MATLAB Examples

Simulink

Hans-Petter Halvorsen, M.Sc.
What is Simulink?

• Simulink is an “add-on” to MATLAB.
• You need to have MATLAB in order to use Simulink
• Simulink is used for Simulation of dynamic models
• In Simulink we create a Graphical Block Diagram for the system (based on the differential equations(s))
In Simulink we create and configure graphical blocks and wire them together (based on the differential equations)
Start Simulink from MATLAB

Click the “Simulink” button in the Toolbar - or type “simulink” in the Command Window
Start creating your Simulink Model here with blocks from the “Simulink Library Browser” (just “Drag and Drop”)

Simulink Model Editor

Library Browser
Simulink Library Browser
Simulink Example
Simulink Example II
Example

My First Simulink Model

Model:

\[ \dot{x} = ax \]

Where

\[ a = -\frac{1}{T} \]

\( T \) is the Time constant

We will use the following:

\[ T = 5 \]

\( x(0) = 1 \)

\[ 0 \leq t \leq 25 \]

We start by drawing a simple Block Diagram for the model like this (“Pen & paper”):

We will create and simulate this block diagram with Simulink
Using ODE Solvers in MATLAB

\[ \dot{x} = ax \]

**Step 1:** Define the differential equation as a MATLAB function (`mydiff.m`):

```matlab
function dx = mydiff(t, x)
T = 5;
a = -1/T;
dx = a*x;
```

**Step 2:** Use one of the built-in ODE solver (ode23, ode45, ...) in a Script.

```matlab
clear;
clc;
tspan = [0 25];
x0 = 1;
[t, x] = ode23(@(x) mydiff(t, x), tspan, x0);
plot(t, x);
```
My First Simulink Model
My First Simulink Model

Solution

Set Initial Value

$x(0) = 1$

Gain

$\alpha = \frac{1}{T}$

Integrator

Scope

Start the Simulation by clicking this icon

Simulation Time $0 \leq t \leq 25$

See the Simulation Results

Double-Click on the different Blocks to enter values
Bacteria Population

Here we will simulate a simple model of a bacteria population in a jar. The model is as follows:

\[ \text{birth rate} = bx \]
\[ \text{death rate} = px^2 \]

Then the total rate of change of bacteria population is:
\[ \dot{x} = bx - px^2 \]

Set \( b=1 \) /hour and \( p=0.5 \) bacteria-hour

\( \rightarrow \) We will simulate the number of bacteria in the jar after 1 hour, assuming that initially there are 100 bacteria present.
function dx = bacteriadiiff(t,x)
% My Simple Differential Equation
b=1;
p=0.5;
dx = b*x - p*x^2;
clear
clc
tspan=[0 1];
x0=100;
[t,y]=ode45(@bacteriadiiff, tspan,x0);
plot(t,y)
[t,y]
Block Diagram for the Model ("Pen and Paper")
Simulink Block Diagram for the Model

Bacteria Population
\[ x' = bx - px^2 \]
Combining MATLAB and Simulink

Data-driven Modelling

• You may use Simulink together with MATLAB in order to specify data and parameters to your Simulink model.
• You may specify commands in the MATLAB Command Window or as commands in an m-file (Script).
• This is called data-driven modeling
• Instead of using values directly we use variables instead - This is more flexible because we can easily change Simulation Parameters without touching the Simulink Model
Instead of using values directly we use variables instead – This is more flexible because we can easily change Simulation Parameters without changing the Simulink Model.

\[ \dot{x} = ax \]
Data-driven Modelling

MATLAB Script for running the Simulink Simulation:

```matlab
clear
clc

% Set Simulator Settings
x0=1;
T=5;
a=-1/T;
t_stop=25; % [s]
T_s=t_stop/1000; % [s]
options=simset('solver', 'ode5', 'fixedstep', T_s);

% Starting Simulation
sim('simulink_ex2', t_stop, options);
```

We get the same results:

This is the Name for our Simulink Model
Mass-Spring-Damper System

In this example we will create a mass-spring-damper model in Simulink and configure and run the simulation from a MATLAB m-file.

The differential equation for the system is as follows:

\[ \ddot{x} = \frac{1}{m} (F - c\dot{x} - kx) \]

Where:
- \( x \) - position
- \( \dot{x} \) - speed
- \( \ddot{x} \) - acceleration

Instead of hard-coding the model parameters in the blocks you should refer to them as variables set in an m-file.
The following variables should then be set in the m-file:

```matlab
x_init = 4; %[m]. Initial position.
dxdt_init = 0; %[m/s]. Initial Speed.
m = 20; %[kg]
c = 4; %[N/(m/s)]
k = 2; %[N/m]
t_step_F = 50; %[s]
F_O = 0; %[N]
F_1 = 4; %[N]
```
%Script of mass-spring-damper simulator.
%Hans-Petter Halvorsen. 20.11.2009

% Model Parameters
x_init=4; %[m]. Initial position.
dxdt_init=0; %[m/s]. Initial Speed.
m=20; %[kg]
c=4; %[N/(m/s)]
k=2; %[N/m]
t_step_F=50; %[s]
F_0=0; %[N]
F_1=4; %[N]

% Simulator Settings
t_stop=100; %[s]
T_s=t_stop/1000; %[s]
options=simset( 'solver', 'ode5', 'fixedstep', T_s);

% Starting simulation
sim('mass_spring_damper', t_stop, options);
Force $F$:

Position $x$ and speed $\dot{x}$:
Hans-Petter Halvorsen, M.Sc.

University College of Southeast Norway

www.usn.no

E-mail: hans.p.halvorsen@hit.no
Blog: http://home.hit.no/~hansha/