

Tutorial 20: Programmatically Interface to SimMechanics Multi-Body Toolbox

2016-11-19: Tim C. Lueth, Professor at Technische Universität München, Germany (URL: <http://www.SG-Lib.org>) - Last Change: 2020-08-25

Contents

- [Complete List of all Tutorials with Publishable MATLAB Files of this Solid-Geoemtries Toolbox](#)
- [Motivation for this tutorial: \(Originally SolidGeometry 3.0 required\)](#)
- [2. Creating a new SimMechanics System](#)
- [3. Create two links with length 50 and 80 and one or two mounting holes](#)
- [4. Create SimMechanics models for the four links in different colors](#)
- [5. Create SimMechanics models for the four joint and connect them with the links](#)
- [6. Connect the base frame of link1 to the world coordinate system](#)
- [7. Run the Simulation of the Simulink/SimMechanics diagram for 1 second](#)
- [8. Create a Video of the Simualatin for 5 seconds](#)
- [Final remarks on toolbox version and execution date](#)

Complete List of all Tutorials with Publishable MATLAB Files of this Solid-Geoemtries Toolbox

The following topics are covered an explained in the specific tutorials:

- Tutorial 01: First Steps Using the VLFL-Toolbox for Solid Object Design
- Tutorial 02: Using the VLFL-Toolbox for STL-File Export and Import
- Tutorial 03: Closed 2D Contours and Boolean Operations in 2D
- Tutorial 04: 2½D Design Using Boolean Operators on Closed Polygon Lists (CPL)
- Tutorial 05: Creation, Relative Positioning and Merging of Solid Geometries (SG)
- Tutorial 06: Relative Positioning and Alignment of Solid Geometries (SG)
- Tutorial 07: Rotation of Closed Polygon Lists for Solid Geometry Design
- Tutorial 08: Slicing, Closing, Cutting and Separation of Solid Geometries
- Tutorial 09: Boolean Operations with Solid Geometries
- Tutorial 10: Packaging of Sets of Solid Geometries (SG)
- Tutorial 11: Attaching Coordinates Frames to Create Kinematik Models
- Tutorial 12: Define Robot Kinematics and Detect Collisions
- Tutorial 13: Mounting Faces and Conversion of Blocks into Leightweight-structures
- Tutorial 14: Manipulation Functions for Closed Polygons and Laser Cutting (SVG)
- Tutorial 15: Create a Solid by 2 Closed Polygons
- Tutorial 16: Create Tube-Style Solids by Succeeding Polygons
- Tutorial 17: Filling and Bending of Polygons and Solids
- Tutorial 18: Analyzing and modifying STL files from CSG modeler (Catia)
- Tutorial 19: Creating drawing templates and dimensioning from polygon lines
- Tutorial 20: Programmatically Interface to SimMechanics Multi-Body Toolbox
- Tutorial 21: Programmatically Convert Joints into Drives (SimMechanics)
- Tutorial 22: Adding Simulink Signals to Record Frame Movements
- Tutorial 23: Automatic Creation of a Missing Link and 3D Print of a Complete Model
- Tutorial 24: Automatic Creation of a Joint Limitations
- Tutorial 25: Automatic Creation of Video Titels, Endtitels and Textpages
- Tutorial 26: Create Mechanisms using Universal Planar Links
- Tutorial 27: Fourbar-Linkage: 2 Pose Syntheses and Linkage Export for 3D Printing
- Tutorial 28: Fourbar-Linkage: 3 Pose Syntheses and Linkage Export for 3D Printing
- Tutorial 29: Create a multi body simulation using several mass points
- Tutorial 30: Creating graphical drawings using point, lines, surfaces, frames etc.
- Tutorial 31: Importing 3D Medical DICOM Image Data and converting into 3D Solids
- Tutorial 32: Exchanging Data with a FileMaker Database
- Tutorial 33: Using a Round-Robin realtime multi-tasking system
- Tutorial 34: 2D Projection Images and Camera Coordinate System Reconstruction
- Tutorial 35: Creation of Kinematic Chains and Robot Structures
- Tutorial 36: Creating a Patient-Individual Arm-Skin Protector-Shell
- Tutorial 37: Dimensioning of STL Files and Surface Data

- Tutorial 38: Some more solid geometry modelling function
- Tutorial 39: HEBO Modules robot design
- Tutorial 40: JACO Robot Simulation and Control
- Tutorial 41: Inserting Blades, Cuts and Joints into Solid Geometries
- Tutorial 42: Performing FEM Stress and Displacement Analysis and Structural Optimization of Solids
- Tutorial 43: Performing FEM Structural Optimization (CAO) and Topological Optimization (SKO) of Solids
- Tutorial 44: Creation of solids and kinematics from 3D curves and transformation matrices
- Tutorial 45: Creation of Solids using the SG-Coder - SGofCPLcommand
- Tutorial 46: Creating Fischertechnik compatible gear boxes using SGofCPLcommand
- Tutorial 47: Creating four-joints by 3 pose synthesis
- Tutorial 52: CPL Buffers and cw/ccw Orientation
- Tutorial 53: SKOL - Soft Kill Option for Large Displacement by Yilun Sun
- Tutorial 54: Automated Design of Precision Joints by Screws or Ball Bearings
- Tutorial 54: Processing Stacks of Slices = CVLz
- Tutorial 55: Automated Design of Manipulators with Screws or Ball Bearing
- Tutorial 56: Checking Functions for Solids
- Tutorial 57: Processing Stacks of Slices = CVLz

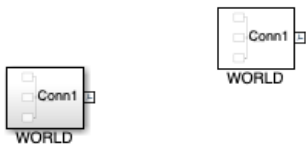
Motivation for this tutorial: (Originally SolidGeometry 3.0 required)

```
%make shure that wd='/Volumes/LUETH-WIN/WIN AIM Matlab Libraries/SolidGeometry-Code'; is used if not in toolbox-test
```

2. Creating a new SimMechanics System

```
smbNewSystem ('SG_LIB_EXP_20');           % Creates the mechsims diagramm
smbDrawNow;
```

Creating temporary directory '/Users/timlueth/Desktop/tmp_SG_LIB_EXP_20/'

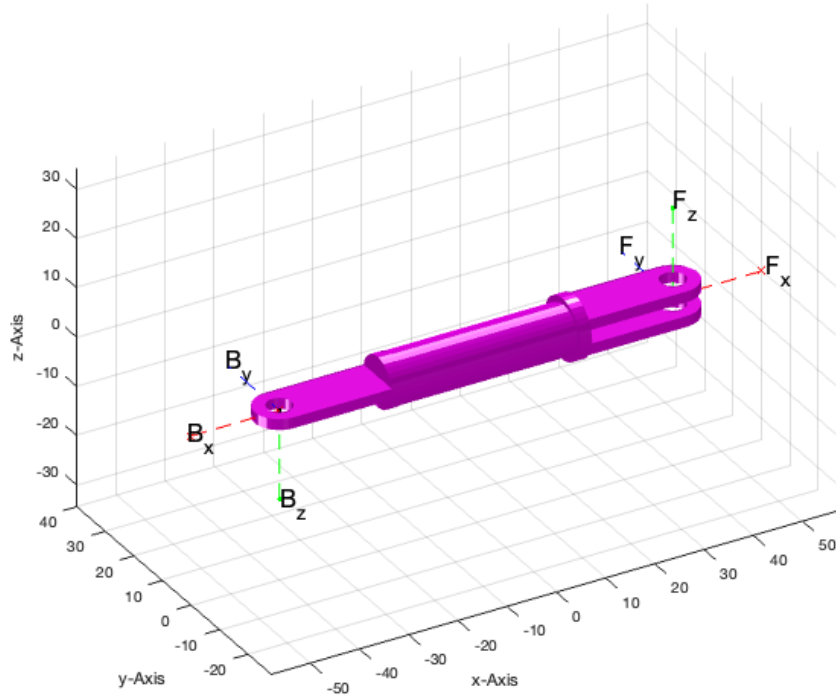


3. Create two links with length 50 and 80 and one or two mounting holes

```
SG1=SGmodelLink(80,'',1,2);           % Creates a long rod with flange
SG2=SGmodelLink(50,'',1,2);          % Creates a short rod with flange
```

```
SGfigure(SG1); view(-30,30);
```

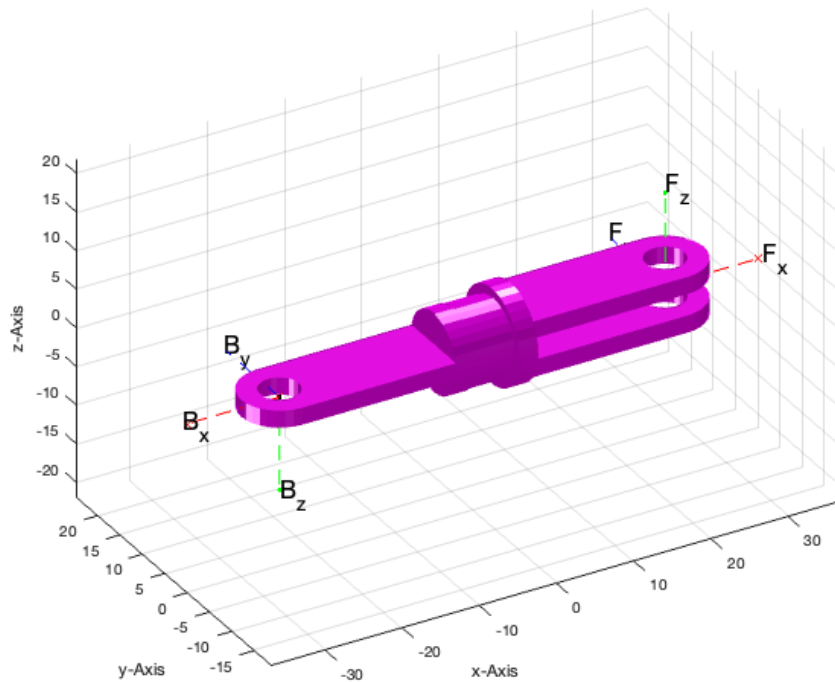
publishSGPDF: 2023-10-03 07:40:02



```
SGfigure(SG2); view(-30,30);
```



publishSGPDF: 2023-10-03 07:40:03



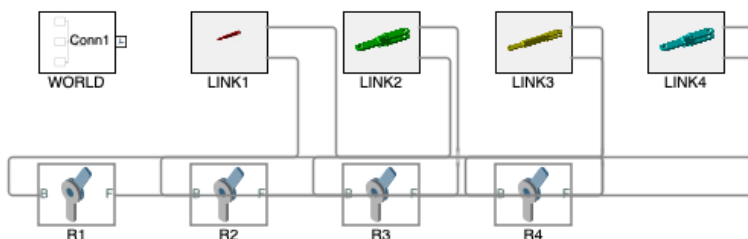
4. Create SimMechanics models for the four links in different colors

```
smbCreateSG (SG1, 'LINK1', 'r');           % Add long rod as LINK1
smbCreateSG (SG2, 'LINK2', 'g');           % Add short rod as LINK2
smbCreateSG (SG1, 'LINK3', 'y');           % Add long rod as LINK3
smbCreateSG (SG2, 'LINK4', 'c');           % Add short rod as LINK4
smbDrawNow;
```



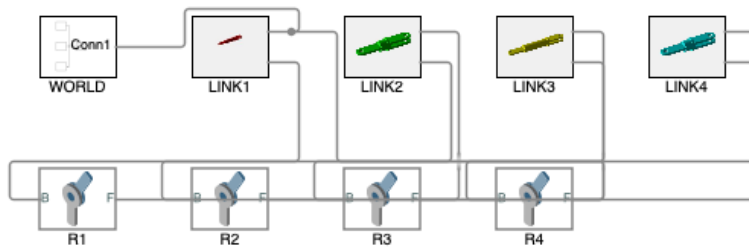
5. Create SimMechanics models for the four joint and connect them with the links

```
smbCreateJoint ('R', 'R1', 'LINK1.F', 'LINK2.B'); % Add a RR Joint
smbCreateJoint ('R', 'R2', 'LINK2.F', 'LINK3.B'); % Add a RR Joint
smbCreateJoint ('R', 'R3', 'LINK3.F', 'LINK4.B'); % Add a RR Joint
smbCreateJoint ('R', 'R4', 'LINK4.F', 'LINK1.B'); % Add a RR Joint
smbDrawNow;
```



6. Connect the base frame of link1 to the world coordinate system

```
smbCreateConnection('WORLD.ORIGIN','LINK1.B'); % Connect Linkage to World Frame
smbDrawNow;
```



7. Run the Simulation of the Simulink/SimMechanics diagram for 1 second

```
smbSimulate(1); % Simulate for 1 second
return
```

8. Create a Video of the Simulatin for 5 seconds

```
close all;
smbVideoSimulation(5); % Show a 5 seconds video
```

Final remarks on toolbox version and execution date

```
VLFLlicense
close all
```

- Tim Lueth, tested and compiled on OSX 10.11.6 with Matlab 2016 on 2016-12-09
- _____, executed and published on 64 Bit PC using Windows with Matlab 2015a on 2015-xx-xx_