

## Tutorial 56: Checking Functions for Solids

2020-08-22: Tim C. Lueth, Professor at Technische Universität München, Germany (URL: <http://www.SG-Lib.org>) - Last Change: 2020-08-22

### Contents

---

- [Complete List of all Tutorials with Publishable MATLAB Files of this Solid-Geometries Toolbox](#)
- [Motivation for this tutorial: \(Originally SolidGeometry 4.9 required\)](#)
- [UNDER DEVELOPMENT NOT](#)
- [Final Remarks](#)

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

---

The following topics are covered and 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 Coordinate Frames to Create Kinematic Models
- Tutorial 12: Define Robot Kinematics and Detect Collisions
- Tutorial 13: Mounting Faces and Conversion of Blocks into Lightweight-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 Titles, Endtitles 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 55: Automated Design of Manipulators with Screws or Ball Bearing
- Tutorial 56: Checking Functions for Solids

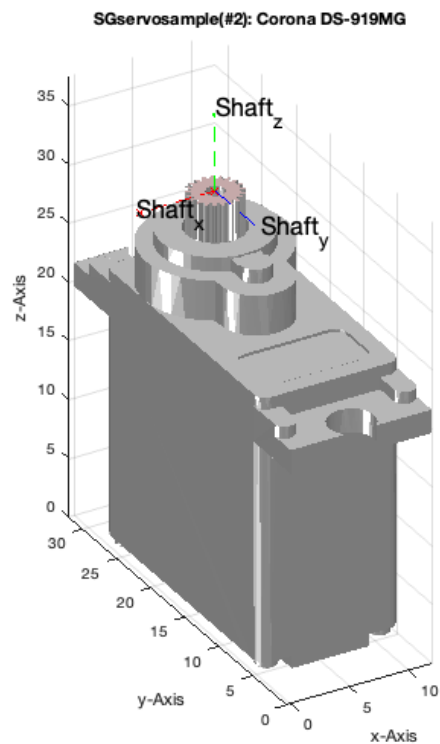
### Motivation for this tutorial: (Originally SolidGeometry 4.9 required)

In this tutorial different important functions for checking solid geometries are called and explained

```
% function VLFL_EXP56
```

```
SGservosample(2); SG=ans;
```

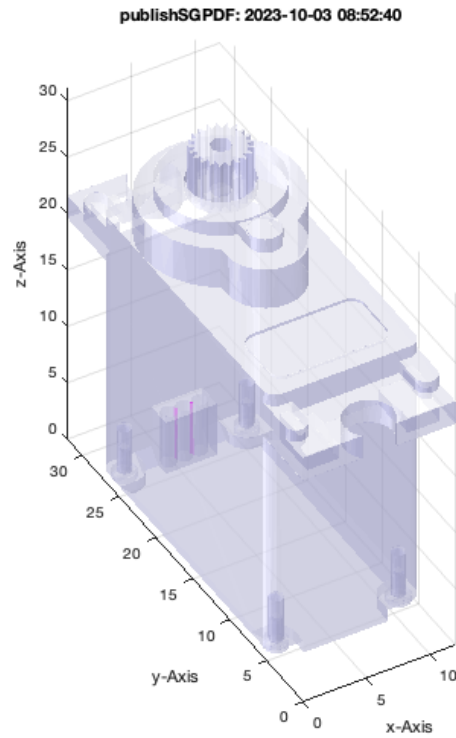
```
SGservosample: Non-manifold edges of this solid: 4
```



```
SGcheckmeshlab(SG)
```

```
publish
OPEN BOUNDARY:
0 open boundary groups
0 open boundary edges
NON MANIFOLD EDGES:
4 non manifold edges
8 faces over non manifold edges
NON MANIFOLD VERTICES:
7 independent surfaces
0 non manifold vertices
0 faces over non manifold vertices
SELF INTERSECTING FACETS:
```

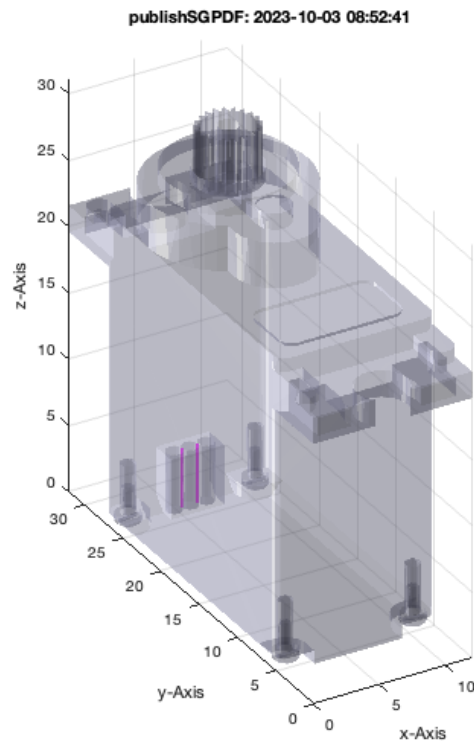
```
too many facets (4000>300) for a self-intersection check
ans =
    4     8     0     0
```



GREEN = Open boundaries: 0 MAGENATA = Non manifold edges: 0 RED = Non manifold vertices: 0
--

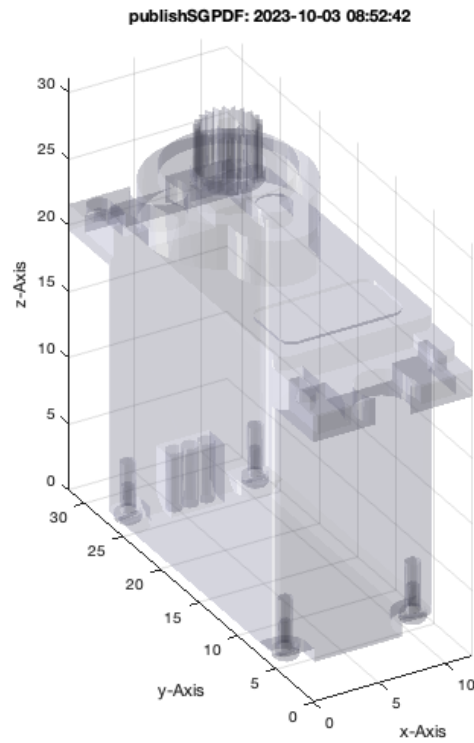
**SGchecknonmanifoldedges (SG)**

```
SGchecknonmanifoldedges:
NON MANIFOLD EDGES:
2 non manifold edges
8 faces over non manifold edges
```



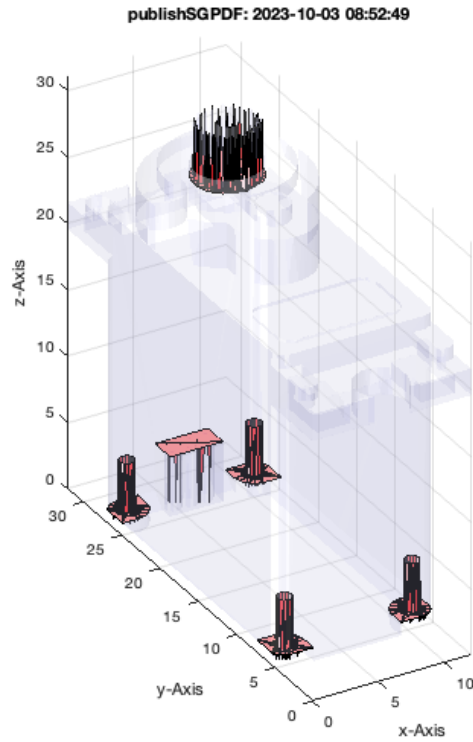
SGchecknonmanifoldvertices(SG)

```
SGchecknonmanifoldvertices:
NON MANIFOLD VERTICES:
0 non manifold vertices
0 faces over non manifold vertices
```



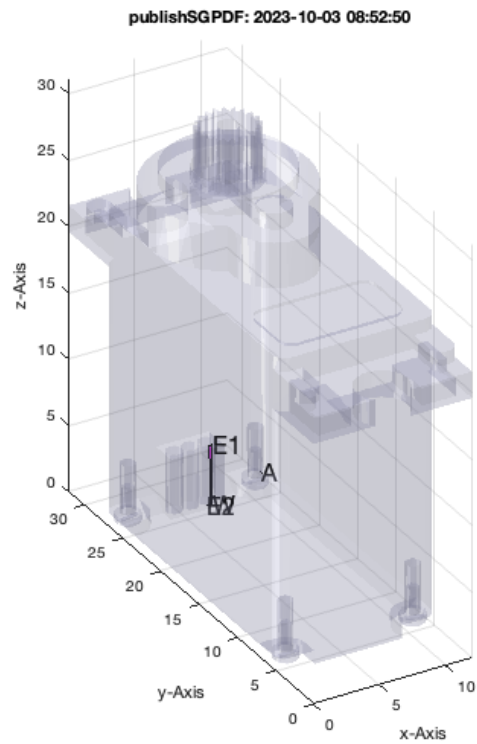
## SGcheckselfintersection(SG)

```
SGcheckselfintersection:  
SELF INTERSECTING FACETS:  
638 self intersecting facets  
ans =  
    638
```



## SGchecktrianglegeometry(SG)

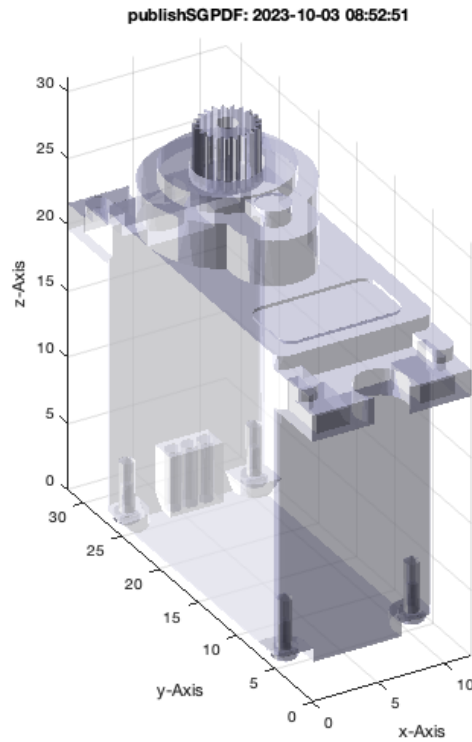
```
SGchecktrianglegeometry: Smallest edges(2): 0.02mm. Smallest angles(1): 2.9e-09rad. Smallest areas(1): 0.000qmm.  
SGchecktrianglegeometry: Number of edges<0.1mm= 3. Number of angles<0.0001rad= 3. Number of areas<.005qmm= 1.  
ans =  
    2.8954e-09
```



```
SGcheckboundaryedges (SG)
```

```
%
```

```
SGcheckboundaryedges:  
BOUNDARY:  
0 open boundary groups  
0 open boundary meshes  
0 unconnected edges  
0 open boundary edges  
ans =  
[]
```



**UNDER DEVELOPMENT NOT**

SGcheckfacelist(SG); % Not test yet SGcheckvertexaccuracy(SG);

**Final Remarks**

```
close all
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 08:52:52!  
 Executed 03-Oct-2023 08:52:54 by 'timlueth' on a MACI64 using Mac OSX 13.6 | R2023a Update 5 | SG-Lib 5.4  
 ===== Used Matlab products: =====  
 database\_toolbox  
 distrib\_computing\_toolbox  
 fixed\_point\_toolbox  
 image\_toolbox  
 map\_toolbox  
 matlab  
 optimization\_toolbox  
 pde\_toolbox  
 simmechanics  
 simscape  
 simulink  
 =====

Published with MATLAB® R2023a