Simulink Basics

SIMULINK

Basics

Matlab/Simulink



Simulink Basics Content

- What's SIMULINK?
- SIMULINK–Libraries *Sources*, *Sinks* und *Math*
- Simulation parameters
- Algorithm's for numerical integration
- SIMULINK-Libraries *Signals&Systems*, *Subsystems*



Matlab/Simulink

SIMULINK Basics What's Simulink?

- Graphical modelling of dynamic systems by signal flow graphs
- Addition to MATLAB (Toolbox)
- Simulink–Additions: Blocksets (SimPower-Systems, SimMechanics)





SIMULINK Basics Starting SIMULINK

• Blocks are basic elements



• Blocks characterized by input, output, name, icon

• Double click opens *Block Parameters* dialog

Fon	
General expression block. Use 'M' as the input variable name. Example: $\sin(u[1])^*\exp\{2,3^*,u[2]\}$	
Parametera	
Expression	
sin u 1 'exp 2.3'(u(2)))	
Sample time (-1 for inherited):	
4	
participation (1997) - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997	- mo
DK Cancel Hel	a Apply



Matlab/Simulink

SIMULINK Basics SIMULINK block libraries Sources and Sinks

Block library Sources:

- Generation of signals
- Import data from MATLAB workspace
- Import of data from files

Block library Sinks:

- Graphical display of signals
- Write data to MATLAB workspace
- Write data to files







SIMULINK Basics Example for Sources and Sinks



Signal Builder: bsp_sigbuild.mdl

Signal Builder Dialog box for group of three signals



Matlab/Simulink



SIMULINK Basics SIMULINK block library Math Opertions

Block library *Math*:

- Arithmetic, logical and relational operators
- Mathematical and trigonometric functions



etc.



SIMULINK Basics Examples for Math Operations

Simulink model bspmath.mdl of equation

$$f(t) = 80 \cdot e^{-\frac{1}{80}t} \cdot \sin(0.25t + \frac{\pi}{3})$$







SIMULINK Basics Simulation: parameters and solvers

Configuration Parameters dialog box

Configuration Paramet	ters: untitled/Confi	guration					×
Select: Solver Data Import/Export	Start time: 0.0			Stop time: 10.0			
Optimization ⊡-Diagnostics Hardware Implementation	Solver options	Variable-step	•	Solver:	ode45 (Dormand-Prince)	_	
Model Referencing Real-Time Workshop	Max step size: Min step size:	auto	_	Relative tolerance: Absolute tolerance:	1e-3 auto	_	
	Initial step size: Zero crossing control:	auto Use local settings	•				
				<u>0</u> K <u>C</u> a	ncel <u>H</u> elp	<u>A</u> pply	

Solver pane:

- Specify start and stop time of simulation
- Solvers for numerical integration
- Output options



Inhomogeneous linear ordinary differential equation:

$$u(t) \longrightarrow DGL \longrightarrow y(t)$$
 $\dot{y}(t) = f(u(t), y(t))$

Integration:

$$y_{n+1} = y_n + \int_{t_n}^{t_{n+1}} f(u(t), y(t)) dt$$

Numerous methods:

- Euler method
- Heun's method
- Other Runge–Kutta–methods
- Adams–Bashforth methods

• • • •

Matlab/Simulink



Euler method (explicit)





Matlab/Simulink







$$y_2 = y_0 + \frac{2h}{6} \cdot [\dot{y}_0 + 4\dot{y}_1 + f(u_2, y_2^P)]$$

Matlab/Simulink

Dipl.-Ing. U. Wohlfarth

11



$$y_{n+1} = y_n + \frac{h}{12} \cdot [23\dot{y}_n - 16\dot{y}_{n-1} + 5\dot{y}_{n-2}]$$

Matlab/Simulink

Dipl.-Ing. U. Wohlfarth

12

SIMULINK Basics Solver: integration algorithms in MATLAB

Variable-step solver:

- use variable step size
- allow error control & detection of *zero crossings*
- for continuous-time, non-stiff systems: \Rightarrow ode45 (first try), ode23, ode113
- for continuous-time, stiff systems:
 - \Rightarrow ode15s, ode23s, ode23t,ode23tb
- for discrete-time systems:
 - \Rightarrow discrete (Variable-step)



SIMULINK Basics

Solver: integration algorithms in MATLAB

Fixed-step solver:

- use fixed step size
- no error control
- no detection of *zero crossings*
- for continuous-time systems:

 \Rightarrow ode5, ode4, ode3, ode2, ode1

• for discrete-time systems:

 \Rightarrow discrete (Fixed-step)



SIMULINK Basics Simulation: parameters and solver methods

Configuration Parameters dialog box

🕼 Configuration Parameters: untitled/Configuration				
Select:	Coad from workspace			
Solver Data Import/Export Optimization Join Diagnostics With Ardware Implementation Model Referencing P-Real-Time Workshop	□ Input: [t. u] □ Initial state: xInitial Save to workspace			
	✓ Signal logging: logsout Save options ✓ ✓ Limit data points to last: 1000 Decimation: 1 Format: Array Output options: Refine output ✓ Refine factor:	-		
	<u>QK</u> <u>Cancel</u> <u>Help</u> <u>Apply</u>			

Data Import/Export pane:

- Initialization
- Load data from Workspace
- Write data to Workspace



SIMULINK Basics Simulation: parameters and solver methods

Configuration	Configuration Parame	ters: untitled/Configuration		_	×
<i>Parameters</i> dialog box	Solver Solver Solver Data Import/Export Optimization Diagnostics Sample Time Orversion Conversion Connectivity Model Referencing Model Referencing	Algebraic loop: Minimize algebraic loop: Block priority violation: Min step size violation: Unspecified inheritability of sample time: Solver data inconsistency: Automatic solver parameter selection:	warning warning warning warning none warning		

Diagnostics pane:

- Control of warning and error messages
- Set simulation options





SIMULINK Basics Simulation

Start and stop of simulation

- from SIMULINK window
- from MATLAB Command Window
 - set_param('sys', 'SimulationCommand', 'cmd')
 get_param('sys', 'SimulationStatus')

```
- [t,x,y] = sim('model', timespan, options, ut)
options = simset(property, value, ...)
newopts = simset(oldopts, property, value, ...)
struct = simget('model')
```

Example:

```
[t,x,y] = sim('m1',[],simset(simget('m1'),'Solver','ode23','MaxStep',0.01))
```

Matlab/Simulink



SIMULINK Basics Error handling

Simulation Diagnostics

Viewer

(1) Vie	bspmath w Font Size				
	Message	Source	Reported by	Summary	
۲	Block error	Product1	Simulink	Data type mismatch. Input port 2 of 'bspma	
۲	Block error	Constant	Simulink	Data type mismatch. Output port 1 of 'bspm	
spmath/Product1					
Data type mismatch. Input port 2 of <u>'bspmath/Product1'</u> expects a signal of data type double'. However, it is driven by a signal of data type 'int8'					
				Open Help Close	

- upper part: error information
- lower part: complete text of error message



SIMULINK Basics SIMULINK block library Signal Routing

Block library Signal Routing:

• Date saving

A	A	×A	
Data Store	Data Store	Data Store	
Memory	Read	Write	

• Connect and selection of signals



-

etc.

Matlab/Simulink

SIMULINK Basics Subsystems

Subsystems:

- Structural order of complex models
- Combining blocks of similar function
- hierarchic structuring
- Creation by
 - 1. menue item *Edit/Create Subsystem*
 - 2. block library *Subsystems*



Matlab/Simulink

SIMULINK Basics SIMULINK block library Subsystems

Block library Subsystems:



etc.



Matlab/Simulink

SIMULINK Basics

Conditionally executed subsystems / Masking

Conditionally Executed Subsystems

- Execution controlled by control signal
- comprise *Enable* or *Trigger*

Masking of subsystems

- building a new block from a subsystem
- easier parametrization of complex subsystems
- Setting up of user defined libraries



SIMULINK Basics Masking of subsystems

Example bspmask.mdl



