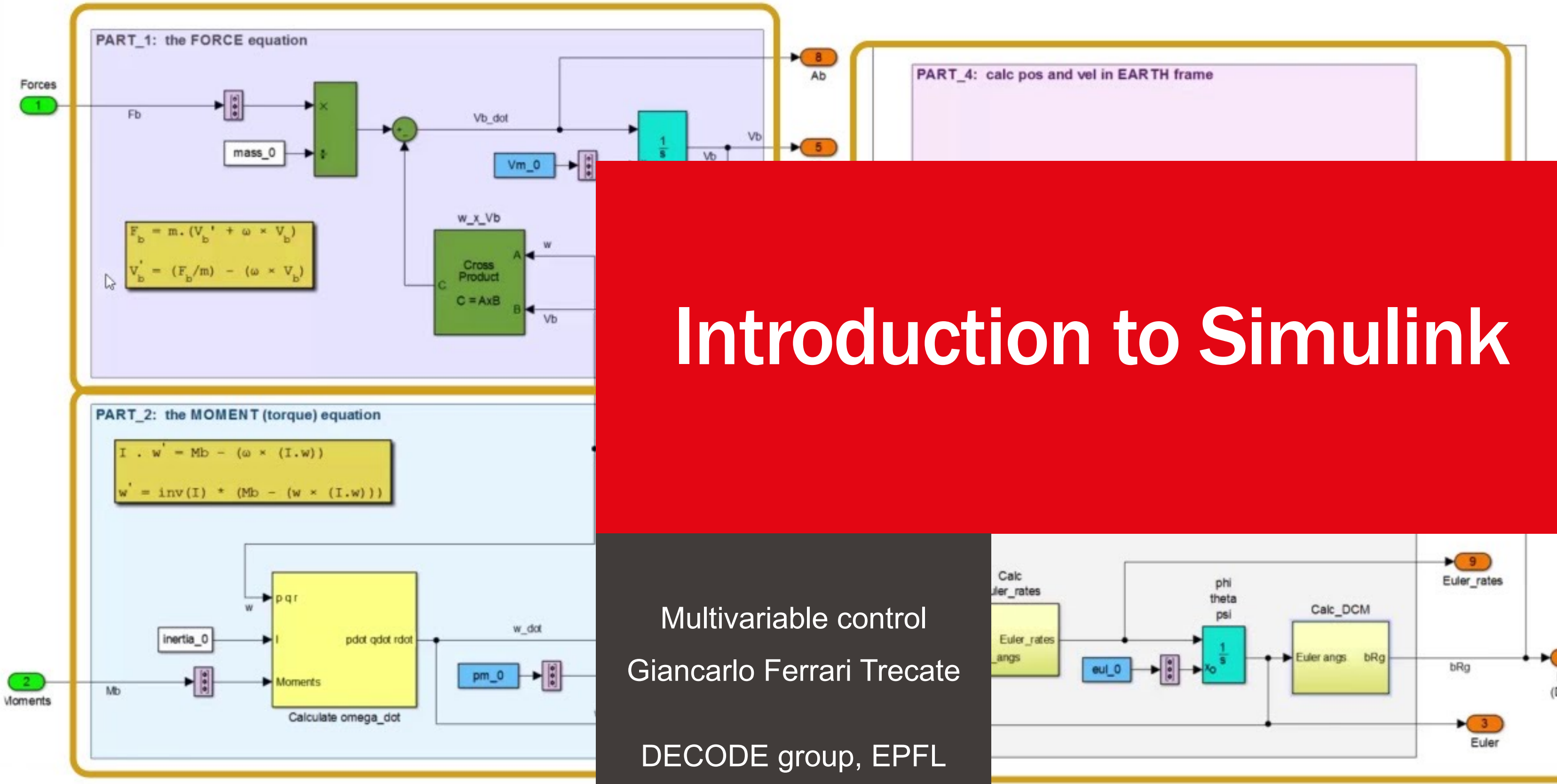
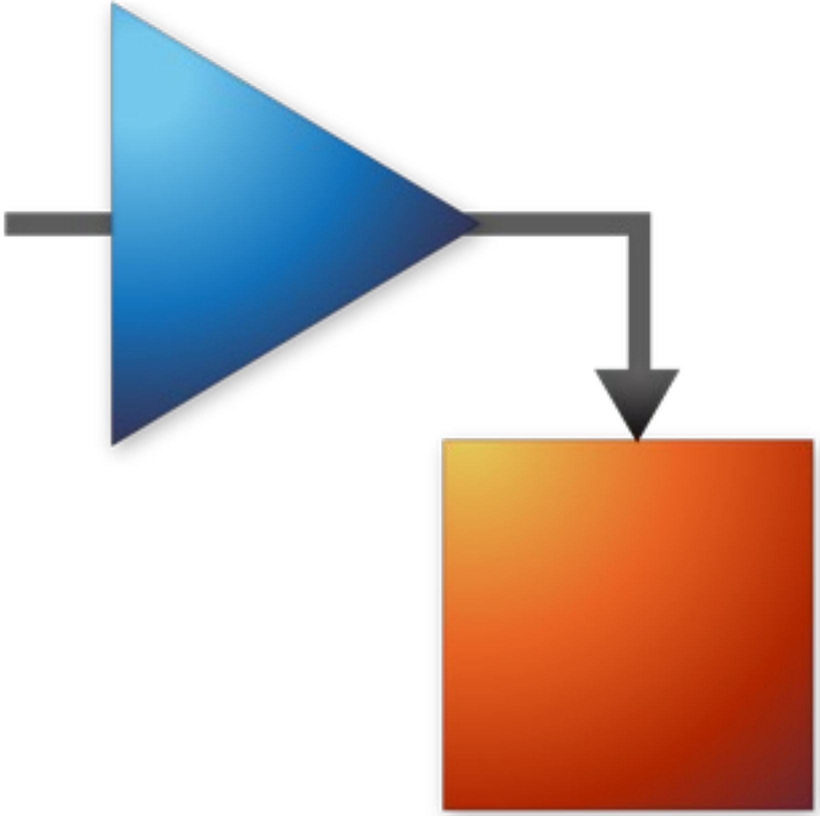


Introduction to Simulink

Multivariable control

Giancarlo Ferrari Trecate

DECODE group, EPFL



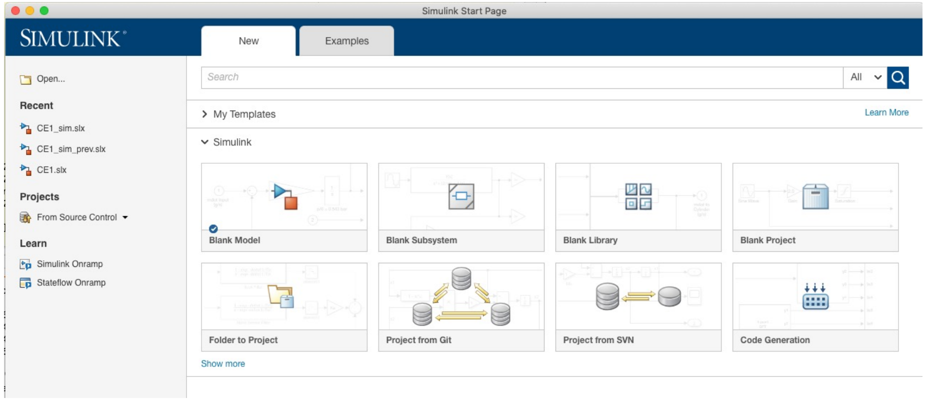
Basic elements

Simulink

- *«Simulink is a block diagram environment used to **design** systems with multidomain models, **simulate** before moving to hardware, and **deploy** without writing code»*

How to start?

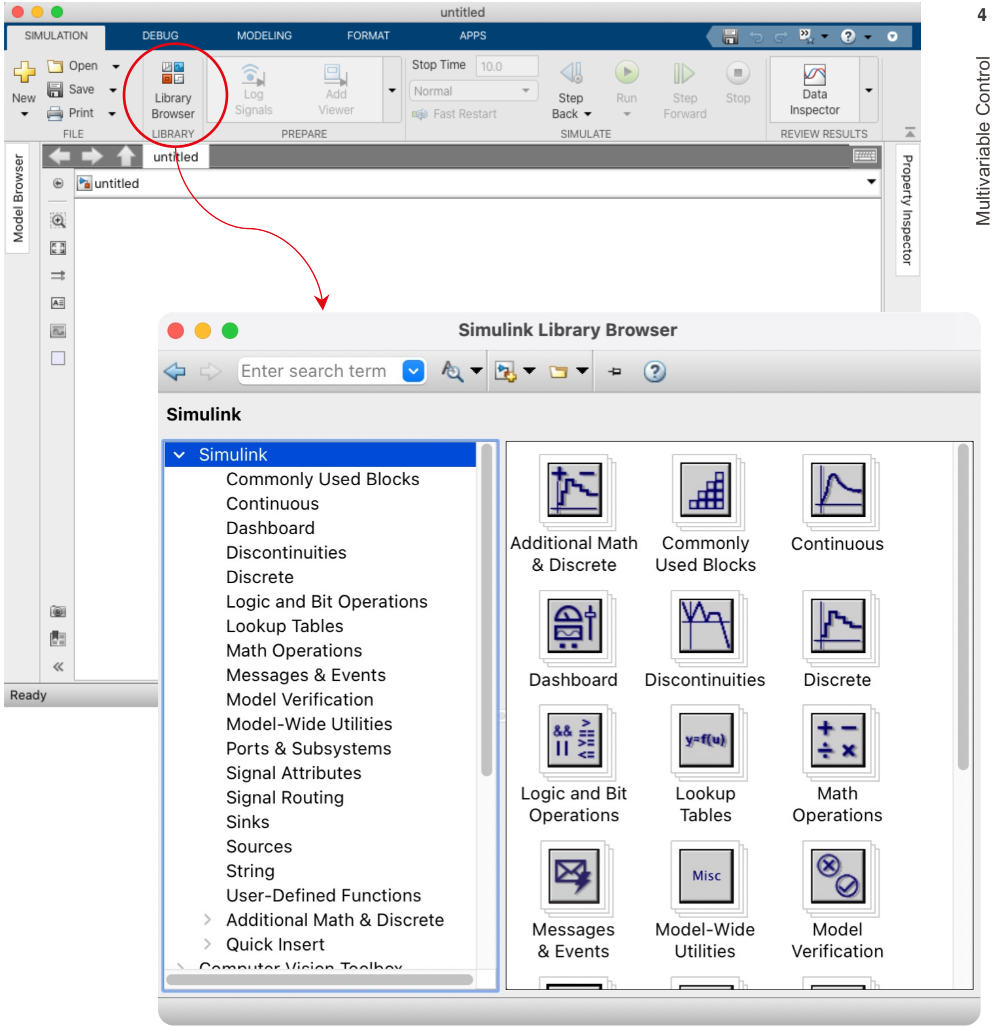
- Type `simulink` at the MATLAB `>>` prompt and press *enter*
  - Select 'Blank Model' in the new window
  - *Note: the control systems toolbox needs to be installed*



Create new model

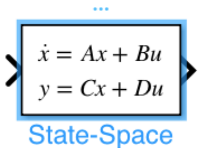
- Click on ‘Library Browser’

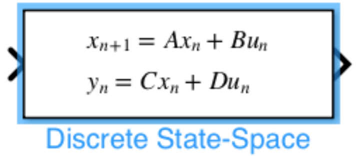
Building blocks for constructing models



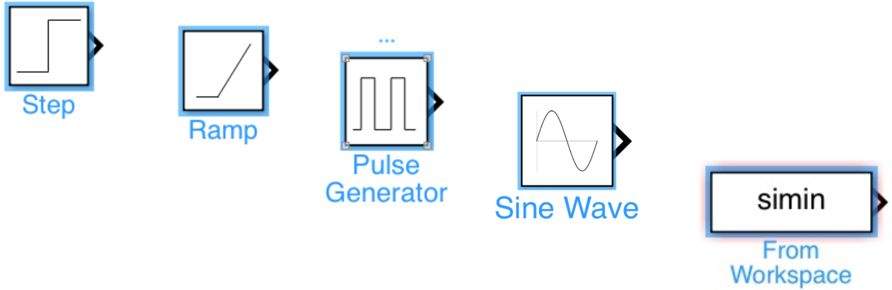
Create the model in Simulink

- For the dynamics: use the (discrete) state-space block which can be found under ‘Continuous’ or ‘Discrete’ tabs



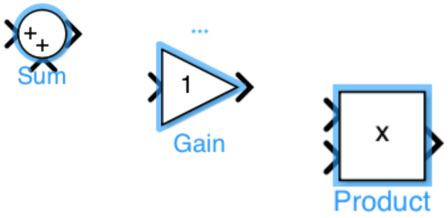


- Input signals can be found under ‘Sources’ tab
  - Step
  - Ramp
  - Pulse
  - Sinusoidal
  - From Workspace
  - …

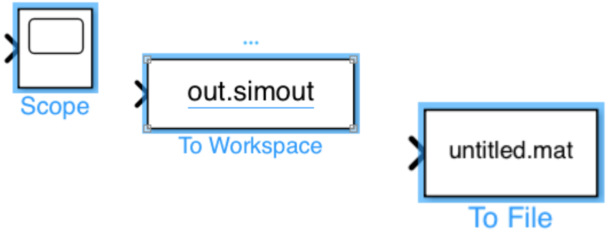


Create the model in Simulink

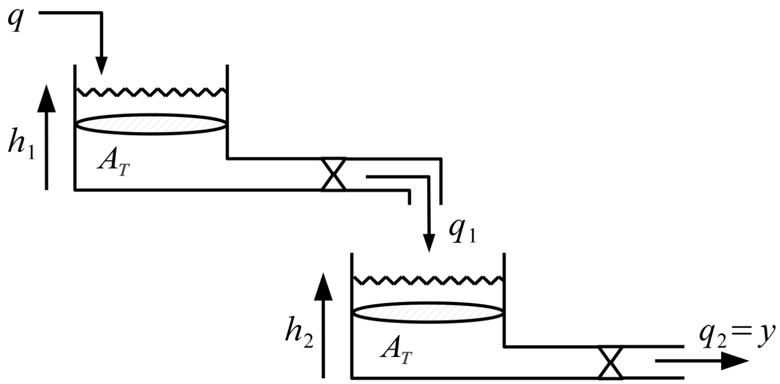
- Combine signals using blocks under ‘Math operations’ tab
  - Sum
  - Gain
  - Product
  - …



- Display or export signals using blocks under ‘Sinks’ tab
  - Scope
  - To Workspace
  - To File
  - …



Example: double tank system



Parameters

$$A_T = 1\ [\mathrm{m}^2]$$

Model

$$A_T \dot{h}_1 = q - 3h_1$$
$$A_T \dot{h}_2 = 3h_1 - 5h_2$$
$$y = 5h_2$$

Dynamical system

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -3 & 0 \\ 3 & -5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u$$

$$y = \begin{bmatrix} 0 & 5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

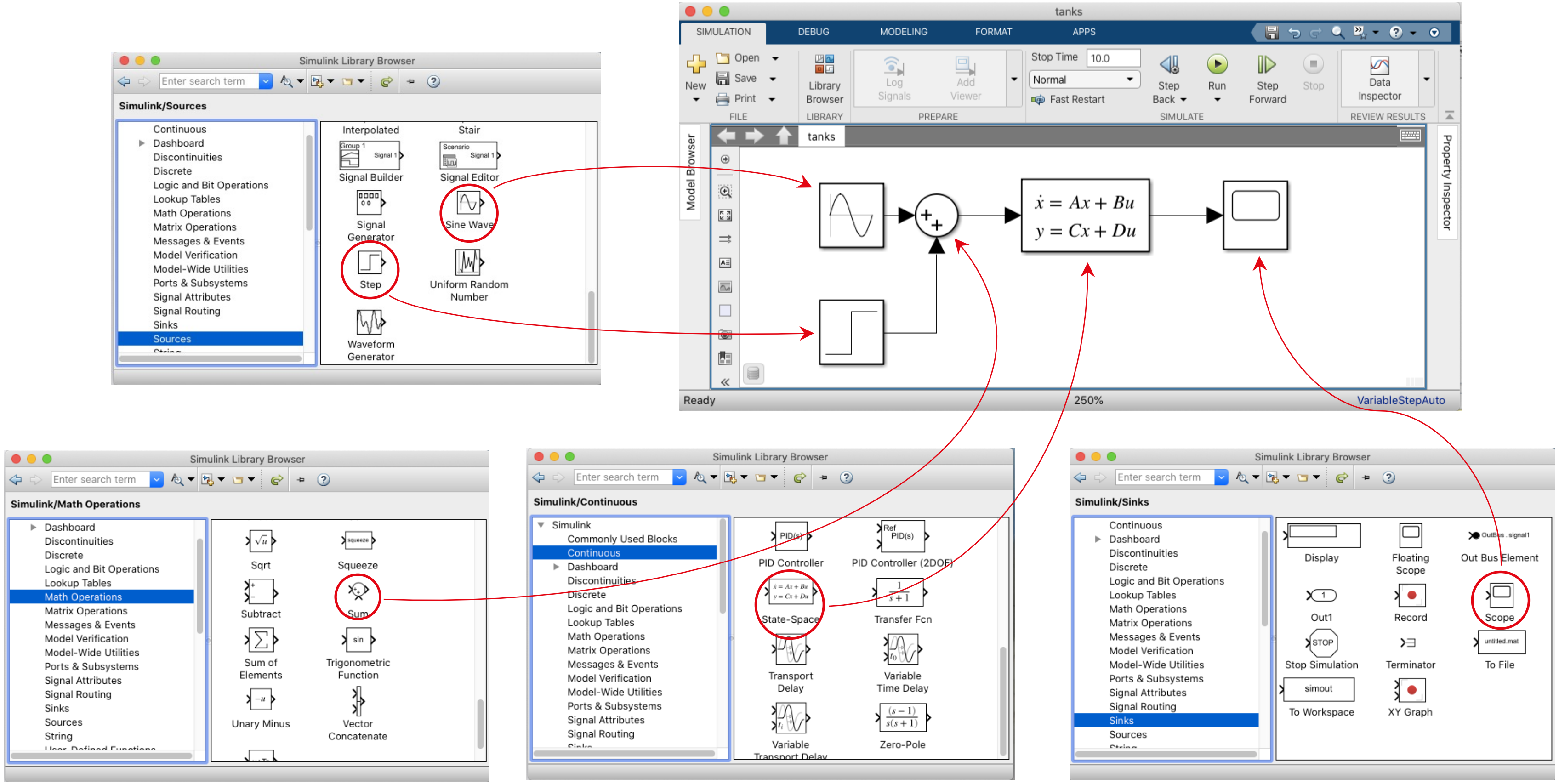
- Input signal: sinusoidal + step functions
  - Define their parameters in a Matlab script:

```
%% Double tank system
% Sinusoidal
a = 0.2;            % [m^3/s]
omega = 5;          % [rad/s]
% Step
qbar = 1.5;         % [m^3/s]
t_step = 0;
```

- Set the initial conditions: $h_1(0) = h_2(0) = 0.1\ [\mathrm{m}]$
- Plot the output using a Scope block

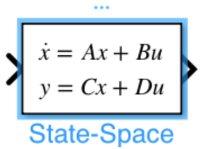
Create the model in Simulink

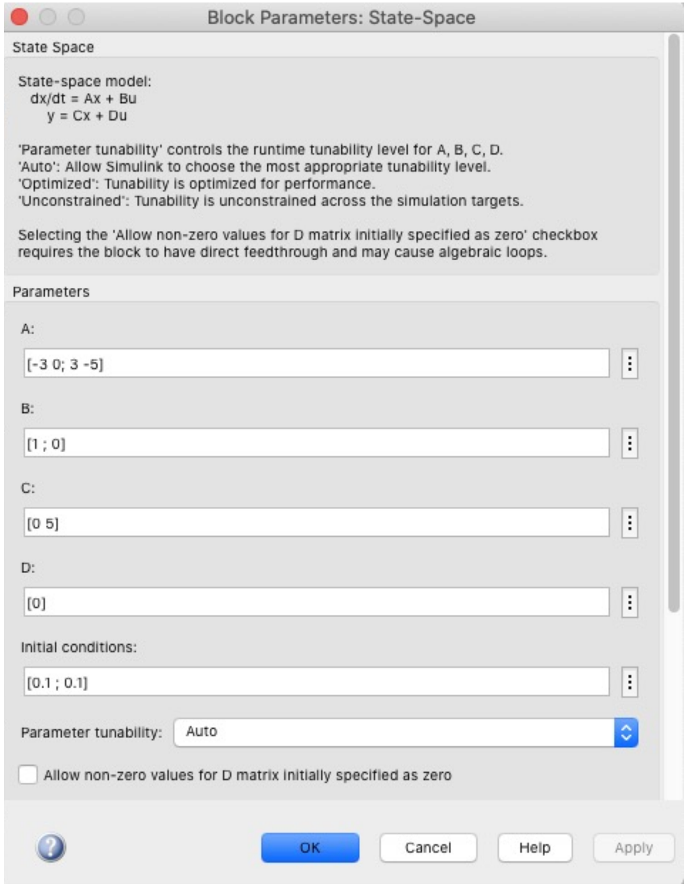
- To add blocks in the model, drag and drop them from the library



State space block

- Double click on a block to access and modify its properties/parameters
  - E.g.:





  - Recall:

Dynamical system

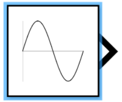
$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -3 & 0 \\ 3 & -5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u$$

$$y = \begin{bmatrix} 0 & 5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

  - Initial conditions: $h_1(0) = h_2(0) = 0.1\ [\mathrm{m}]$

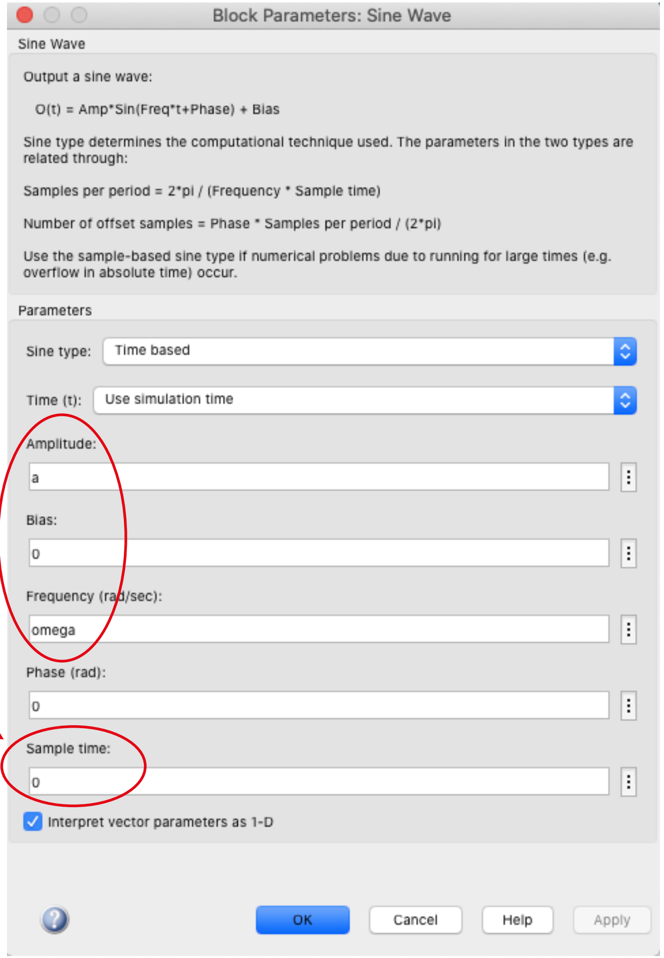
Source block

- E.g.:



Sine Wave

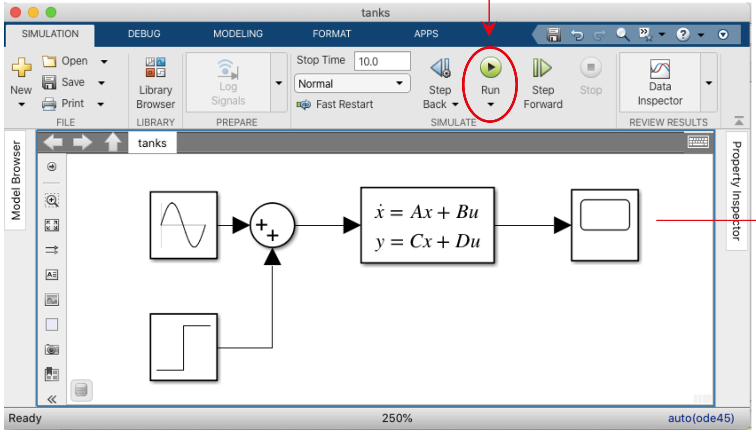
- Variables defined in the workspace can be used for setting the parameters

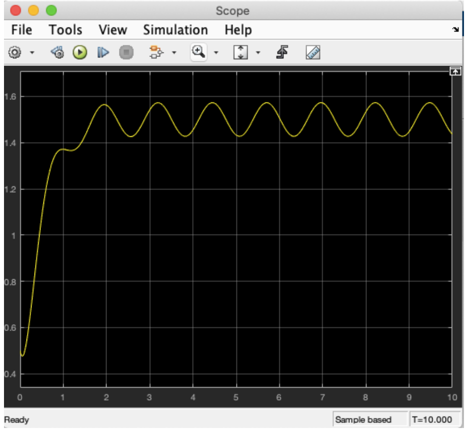


- *Note: sample time can be set for discrete and hybrid blocks*
  - *Set it to **0** for continuous time*
  - *Set it **-1** to indicate it is inherited from previous block or model*

Run the Simulink model

- Press on the ‘Run’ button to simulate the system
- Double click on the scope block to observe the results

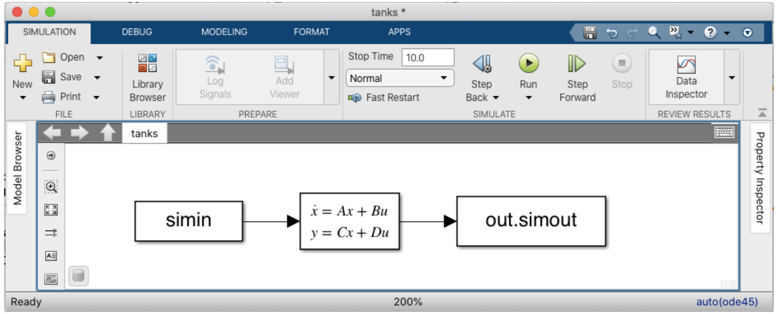




- *Note: output can be refined by changing the ‘Refine factor’ in Modelling → Model Settings → Data Import/Export →Additional parameters → Save options*

Run the Simulink model from Matlab

- To do at home:
  - Replace source and sink blocks with from/to workspace blocks

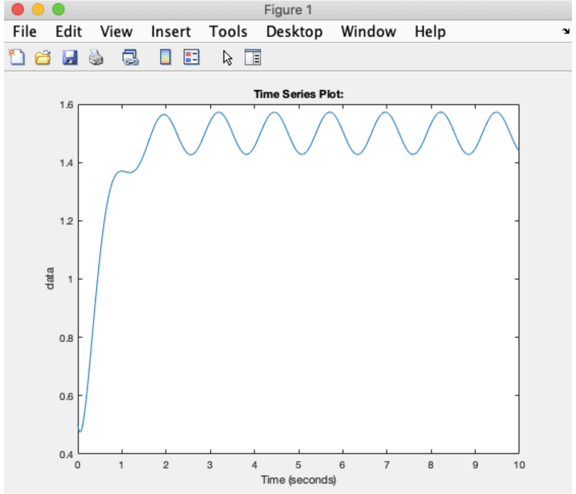


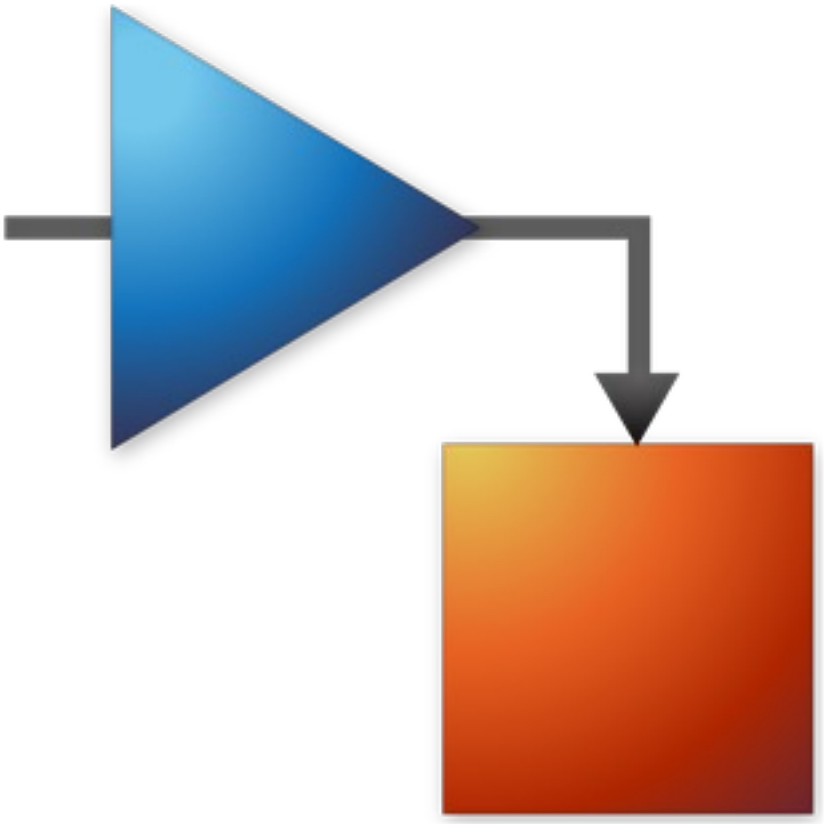
  - Define the input signal in the workspace:

```
% Simin signal
simin.time = [0:0.1:10]';
simin.signals.values = qbar + a*sin(omega*simin.time);
```

  - Save the model as ‘tanks’ and run the system from the workspace

```
% Simulate the system
model = sim("tanks");
plot(model.simout)
```

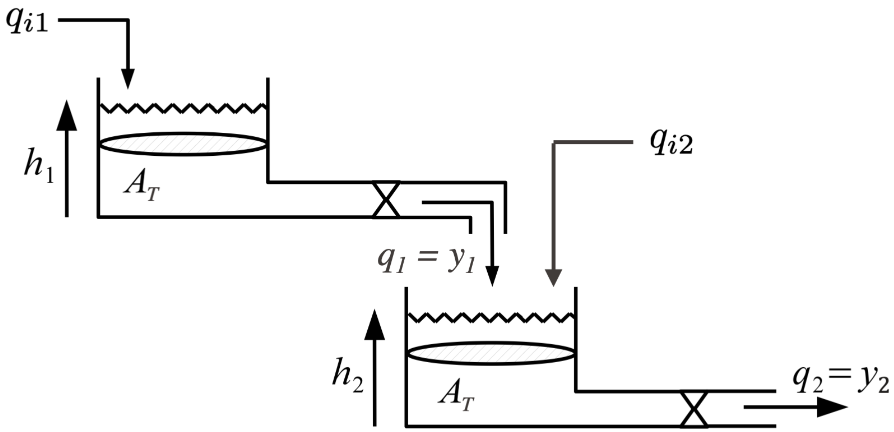




Multi-input multi-output systems

MIMO Simulink models

Consider the previous example with an additional input and output:



Model

$$A_T \dot{h}_1 = q_{i1} - 3h_1$$
$$A_T \dot{h}_2 = q_{i2} + 3h_1 - 5h_2$$
$$y_1 = 3h_1$$
$$y_2 = 5h_2$$

- with the following inputs:
  - $q_{i1} = \bar{q}_{i1} + a\sin(\omega_1 t)$
  - $q_{i2} = \bar{q}_{i2} + a\sin(\omega_2 t)$
  - $\bar{q}_{i1} = 1.5 \ \left[\mathrm{m}^3/\mathrm{s}\right] \ , \ \bar{q}_{i2} = 0.5 \ \left[\mathrm{m}^3/\mathrm{s}\right]$
  - $a = 0.2 \left[\mathrm{m}^3/\mathrm{s}\right] \ , \ \omega_1 = 8 \,[\mathrm{rad/s}] \ , \ \omega_2 = 5 \,[\mathrm{rad/s}]$
- and initial conditions: $h_1(0) = h_2(0) = 0.1 \ [\mathrm{m}]$

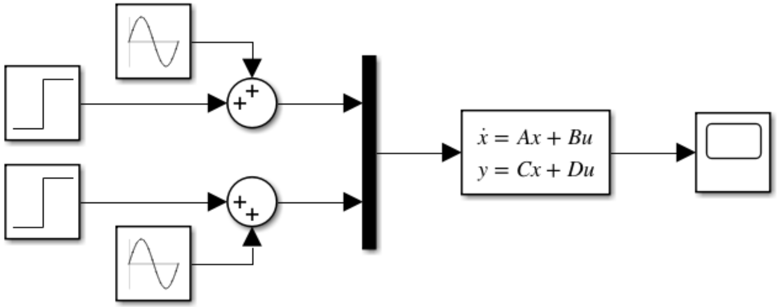
Dynamical system

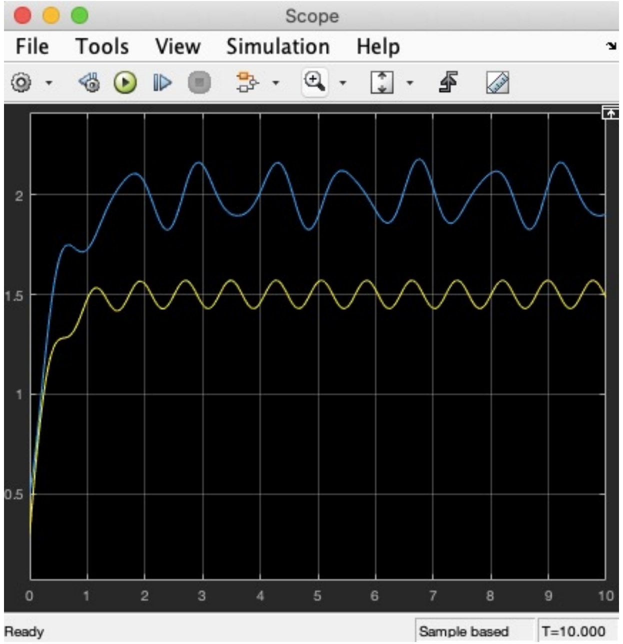
$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -3 & 0 \\ 3 & -5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} u$$
$$y = \begin{bmatrix} 3 & 0 \\ 0 & 5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

MIMO Simulink models

- Using signals from Simulink:
  - Use ‘mux’ and ‘demux’ blocks for combining signals (in ‘Signal routing’ tab, in the Library)

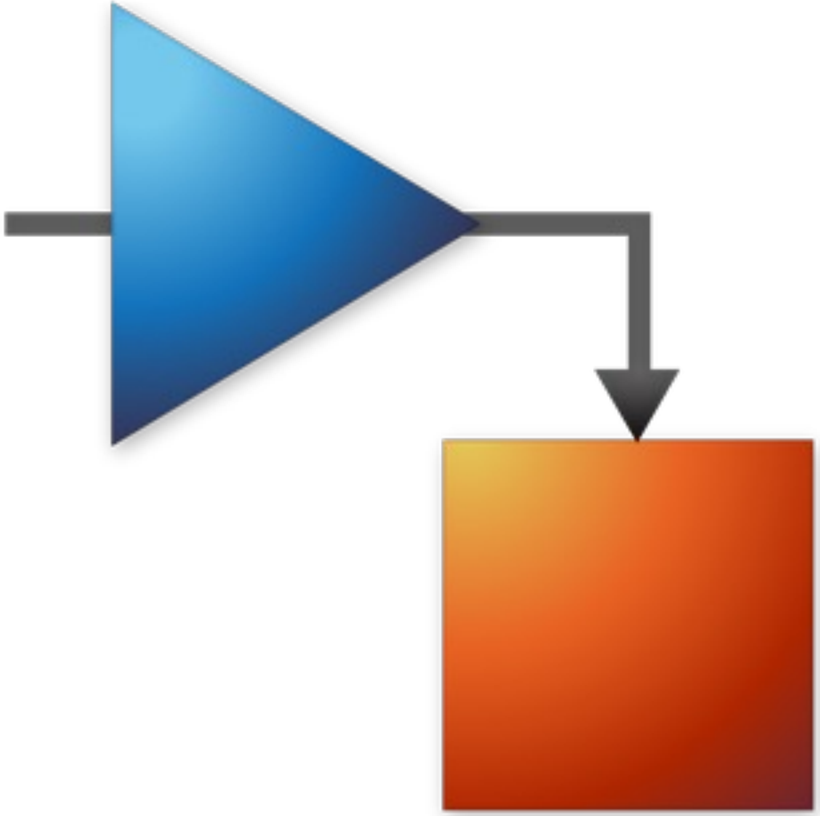
    ↪ Change the number of inputs/outputs in the properties panel





- Using signals from workspace:
  - Set `simin` to a multi-column matrix, e.g.:

```
simin2.time = [0:0.1:10]';
simin2.signals.values = horzcat(qbar1 + a*sin(omega1*simin2.time),...
    qbar2 + a*sin(omega2*simin2.time));
```



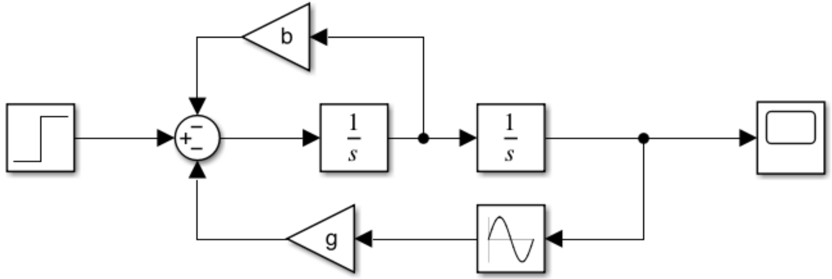
Non-linear systems

Non-linear systems in Simulink

- **Idea 1:** Interconnect different non-linear blocks to create the desired function
  - E.g.: Pendulum

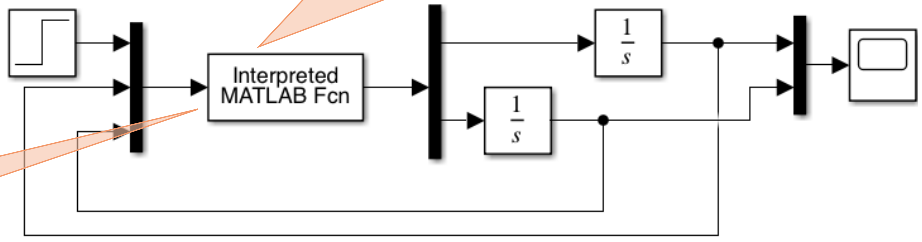
$$\dot{\theta} = \omega$$

$$\dot{\omega} = -b\omega - g\sin\theta + u$$



- **Idea 2:** Define a non-linear function for characterizing the dynamics

```
% File: pendulum_func.m
function dx = pendulum_func(u,b,g)
uin = u(1);
theta = u(2);
omega = u(3);
dx(1) = omega;
dx(2) = -b*omega - g*sin(theta) + uin
end
```



Block in 'User-defined functions' tab of the Simulink library

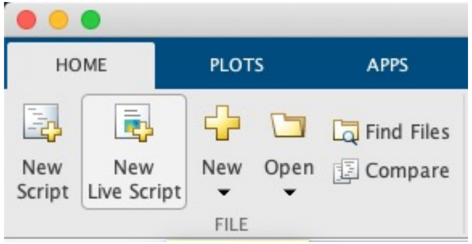
In its properties, set `pendulum_func(u,b,g)` as the Matlab function

Don't forget to define b and g in the workspace

MATLAB live scripts

Live scripts

- *«Live scripts are interactive documents that combine Matlab code with formatted text, equations, and images in a single environment. In addition, live scripts store and display output alongside the code that creates it.»*



How to start?

- Live scripts are divided in sections
- Right-click on the document for:
  - 'Section break', and thus create a new section
  - 'Convert between code and text'
  - 'Convert to (local) function'
- Run a section with ctrl+enter (cmd+enter, in Mac)
- Live scripts are saved as `.mlx` files

An example of a livescript containing today's examples and exercises can be found under `intro_simulink.mlx`

Live script example

Let define now the input signals. For that purpose, we add a step and sinusoidal block in the Simulink model and define here their parameters.

```
% Sinusoidal
a = 0.2;        % [m^3/s]
omega = 5;      % [rad/s]
% Step
qbar = 1.5;     % [m^3/s]
t_step = 0;
```

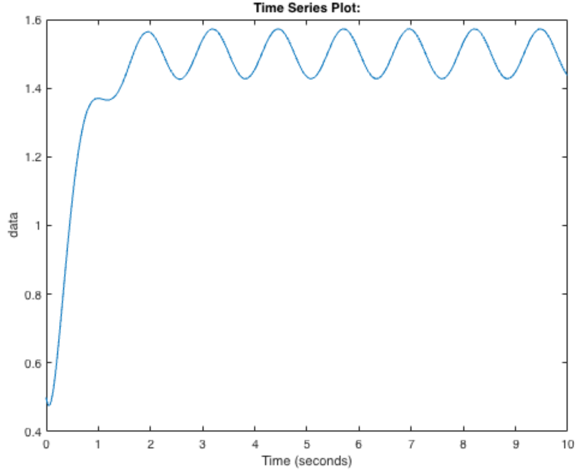
We also define the input in an alternative way. We construct it in Matlab and pass the sequence to Simulink using the simin block.

```
% Simin signal
simin.time = [0:0.1:10]';
simin.signals.values = qbar + a*sin(omega*simin.time);
```

By clicking on run, or running the following lines, we simulate the system. The output can be observed by double click on the scope block. Alternatively, one can export the signals to Matlab and then plot them, as done in next section.

Note that the plot and displayed text appears in this live-script. In may appear in the right column on below the section of the code tha generated it, depending on your settings.

```
% Simulate the system
model = sim("tanks");
plot(model.simout)
fprintf("The input signal consists of a step of " + ...
    "amplitude %.1f, \nplus a sinusoidal of amplitude " + ...
    "%.1f and frequency %.1f rad/s.", qbar, a, omega)
```

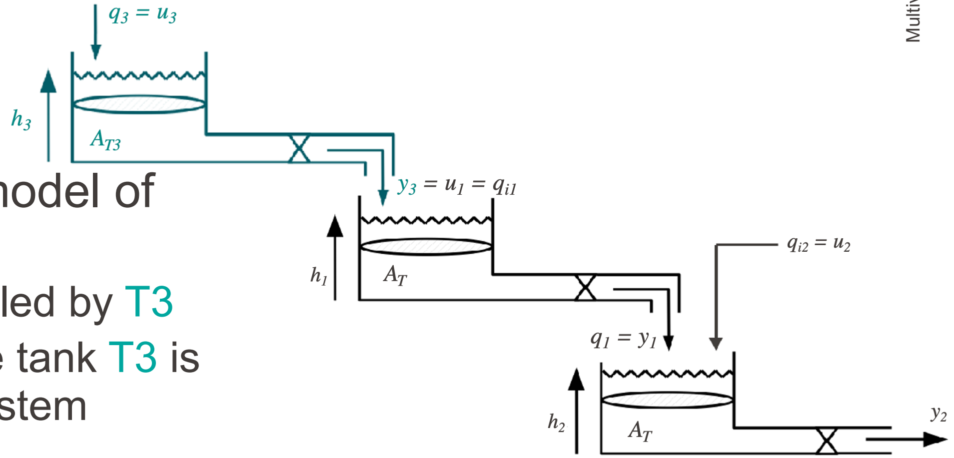



```
The input signal consists of a step of amplitude 1.5,
plus a sinusoidal of amplitude 0.2 and frequency 5.0 rad/s.
```

Exercise (optional)

Consider a third tank (T3) whose model is given by

- $3\dot{h}_3 = 5u_3 - 2h_3$
- $y_3 = 2h_3$



1. Consider the previous MIMO model of two tanks
   - Add a third tank at the top modelled by T3
   - Set $u_1 = y_3$, i.e. the output of the tank T3 is the first input of the two-tanks system

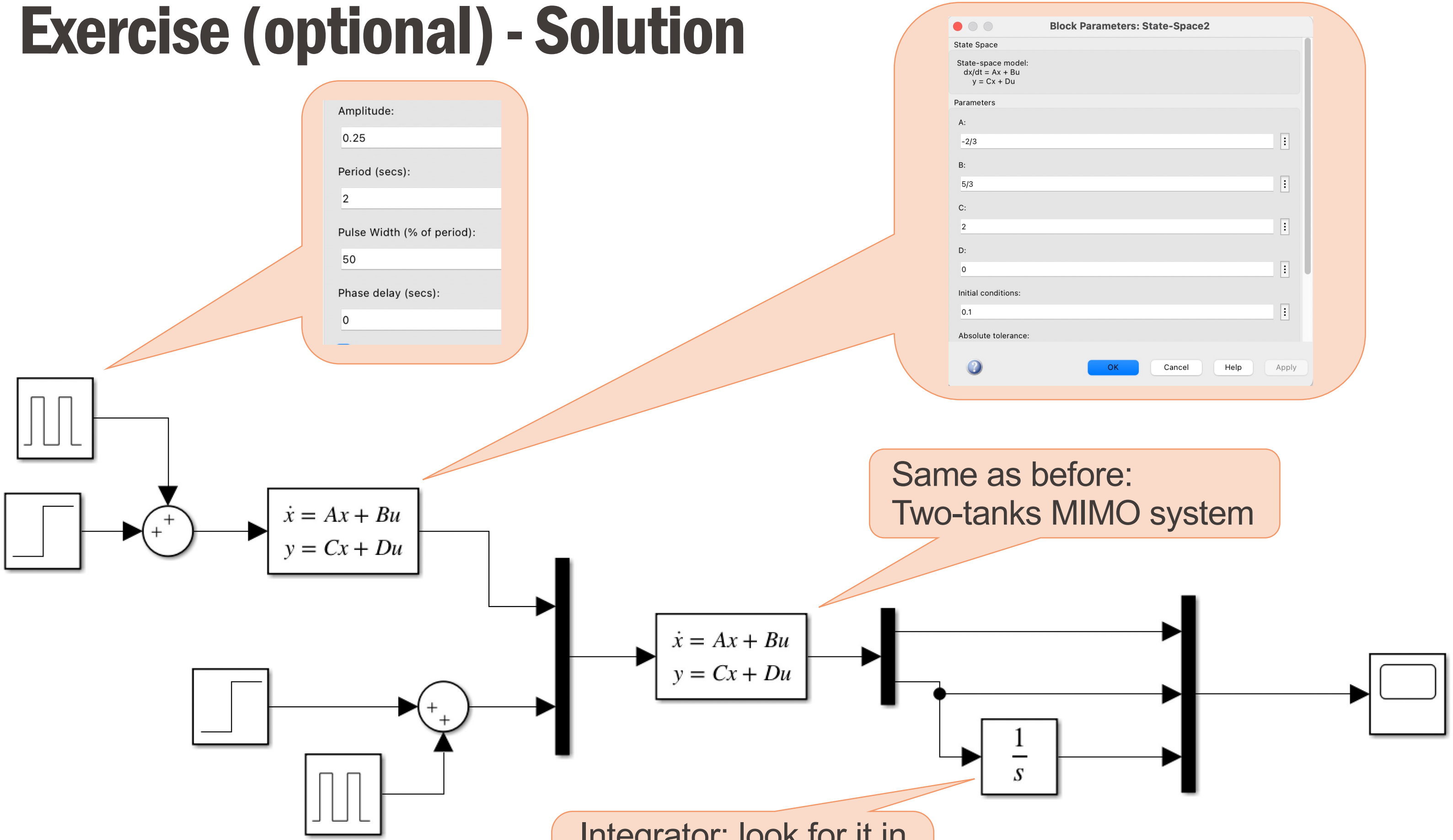
   a. Plot the output of the system. Consider the inputs u_3 and u_2 given by a pulse with an offset, i.e.
      - $u_3 = \bar{q}_{i1} + \mathrm{pulse}(a_1, \tau_1)$
      - $u_2 = \bar{q}_{i2} + \mathrm{pulse}(a_2, \tau_2)$

      Set the initial conditions $h_1(0) = h_2(0) = h_3(0) = 0.1\ [\mathrm{m}]$

   b. Add as an output the integral of the bottom tank's flow

$$\bar{q}_{i1} = 1.5\ [\mathrm{m}^3/\mathrm{s}]\ ,\ \bar{q}_{i2} = 0.5\ [\mathrm{m}^3/\mathrm{s}]$$
$$a_1 = 0.25\ [\mathrm{m}^3/\mathrm{s}]\ ,\ a_2 = 0.4\ [\mathrm{m}^3/\mathrm{s}]$$
$$\tau_1 = 2\ [\mathrm{s}]\ ,\ \tau_2 = 3\ [\mathrm{s}]$$

Exercise (optional) - Solution



Same as before: Two-tanks MIMO system

Integrator: look for it in the ‘Continuous’ tab of the Library

Solution in file `tanks3.slx`