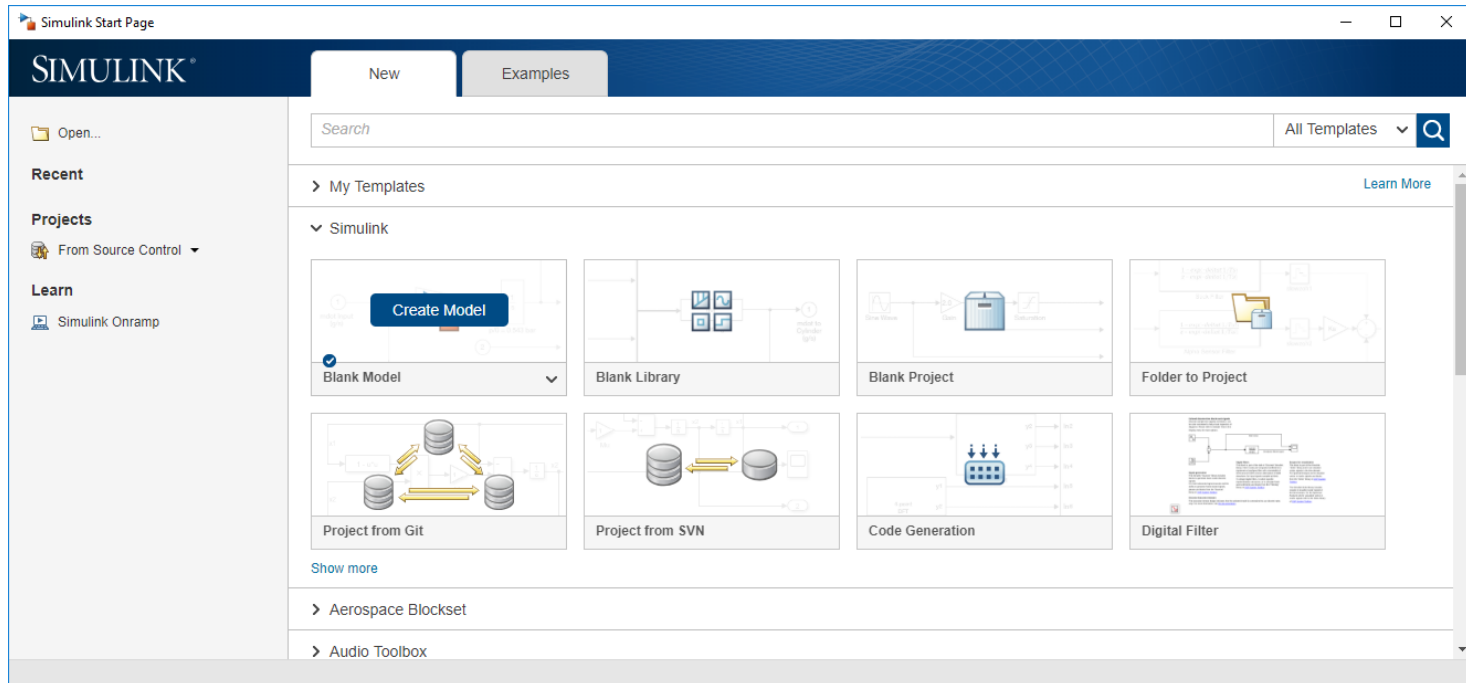


(integraattoriin sisään $\frac{d}{dt}x(t) \Rightarrow$ ulos tulee $\int[\frac{d}{dt}x(t)]dt = x(t)$)

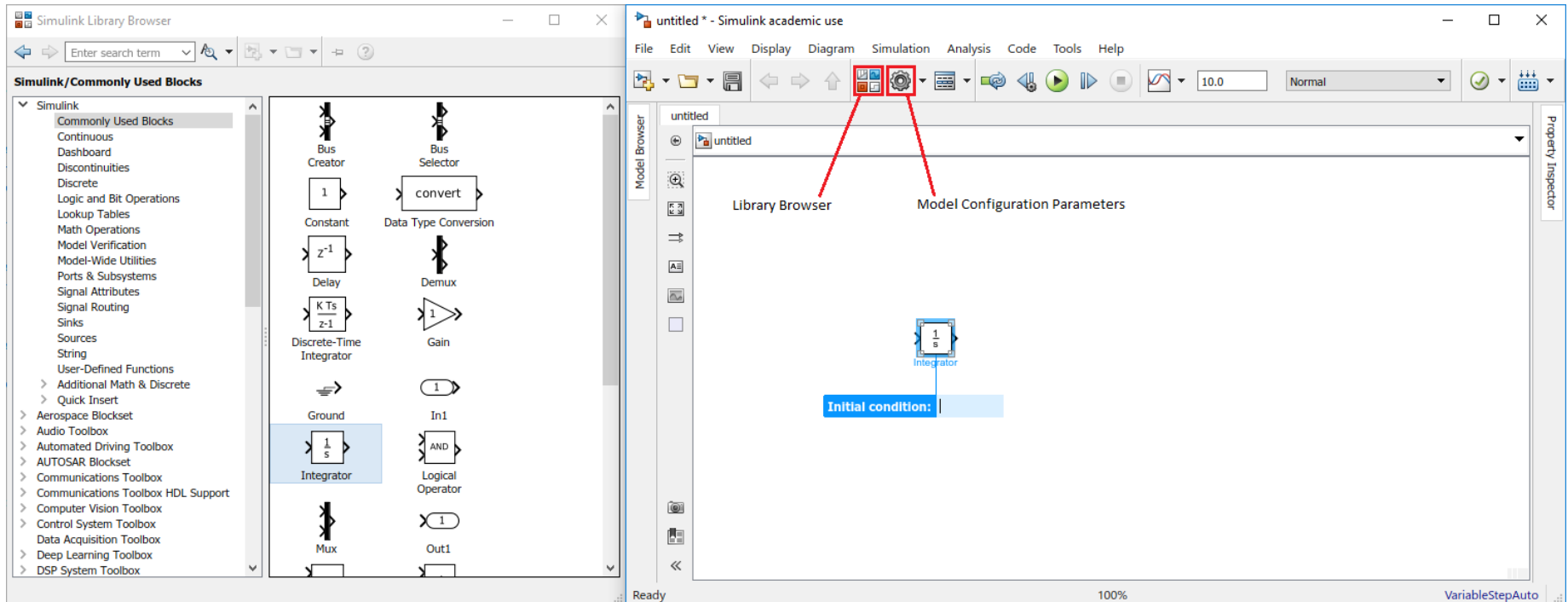
- vrt. dif.yhtälön koodaaminen minunSystemi.m-tiedostoon ja ode45-komennon käyttö

Simulinkin käynnistäminen

- Simulinkin voi käynnistää Matlabista komennolla `simulink` tai vaihtoehtoisesti yläpalkissa olevan ikonin kautta
- Uusi malli (tyhjä ikkuna) saadaan luotua klikkaamalla avautuvasta **Simulink Start Page** ikkunasta kohtaa **Blank Model**



Simulinkin käyttöliittymä

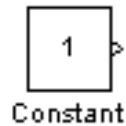


- Lohkoja lisätään vetämällä ne Library Browser-ikkunasta malliin
- Toinen tapa lisätä lohkoja on klikata mallia, aloittaa kirjoittaa lohkon nimeä ja valita oikea lohko ilmestyvästä listasta

Sources ja Sinks - lohkot

- Simulink-mallin signaalit lähtevät liikkeelle sources-lohkoista.

Vakiosignaali



Constant

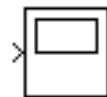
Normaalijakautunut satunnaisluku



Random Number

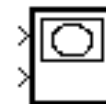
- Huom! Jos käytetään [takaisinkytkentää](#), ulkoista signaalilähdettä ei välttämättä tarvita (vrt. $\frac{d}{dt}x(t) = -0.9x(t)$ Simulinkissä).
- Simulointitulokset voidaan ohjata Sinks-lohkoihin tarkastelua varten.
Esim:

Signaali ajan funktiona



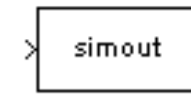
Scope

Kahden signaalin piirtäminen vaihetasossa



XY Graph

Tulosten tallentaminen Matlabin työtilaan



To Workspace

Lohkojen käsittelyä

- Lohko valitaan klikkaamalla sitä hiirellä. Useampi lohko voidaan valita painamalla Shift-nappia tai hiirellä maalaamalla.
- Tuplaklikkaamalla lohkoa päästään muuttamaan lohkon [parametreja](#).
- Lohko voidaan [monistaa](#) vetämällä hiiren oikealla napilla (tai vasemmalla napilla Ctrl-nappi painettuna).
 - Monistetuilla lohkoilla on samat parametrien arvot kuin alkuperäisillä lohkoilla.
- Lohkon suuntaa voi muuttaa valikosta [Format](#) → [Flip Block](#) ja [Format](#) → [Rotate Block](#).
- Lohkojen kokoa voidaan muuttaa lohkon kulmista vetämällä.

Lohkojen yhdistäminen

- Kaksi lohkoa voidaan yhdistää kahdella tavalla:
 - Valitaan ensin yksi lohko (lohko, josta signaali lähtee) ja klikkaamalla toista lohkoa Ctrl-nappi painettuna.
 - Vedetään nuoli käsin lohkon reunasta toiseen lohkoon asti.
- Nuolet lohkojen välillä voidaan jakaa useammaksi haaraksi vetämällä nuolesta hiiren oikea näppäin painettuna.
 - Sama signaali molemmissa haarassa.
- Lohkoja ja nuolia voi poistaa [delete](#)-näppäimellä.

Simuloinnin ajaminen

- Simulointi voidaan joko ajaa Simulinkista käsin:



(Yllä olevassa kuvassa 10.0 on simulointiaika)

- TAI Matlabin komentojonotiedostosta tai komentoriviltä komennolla **sim**. Komentojonotiedostoja kannattaa käyttää simulaatioiden ajoon, sillä Simulink -malli voi käyttää työtilaan tallennettuja muuttujia.

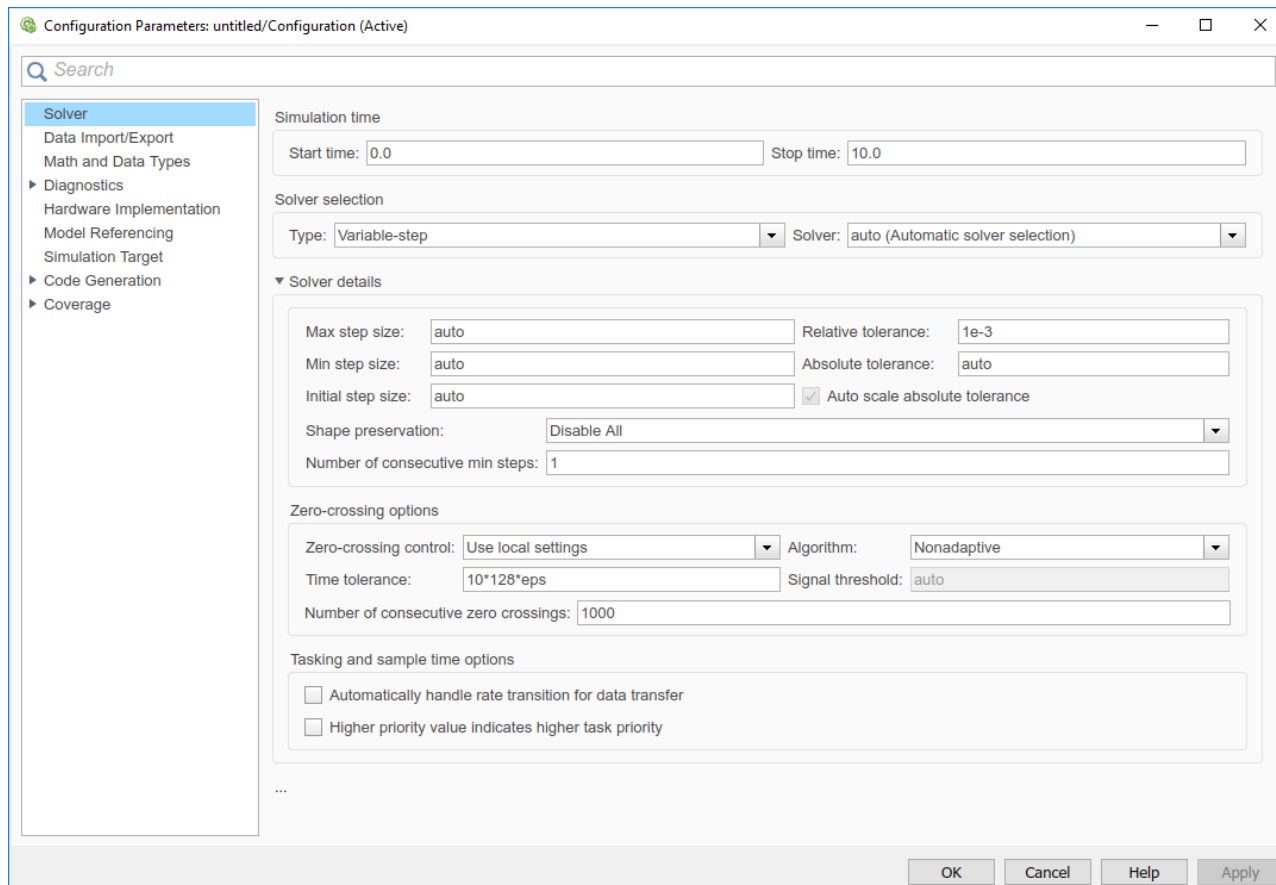
```
>> help sim
```

```
SIM Simulate a Simulink model
```

```
SIM('model') will simulate your Simulink model using all  
simulation parameter dialog settings including Workspace  
I/O options.
```

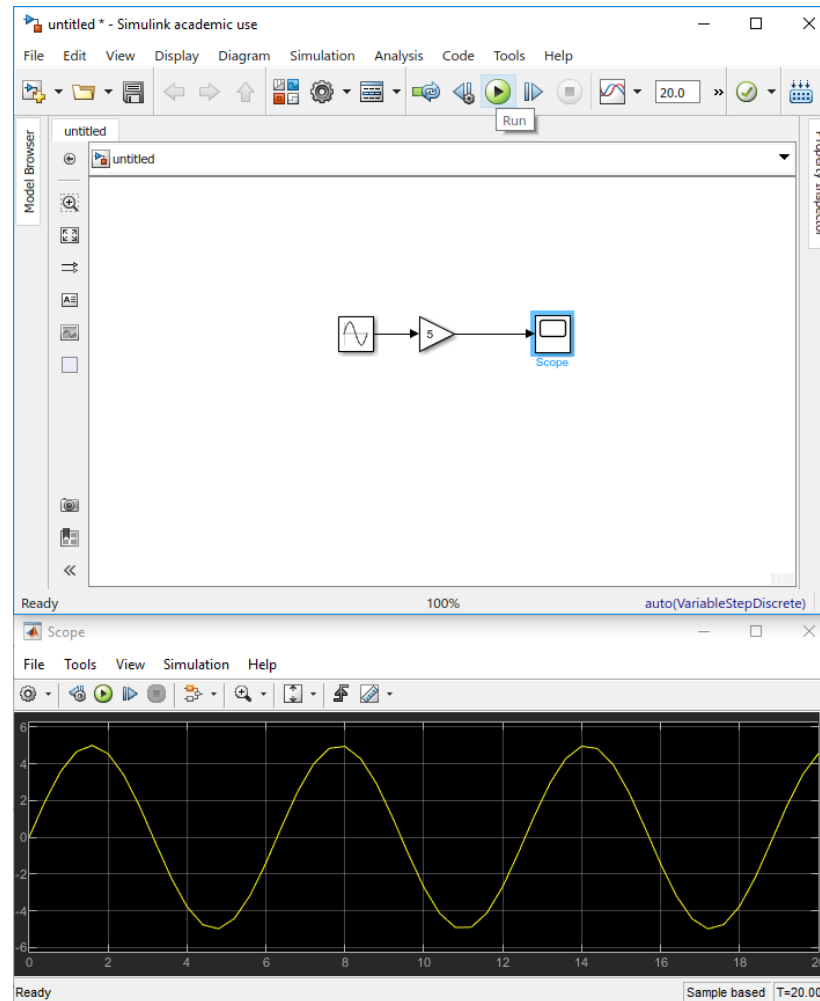
Simuloinnin asetukset

- Simuloinnin asetuksia voidaan muuttaa valikosta **Simulation** → **Model Configuration Parameters...**



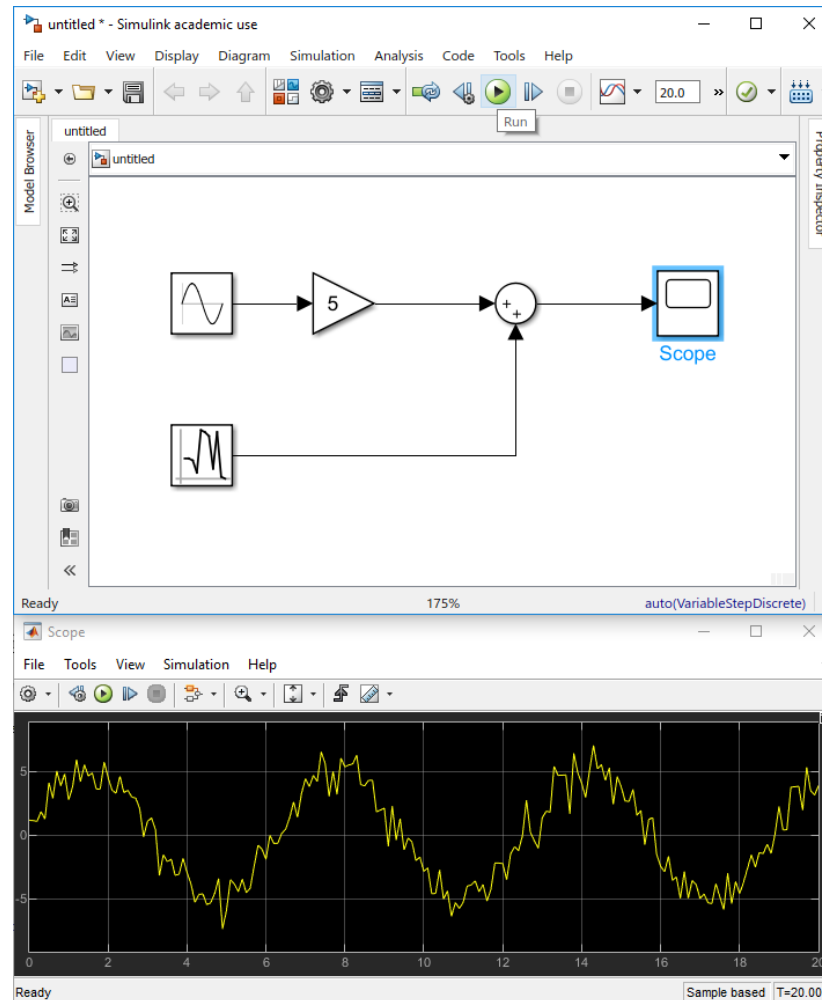
Simulink-mallin rakentaminen - esimerkki 1

- Muodostetaan siniaalto, jonka amplitudi on 5. (Amplitudin voi muuttaa myös siniaallon parametreista.)



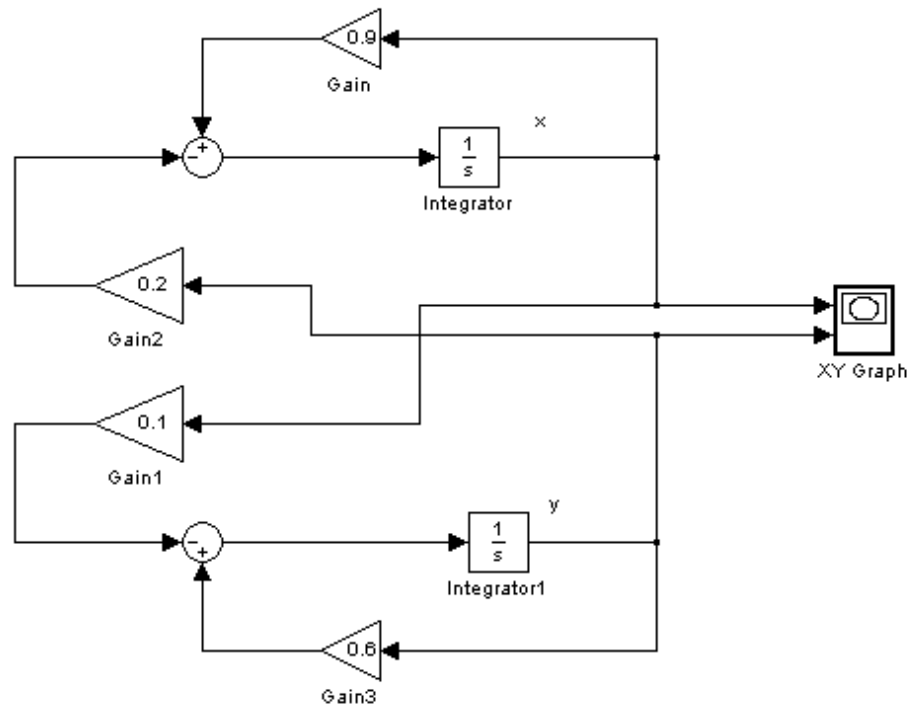
Simulink-mallin rakentaminen - esimerkki 2

- Lisätään malliin mukaan häiriösignaali.



Integrator-lohkon käyttö

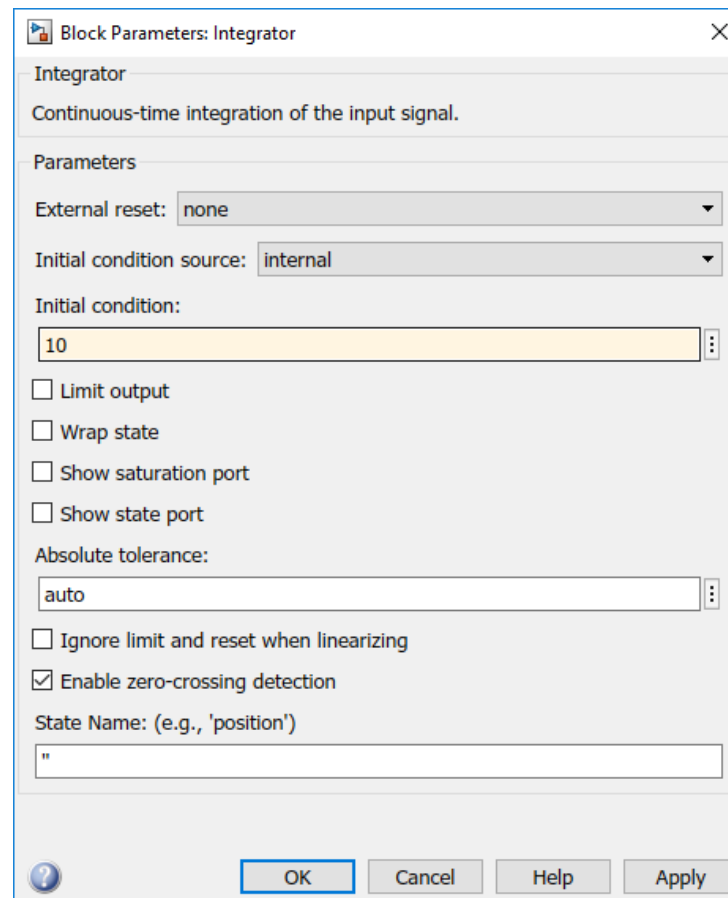
- Esim. diff. yhtälö systeemi $\frac{d}{dt}x(t) = 0.9x(t) - 0.2y(t)$,
 $\frac{d}{dt}y(t) = 0.6y(t) - 0.1x(t)$ Simulinkissä:



- Tarkistus: Esim. alemman integraattoriin sisäänmeno $-0.1x(t) + 0.6y(t)$ ja ulostulo on $y(t)$, OK!

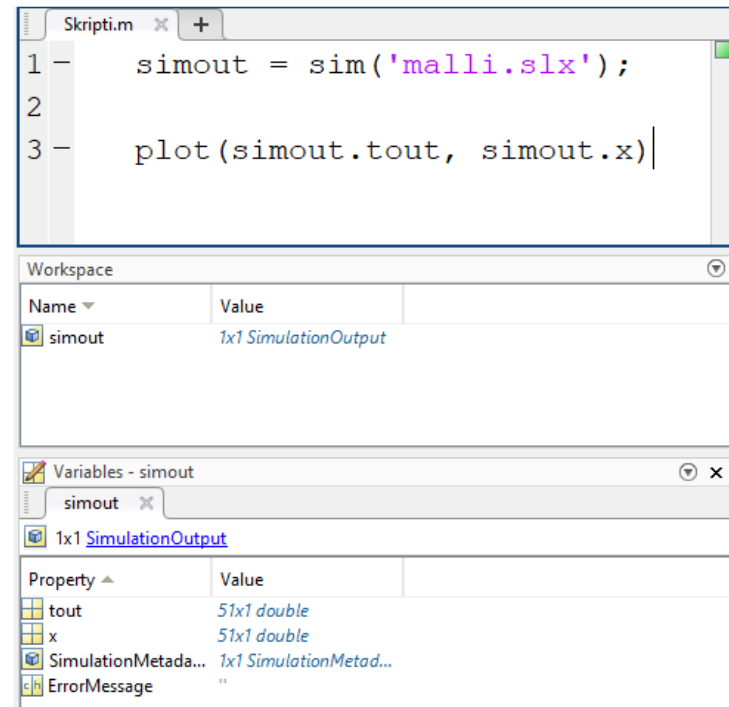
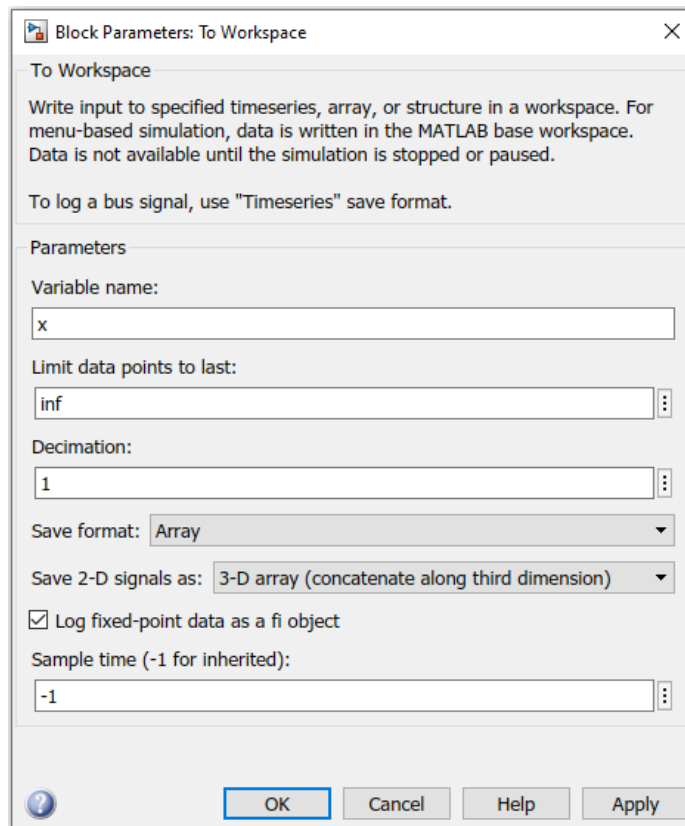
Integrator-lohkon käyttö

- Esim. alkuarvo $x(0) = 10$ asetetaan sen integrator-lohkon parametrejä muuttamalla (tuplaklikkaamalla integrator-lohkoa), jonka sisäänmeno on $\frac{d}{dt}x(t)$ ja ulostulo $x(t)$.



To workspace -lohkon käyttö

- Muuttujien arvoja voi tallentaa Matlabin työtilaan to workspace -lohkon avulla. Lohkon parametrejä muuttamalla voi valita muuttujan nimen ja tallennusformaatin. Kuvassa muuttujan nimeksi on valittu x ja tallennusformaatiksi array (matriisi).

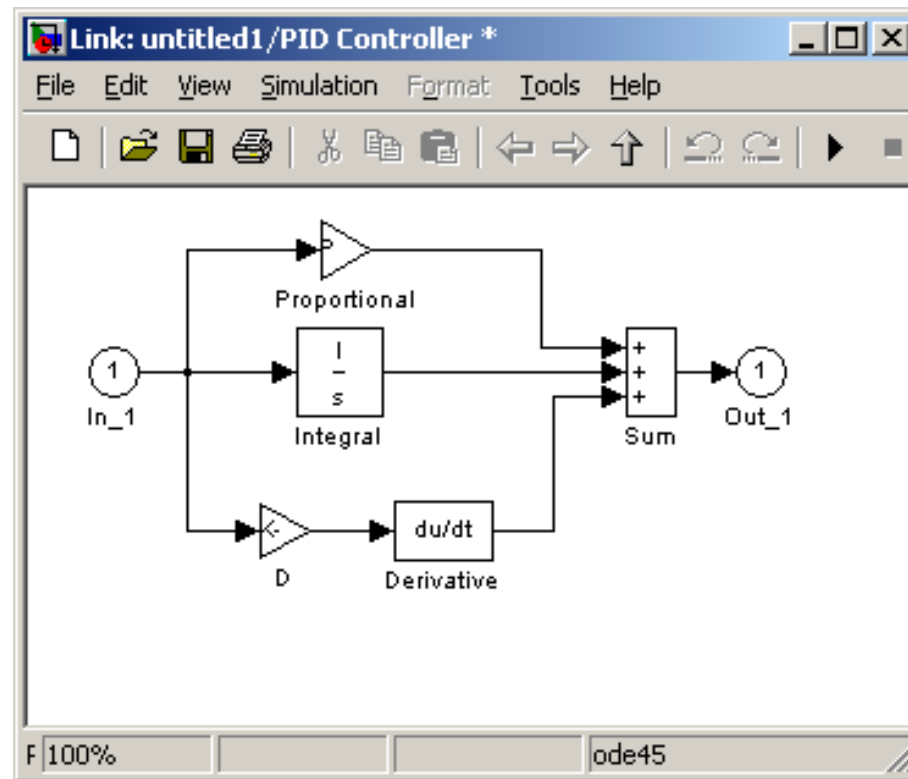


Joitakin Simulink ongelmatilanteita

- Simulointitulokset eivät ole oikean kaltaisia: tarkista Integrator-lohkojen alkuarvot (initial condition).
- Simulointitulokset eivät näy XY Graphissa: [akselit](#) on asetettava käsin tuplaklikkaamalla XY Graphia.
- Plottaaminen Matlabin puolella ei onnistu 'To Workspace'-lohkon kanssa: vaihda 'To Workspace'-lohkon tallennusformaatiksi [Array](#) (EI Structure).
- Simulointitulokset ovat karkeita: muuta Solverin max step size pienempään arvoon kuin 'auto', esim. 0.01. ([Simulation](#) → [Configuration parameters](#))

PID-säädin Simulinkissä

- Simulinkissä PID-säädin löytyy kohdasta [Simulink](#) → [Continuous](#).
- Säätimen (ja muidenkin lohkojen) rakennetta voi tutkia [Mask](#) → [Look Under Mask](#)-kohdasta. (Oikealla hiiren näppäimellä aukeava valikko)



Tehtävä A: Peto-saalis-malli

1. Palauta mieleen differentiaaliyhtälömalli

$$\begin{aligned}\frac{d}{dt}x(t) &= a \cdot x(t) - b \cdot x(t) \cdot y(t) \\ \frac{d}{dt}y(t) &= -p \cdot y(t) + q \cdot x(t) \cdot y(t),\end{aligned}$$

joka kuvaa jänispopulaation $x(t)$ ja ilvespopulaation $y(t)$ kokoa ajanhetkellä t . Toteuta malli Simulinkillä.

- Vinkki: Katso kalvo Integrator-lohkon käytöstä (Et tarvitse säätimiä tässä tehtävässä)



Liitä vastauksiisi ruutukaappauskuva Simulink-mallistasi.


Tehtävä A: Peto-saalis-malli

2. Tutki populaatioiden kokojen muutoksia ajan funktiona, kun alussa jäniksiä on 5 yksilöä ja ilveksiä 2 yksilöä. Parametrien arvot ovat $a = 2$, $b = 0.2$, $p = 3$, ja $q = 0.1$.

 Liitä vastauksiisi kuva kummankin populaatioin koosta ajan funktiona.

Vinkki: tuo Simulinkistä Matlabin työtilaan kaksi vektoria (To Workspace-lohkoilla), joissa on jänis- ja ilvespopulaatioiden koot ajan funktiona ja käytä `plot` -komentoa.

Huom! Laita To Workspace lohkoissa save format: array.

 Miten populaatioiden koot käyttäytyvät toisiinsa nähden? (Arvioi esim. miten ilvesten lukumäärän kasvu vaikuttaa jänisten lukumäärään)

Tehtävä A: Peto-saalis-malli

3. Tutki populaatioiden kokojen muutoksia *vaihetasossa* (pystyakselilla ilvekset, vaaka-akselilla jänikset).



Liitä vastauksiisi kuva populaatioiden kokojen muutoksista vaihetasossa. Nimeä akselit siten, että kuvaajan pystyy tulkita.



Aseta alkutilanteessa jänisten määräksi p/q ja ilvesten määräksi a/b . Kuinka populaatiot tällöin käyttäytyvät ja miten tulkitset tilanteen?

Tehtävä B: Läpivirtaussäiliö

- Tehtävänäsi on johtaa tuotantoprosessiin nestettä tasaisella 20 l/s virtauksella kun tulevassa virtauksessa esiintyy häiriöitä.
- Nesteen määrän vaihteluita pyritään tasaamaan lieriön muotoisella läpivirtaussäiliöllä ja PID-säätimellä.
- Lataa kurssisivulta “Lisämateriaali”-osiosta prosessin ensimmäinen Simulink-malli. Tehtävänäsi on tutustua mallin sisältöön ja laajentaa sitä. Kokeile mallin toimintaa jokaisessa vaiheessa, joten piirrä Simulinkkiin kaavio jota laajennat vaiheittain.
- Tarkastele erityisesti säiliön vedenpinnan korkeutta ja tulo- ja poistovirtausta.

Tehtävä B: Läpivirtaussäiliö

1. Tarkastellaan aluksi vain joka on aluksi tyhjä ja johon johdetaan virtaus f_{in} . Säiliön nesteen tilavuus V riippuu syöttöputken virtauksesta f_{in} eli

$$\frac{dV(t)}{dt} = f_{in}(t)$$

Säiliössä nesteen tilavuuden mittaaminen suoraan on vaikeaa, mutta säiliön nestepinnan korkeutta h voi mitata helposti. Säiliöllä jonka pohjan pinta-ala A on vakio ($A = 5$) saadaan

$$A \frac{dh(t)}{dt} = f_{in}(t)$$

- Tutki Simulink-mallia käyttäen $h(t)$:tä systeemin tilamuuttujana.
- 🔗 Kuinka nestepinnan korkeus käyttäytyy, kun syöttöputken virtaus $f_{in} > 0$ on vakio eikä säiliössä ole poistoputkea?

Tehtävä B: Läpivirtaussäiliö

2. Lisää säiliöön poistoputki jolloin syntyy läpivirtaussäiliö. Tällöin

$$A \frac{dh(t)}{dt} = f_{in}(t) - f_{out}(t)$$

Poistovirtaus on pinnankorkeuden neliöjuureen verrannollinen:

$f_{out}(t) = k_v \sqrt{h(t)}$, missä $k_v = 2.0$. Tällöin saadaan epälineaarinen 1. kertaluvun differentiaaliyhtälö

$$A \frac{dh(t)}{dt} + k_v \sqrt{h(t)} = f_{in}(t)$$

- ✎ Kuinka nestepinnan korkeus käyttäytyy, kun lisäät poistoputken?
- ✎ Mille korkeudelle nestepinta tasapainottuu kun $f_{in} = 20$?

Tehtävä B: Läpivirtaussäiliö

3. Toteuta systeemille *askelvastekoe*, jossa systeemi on aluksi tasapainotilassa, jonka jälkeen sisäänmenoon aiheutetaan askelmainen häiriö.
- Aseta $h = 100$ Integrator-lohkossa.
 - Lisää Simulink mallissa systeemin sisäänmenoon 10:n yksikön suuruinen askelmainen poikkeama ajanhetkellä 10. Toteuta tämä step-lohkolla.
 - ✎ Kuinka kauan askelmaisen poikkeaman jälkeen kestää, että systeemi päätyy uuteen tasapainopisteeseen? Mikä on tällöin pinnankorkeus? Mikä on tällöin ulosmenovirtaus?

Tehtävä B: Läpivirtaussäiliö

- Lisää systeemiin PID-säätäjä, joka pyrkii pitämään pitämään poistovirtauksen tavoitetasolla 20 l/s säätelemällä tulovirtausta.
- Poistovirtauksen nopeutta on vaikea mitata, joten säätimen sisäänmenona täytyy käyttää tavoitellun ja havaitun pinnankorkeuden erotusta.
- ✎ Mikä on pinnankorkeuden tavoitetaso h_{ref} ?
- Tulovirtausta ei voi muuttaa äärettömän nopeasti (mieti todellista venttiiliä), vaan maksimi muutosnopeus on $\pm 5 \text{ l/s}^2$ (Vinkki: Johda säätimen tuottama ohjaus `Rate Limiter`-lohkon läpi ja summaa se vasta sitten f_{in} virtaukseen).

Tehtävä B: Läpivirtaussäiliö


4. Etsi säätimen parametrit siten, että askelmaisen häiriön jälkeen systeemi säätyy takaisin haluttuun tilaan (pinnankorkeus 100, poistovirtaus 20. Huom! Systeemin pitää siis säätyä tähän tilaan, ei siis riitä, että pelkästään seuraava ehto täyttyy.) Lisäksi poistovirtauksen tulee kokoajan pysyä välillä $f_{out} \in [19.8l/s, 20.2l/s]$.

Kokeile ensin pelkällä P-säätimellä . Käytä esim. vahvistuksen K_P arvoja 1,2,3,4,5,10. Kokeile sitten PI-säädintä.

- ☞ Saavutetaanko säätötavoite pelkällä P-säätimellä?
- ☞ Mitkä valitsit PI-säätimen parametreiksi lopulliseen malliisi?





Kotitehtävä: Läpivirtaussäiliö

1. Visualisoi vesisäiliösystemin toimintaa kun tulovirtausta säädetään PI-säätimellä ja siihen vaikuttaa edellä kuvattu askelmainen häiriö.

 Liitä vastauksiisi `subplot`-komennolla piirretty kuva, jossa on ajan funktiona f_{in} , f_{out} , h (Huom! tässä tarkoitetaan f_{in} :llä kaikkien säiliöön virtaavien virtausten summaa) ja PI-säätimen ulostulo (sekä ennen että jälkeen muutosnopeudenrajoittimen). Nimeä kuvaajien akselit ja suureiden yksiköt.

 Liitä vastauksiisi ruutukaappauskuva Simulink-mallistasi.

Kotitehtävä: Läpivirtaussäiliö

2. Tutkitaan systeemin taajuusvastetta. Korvaa askelmainen häiriö sinimuotoisella (amplitudi $1l/s$) ja aseta PI-säätimen kaikki parametrit nolaksi. Kokeile eri taajuuksilla ($0.01 \dots 1 \text{ rad/s}$) ja piirrä samaan kuvaan f_{in} ja f_{out} .
 -  Miten tulovirtauksen häiriön siirtyminen poistovirtaukseen riippuu häiriön taajuudesta?
 -  Voiko vesisäiliösystemin tulkita ali- tai ylipäästösuodattimeksi? (Katso tarvittaessa nämä termit netistä)
 -  Liitä vastauksiisi `subplot`-komennolla piirretty kuva, jossa on yhdessä kuvassa ajan funktiona f_{in} ja f_{out} , kun tulovirtaukseen vaikuttaa sinimuotoinen häiriö, jonka taajuus on 1 rad/s . Piirrä toiseen kuvaan f_{in} ja f_{out} , kun häiriön taajuus on 0.01 rad/s .
 -  Liitä vastauksiisi ruutukaappauskuva Simulink-mallistasi.