Tutorial 24: Automatic Creation of a Joint Limitations

2016-12-25: Tim C. Lueth, MIMED - Technische Universität München, Germany (URL: http://www.mimed.de) - Last Change: 2017-07-07

Contents

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: Collection of Ideas for Tutorials
- Tutorial 36: Creating a Patient-Individual Arm-Skin Protector-Shell

Motivation for this tutorial: (Originally SolidGeometry 3.2 required)

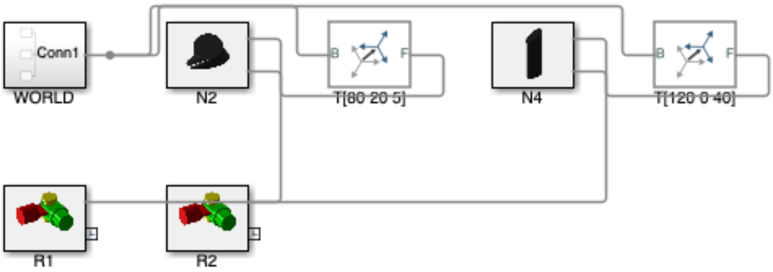
2. Open a system and create several fixed nodes and attach revolute joints

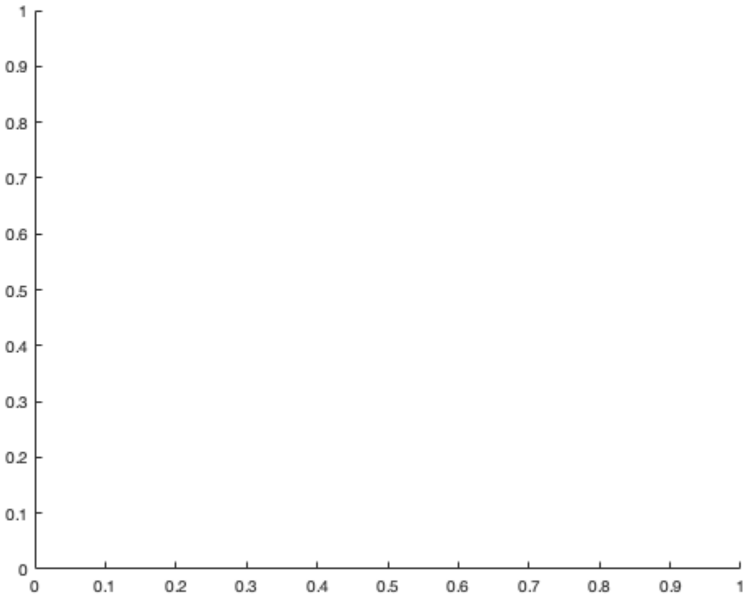
function VLFL_EXP24

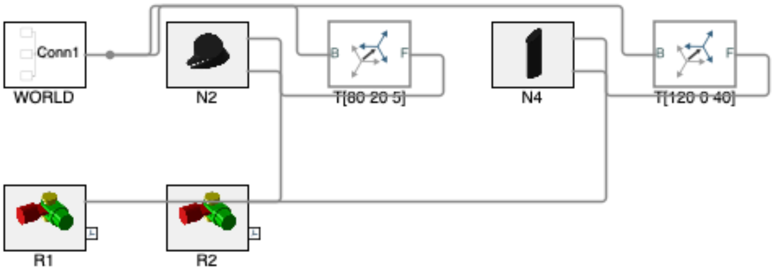
```
smbsys='SG_LIB_EXP_24';
smbNewSystem (smbsys);

smbCreateSGNode ([80  20  5],'N2','',rot(0,0,pi/3));
smbCreateSGNode ([120 0 40],'N4','',rot(0,-pi/8,0));
A=SGmodelJoint('R',pi/2);
smbCreateSGJoint('R','R1', A,'N4.F');
smbCreateSGJoint('R','R2',A,'N2.F');
smbDrawNow;
```

```
Creating temporary directory '/Users/timlueth/Desktop/tmp_SG_LIB_EXP_24/'
ans =
     1     0     0
     0     0     1
     0    -1     0
ans =
     1     0     0
     0     0     1
     0    -1     0
```







3. Create a cylindric joint from two solids an attach it to revolute joint

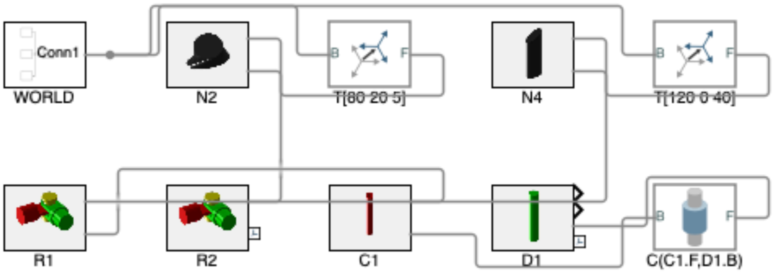
```
Ro=5;
Ri=3;
slot=0.3;

C1=SGofCPLz([PLcircle(Ro);NaN NaN;PLcircle(Ri+slot)],30);
% C1=SGTset(C1,'B',TofSG(C1,'bottom','roty',pi));
C1=SGTset(C1,'B',TofSG(C1,'incenter','right',-1,'roty',pi/2,'5.1'));
C1=SGTset(C1,'F',TofSG(C1,'bottom','5.1'));
smbCreateSG(C1,'C1','r','R1_M');
D1=SGofCPLz(PLcircle(3),30);
D1=SGTset(D1,'B',TofSG(D1,'incenter'));
D1=SGTset(D1,'F',TofSG(D1,'top'));
```

```
ans =
     1     0     0
     0     0     1
     0    -1     0
ans =
     1     0     0
     0     0     1
     0    -1     0
```

4. Attach two frame sensor to record the movement of the falling cylinder

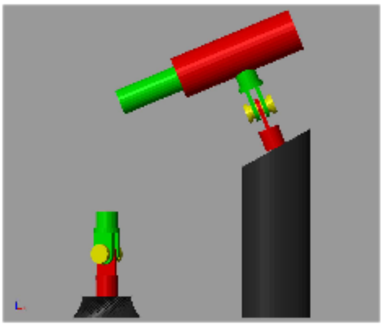
```
smbCreateSG(D1,'D1','g');
smbCreateConnection('C1.F','D1.B','C');
smbAddFrameSensor('R2_M.F');
smbAddFrameSensor('D1.F');
smbDrawNow;
```



5. Show the Simulation

```
simOut=smbSimulate(0.1);
smbVideoSimulation(4);
```

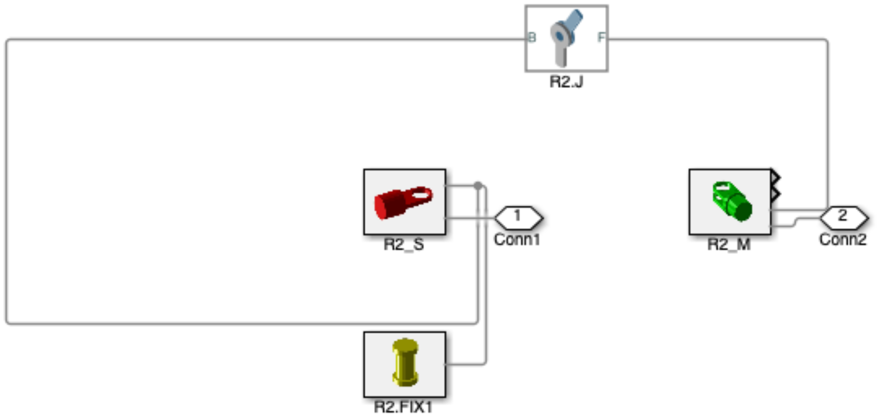
....

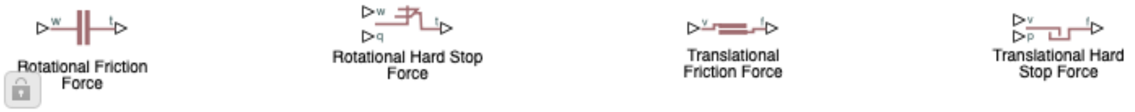


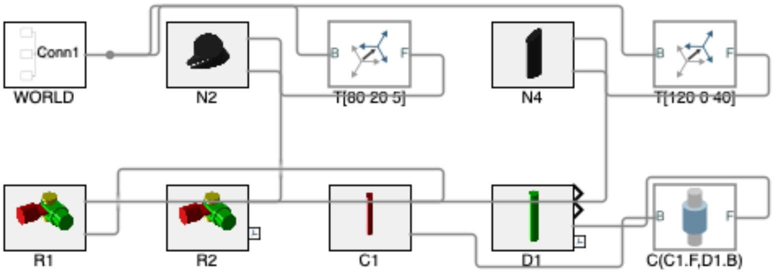
6. Install additional block funktion for joint restrictions

```
smbPSLibInstall
open_system(smbPSBlockname);
open_system(smbsys,'tab');
open_system(smbWhich('R2'),'tab');smbDrawNow;
```

```
Create /Users/timlueth/Desktop/tmp_SG_LIB_EXP_24/+mechPS_Tim_Lueth/PS_force_friction_rot.ssc
Create /Users/timlueth/Desktop/tmp_SG_LIB_EXP_24/+mechPS_Tim_Lueth/PS_force_friction_rot.svg
Create /Users/timlueth/Desktop/tmp_SG_LIB_EXP_24/+mechPS_Tim_Lueth/PS_force_hardstop_rot.ssc
Create /Users/timlueth/Desktop/tmp_SG_LIB_EXP_24/+mechPS_Tim_Lueth/PS_force_hardstop_rot.svg
Create /Users/timlueth/Desktop/tmp_SG_LIB_EXP_24/+mechPS_Tim_Lueth/PS_force_friction_trans.ssc
Create /Users/timlueth/Desktop/tmp_SG_LIB_EXP_24/+mechPS_Tim_Lueth/PS_force_friction_trans.svg
Create /Users/timlueth/Desktop/tmp_SG_LIB_EXP_24/+mechPS_Tim_Lueth/PS_force_hardstop_trans.ssc
Create /Users/timlueth/Desktop/tmp_SG_LIB_EXP_24/+mechPS_Tim_Lueth/PS_force_hardstop_trans.svg
Generating Simulink library 'mechPS_Tim_Lueth_lib' in the current directory '/Users/timlueth/Desktop/tmp_SG_LIB_EXP_24' ...
```

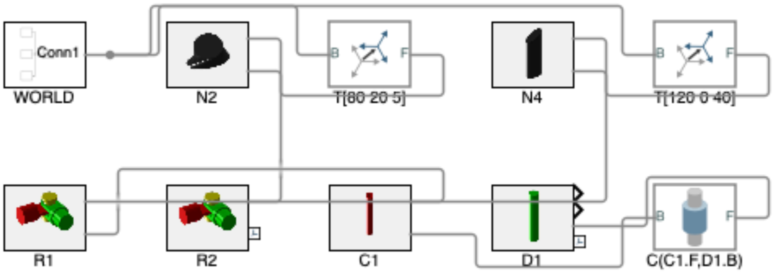



w
f
Rotational Friction Force
w
q
f
Rotational Hard Stop Force
v
f
Translational Friction Force
v
p
f
Translational Hard Stop Force



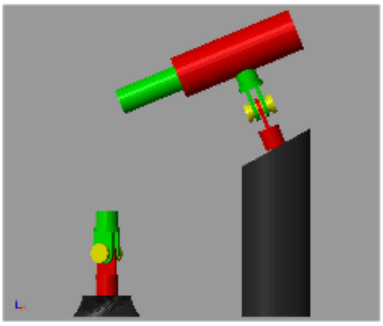
7. Create a stopp joint and copy all connections of an existing joint

```
smbCreateStopJointR ('R2stop.J',[-pi/2 +pi/2]);
smbCopyConnections  ('R2.J','R2stop.J');
smbDrawNow;
```



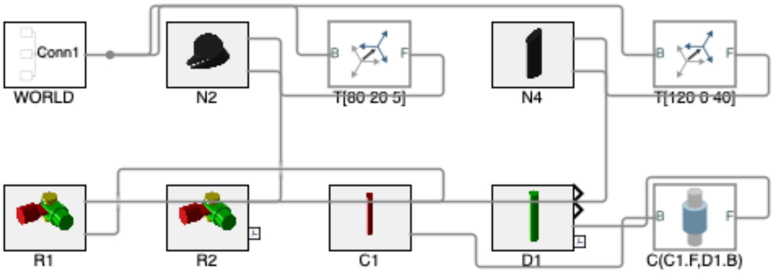
```
smbVideoSimulation(4);
```

```
.........
```

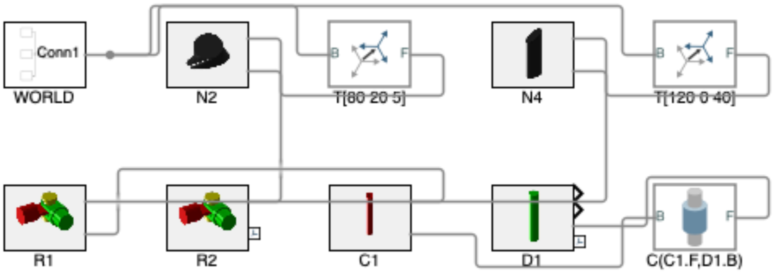


8. Create a stopp joint and replace an existing joint

```
delete_block(smbWhich('R2stop.J'));
smbDeleteUnconnectedLines;
smbDrawNow;
```



```
smbCreateStopJointR ('R2new.J',[-pi/2 +pi/2]);
smbCopyConnections  ('R2.J','R2new.J','replace');
smbDrawNow;
```



9. Final Remarks

```
VLFLlicense
```

```
This VLFL-Lib, Rel.  (2023-Oct-03), is for limited non commercial educational use only!
Licensee: Tim Lueth (Development Version)!
Please contact Tim Lueth, Professor at TU Munich, Germany!
WARNING: This VLFL-Lib (Rel. ) license will exceed at 06-Jul-2078 07:47:20!
Executed 03-Oct-2023 07:47:22 by 'timlueth' on a MACI64 using Mac OSX 13.6 | R2023a Update 5 | SG-Lib 5.4
========================================= Used Matlab products: =======================================
distrib_computing_toolbox
fixed_point_toolbox
map_toolbox
matlab
simmechanics
simscape
simulink
=========================================================================================================
```

Published with MATLAB® R2023a