Tutorial 07: Rotation of Closed Polygon Lists for Solid Geometry Design

2014-11-26: 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
- Motivation for this tutorial: (Originally SolidGeometry 2.0 required)
- 2. List of functions used in this example
- 3. Rotation of closed polygon lists (CPL)
- 4. Creating spheres by rotating half-circles
- 5. Creating embedded contours
- 6. Rotate Contours around the z-axis
- 7. Samples of 3D Design
- 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: Collection of Ideas for Tutorials
- Tutorial 36: Creating a Patient-Individual Arm-Skin Protector-Shell

Motivation for this tutorial: (Originally SolidGeometry 2.0 required)

2. List of functions used in this example

As we learned in example 2 and 4, it is possible to extrude a planar point list (PL) or closed polygon (CPL) list into a 2.5D solid geometry. Now, we will rotate a CPL around th z-axis. In this case, we consider the CPL or the PL always as a x/z-list. Using closed polygon lists, we have to remember that before extruding them or rotating them it is necessary to guarantee that the outer contour has a counter-clockwise order (ccw).

In this example, some new functions are introduced:

- CPLplot to draw the closed polygon list in the x/y plane.
- PLELofCPL to draw the direction, starting point and end point.
- CPLuniteCPL to unite several CPL into one and adapt their original directions.
- SGofCPLrot to rotate a contour around the z-axis

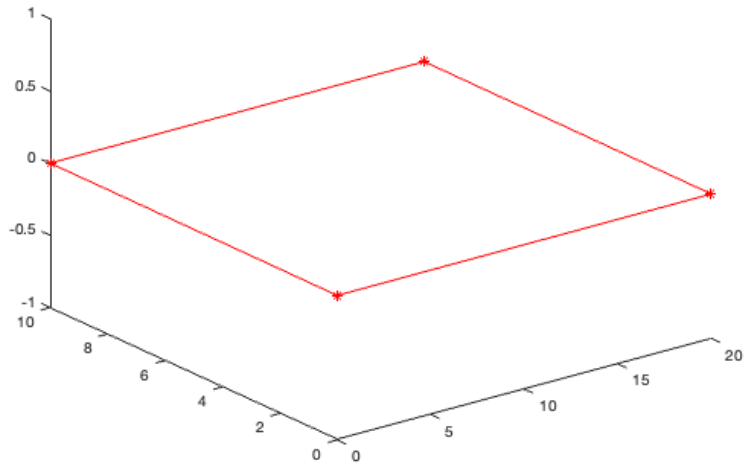
3. Rotation of closed polygon lists (CPL)

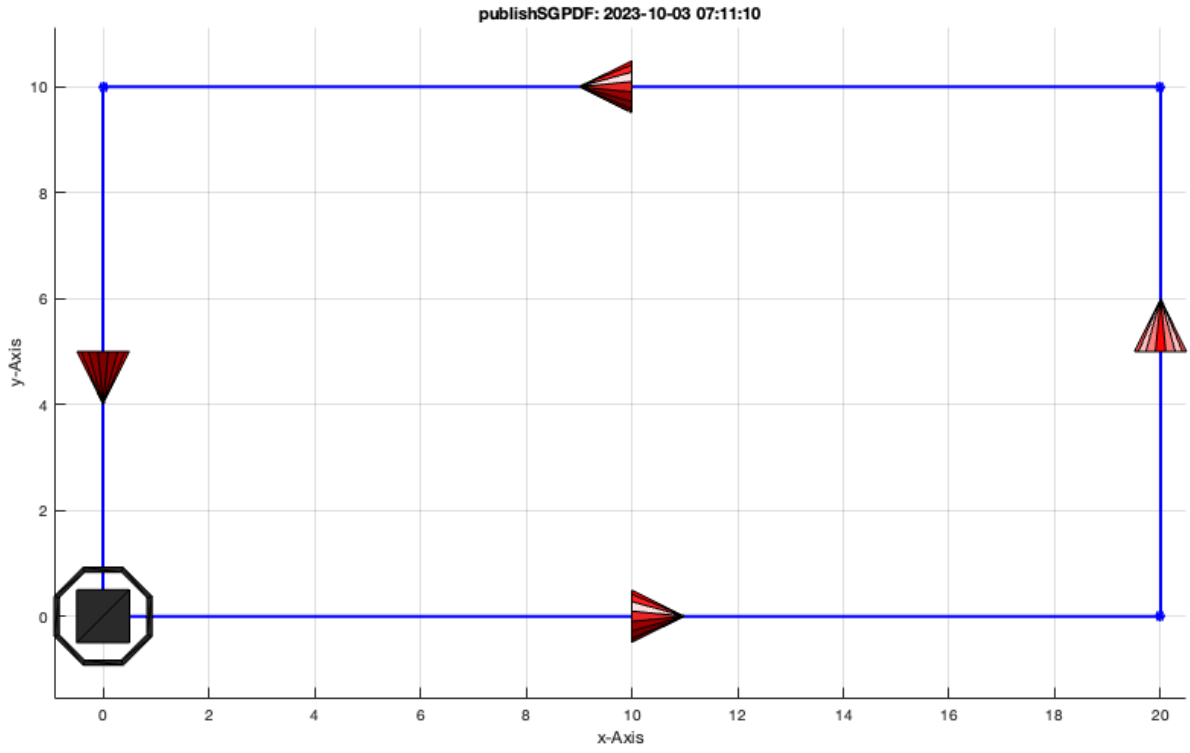
For the rotation of a simple contour we use the following functions

- **CPLplot** to draw the closed polygon list in the x/y plane.
- **PLELofCPL** to draw the direction, starting point and end point.
- **SGofCPLrot** to rotate a contour around the z-axis

Exercise: Create a simple point list that touches the y-axis

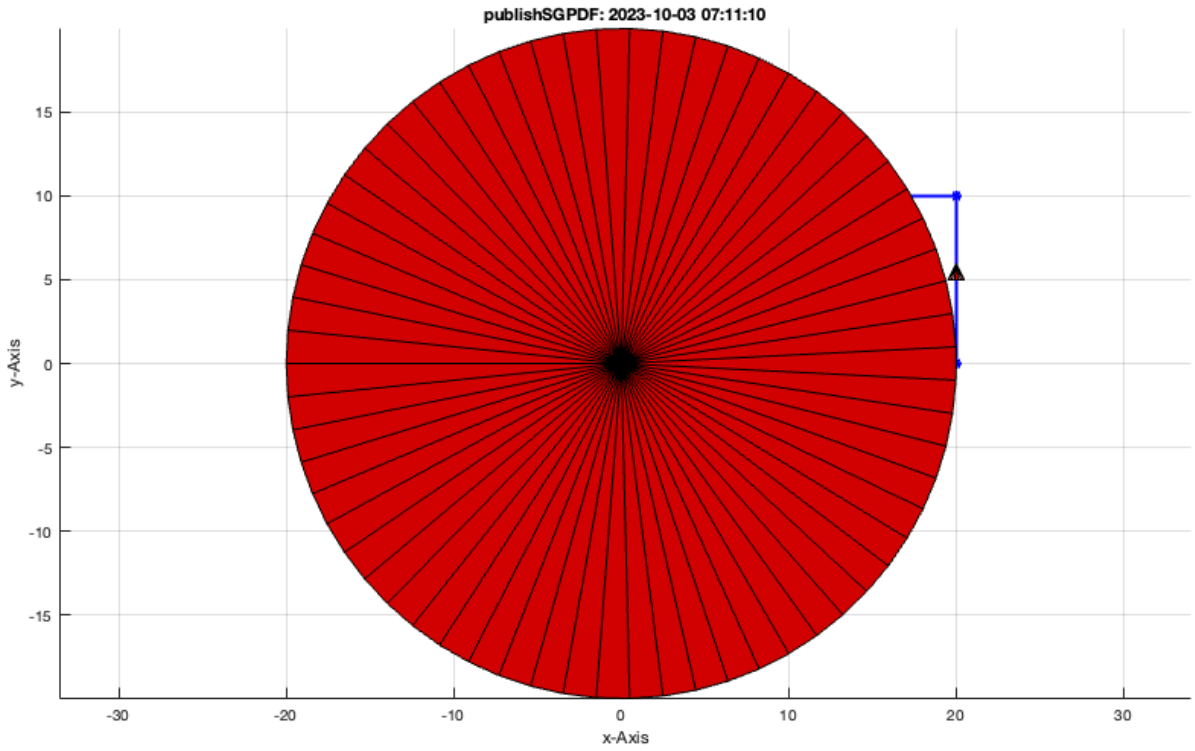
```
close all;
CPL=[0 0; 20 0; 20 10; 0 10];       % Create a simple rectangle (ccw)
CPLplot(CPL);                       % plot the rectangle
PLELofCPL(CPL);                     % show edges and directions
```





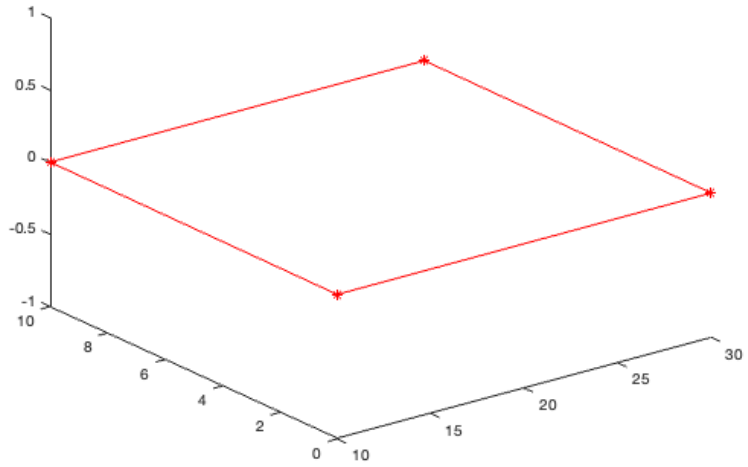
Exercise: Rotate the point list around the z-axis to create a cylinder

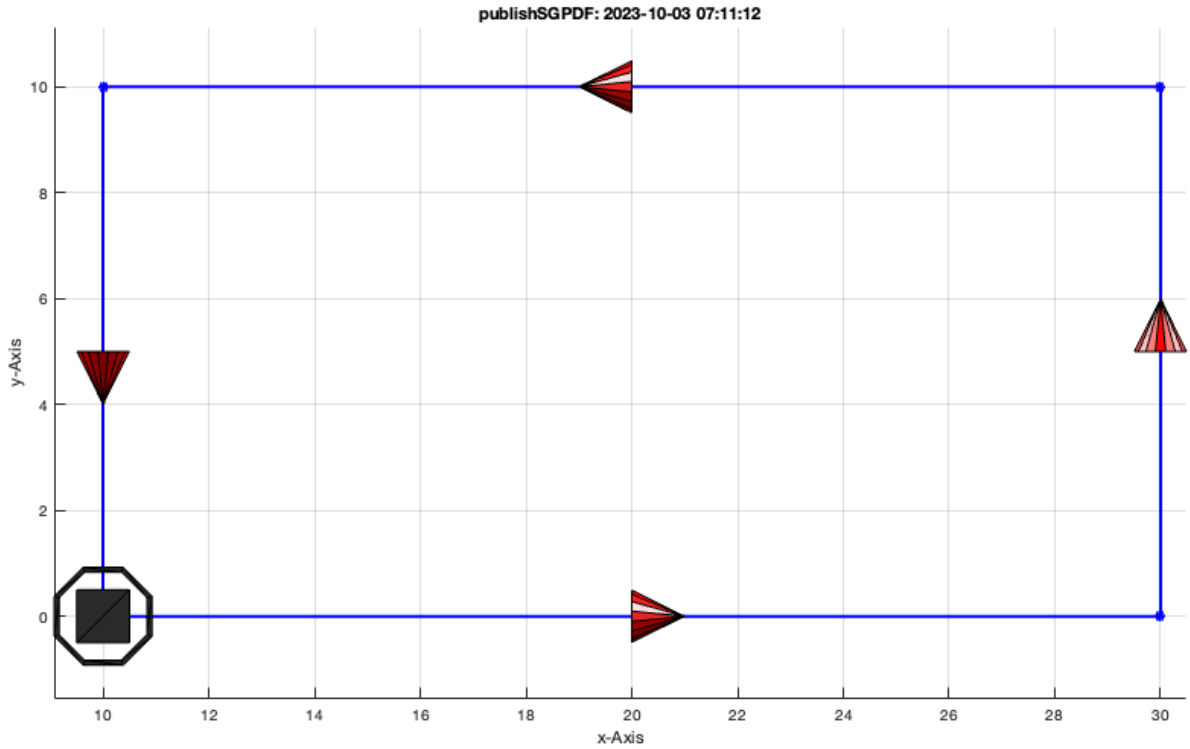
```
SG=SGofCPLrot(CPL);                % Solid contour rotation
SGplot(SG);                        % show the solid
```



Exercise: Create a simple point list with distance to the y-axis

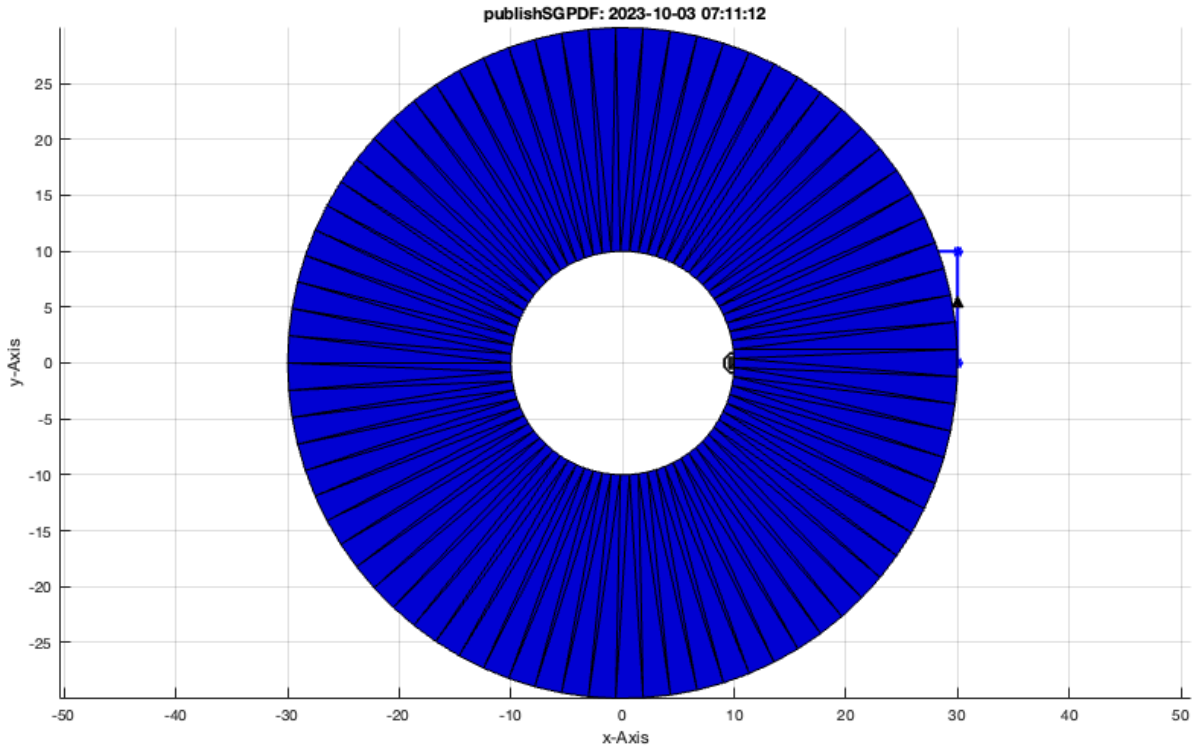
```
close all;
CPL=[0 0; 20 0; 20 10; 0 10];       % Create a simple rectangle (ccw)
CPL(:,1)=CPL(:,1)+10;               % shift by 1 on the x-axis
CPLplot(CPL);                       % plot the rectangle
PLELofCPL(CPL);                     % show edges and directions
```





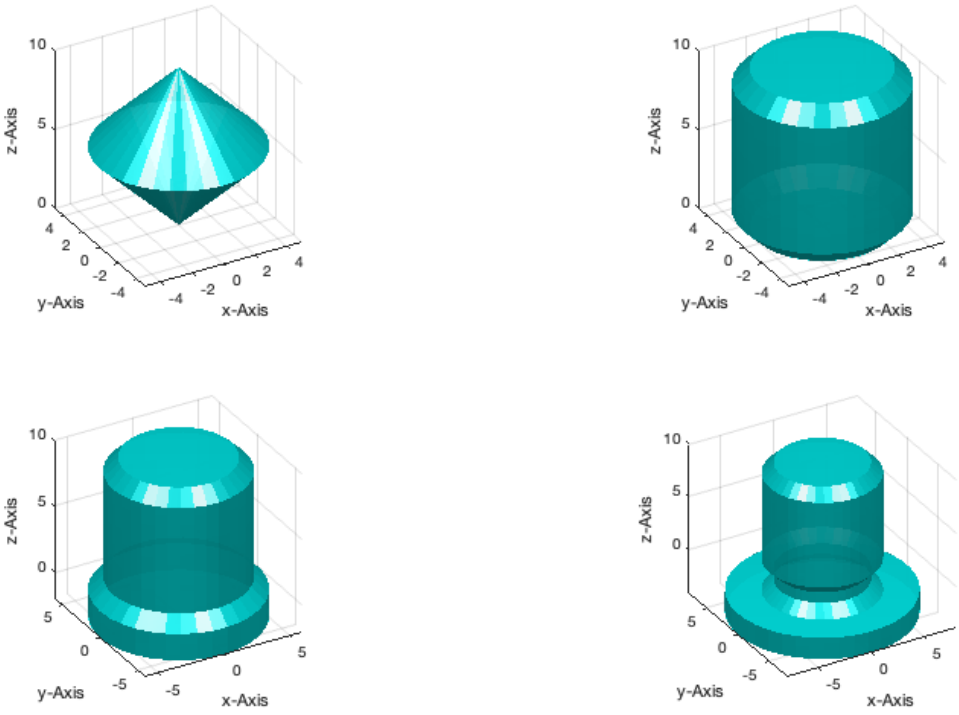
Exercise: Rotate the point list around the z-axis to create a hollow cylinder

```
SG=SGofCPLrot(CPL);                 % Solid contour rotation
SGplot(SG,'b');
```



Exercise: Some other examples for massiv rotational symetric solids

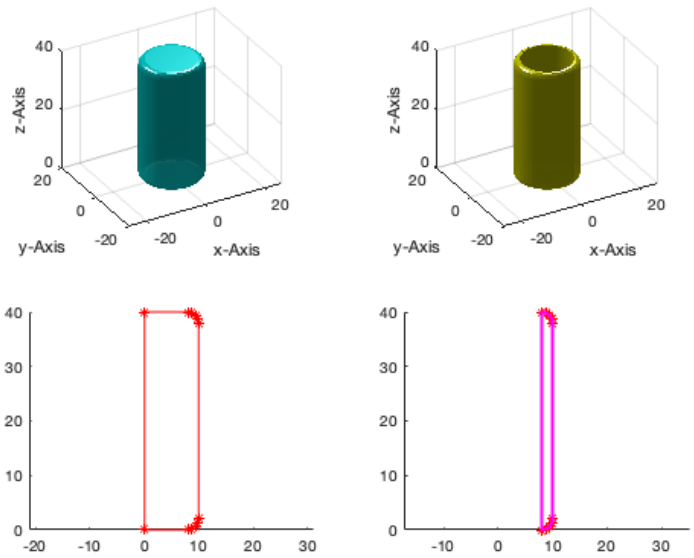
```
SG=SGofCPLrot([0 0; 5 5; 0 10]);                  % Solid contour rotation
subplot(2,2,1); view (-30,30); SGplot(SG,'c'); VLFLplotlight (1,0.9);
SG=SGofCPLrot([0 0; 4 0;  5 1; 5 9; 4 10; 0 10;]); % Solid contour rotation
subplot(2,2,2); view (-30,30); SGplot(SG,'c'); VLFLplotlight (1,0.9);
SG=SGofCPLrot([0 -2; 6 -2; 6 0;  5 1; 5 9; 4 10; 0 10;]); % Solid contour rotation
subplot(2,2,3); view (-30,30); SGplot(SG,'c'); VLFLplotlight (1,0.9);
SG=SGofCPLrot([0 -4; 8 -4; 8 -2; 5 -2; 4 -1; 4 0;  5 1; 5 9; 4 10; 0 10;]); % Solid contour rotation
subplot(2,2,4); view (-30,30); SGplot(SG,'c'); VLFLplotlight (1,0.9);
```



The warnings 'Removed n(m) facets' can be ignored. These warning appear if a part of the contour touches or crosses the x=0 line (y-axis).

Exercise: Creating a bold and a sleeve

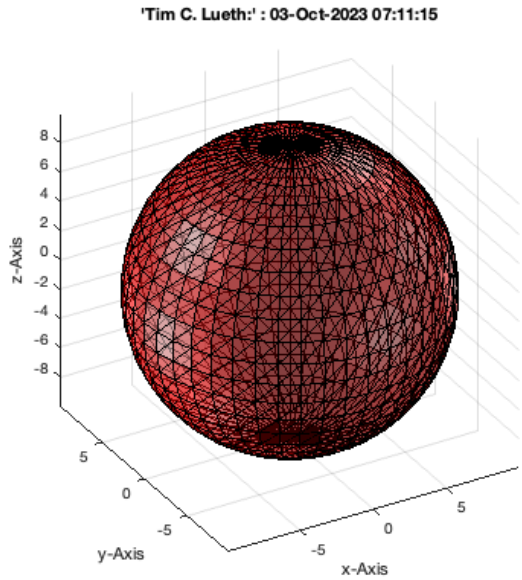
```
closeall;
r=2; H=40; R=10;

PL=PLcircseg (r,[],0,pi/2);  CPL=PLtransP(PL,[R-r,H-r]);
PL=PLcircseg (r,[],-pi/2,0); CPL=[CPL;0 H; 0 0;PLtransP(PL,[R-r,r])];
SG=SGofCPLrot(CPL);                % Solid contour rotation

subplot(2,2,1); SGplot(SG,'c'); VLFLplotlight (1,0.9); view (-30,30);
subplot(2,2,3); [XPL,EL]=PLELofCPL (CPL); PLELplot(XPL,EL); view(0,90);axis equal ;

PL=PLcircseg (r,[],0,pi/2);  CPL=PLtransP(PL,[R-r,H-r]);
PL=PLcircseg (r,[],-pi/2,0); CPL=[CPL;PLtransP(PL,[R-r,r])];
SG=SGofCPLrot(CPL);
subplot(2,2,2); SGplot(SG,'y'); VLFLplotlight (1,0.9); view (-30,30);
subplot(2,2,4); [XPL,EL]=PLELofCPL(CPL);  PLELplot(XPL,EL); view(0,90); CPLplot(CPL,'m-',2); axis equal;
```



4. Creating spheres by rotating half-circles

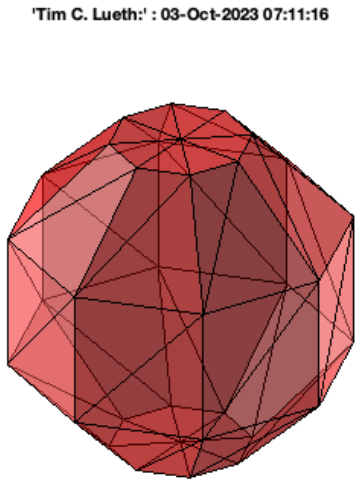
Exercise: Creating a full sphere

```
close all;
PL=PLcircle(10);

VLFLfigure; view(-30,30); grid on;
SG=SGofCPLrot(PL);
SGplot(SG); VLFLplotlight (0,0.5);
```



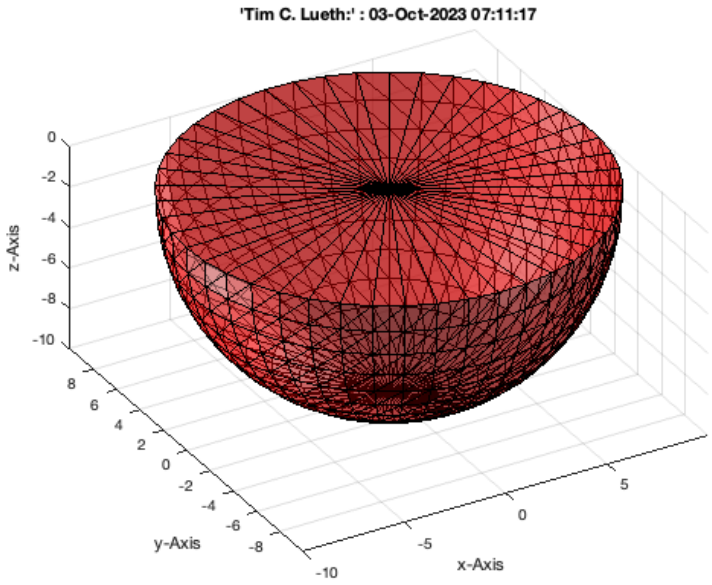
Exercise: Creating a 8 by 8 sphere

```
close all;
PL=PLcircle(10,8);

VLFLfigure; view(-30,30); grid on;
SG=SGofCPLrot(PL,8);
SGplot(SG); VLFLplotlight (0,0.5);
```



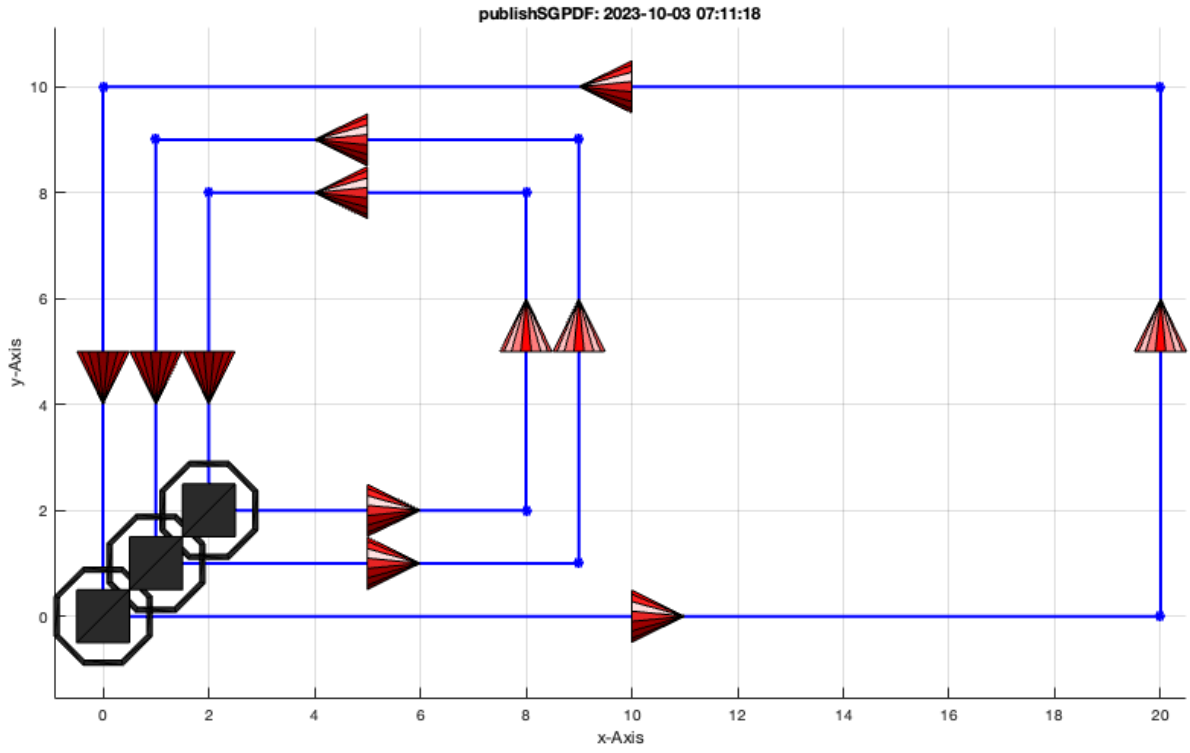
Exercise: Creating a half sphere

```
close all;
PL=[PLcircseg(10,[],-pi/2,0); 0 0];

VLFLfigure; view(-30,30); grid on;
SG=SGofCPLrot(PL);
SGplot(SG); VLFLplotlight (0,0.5);
```

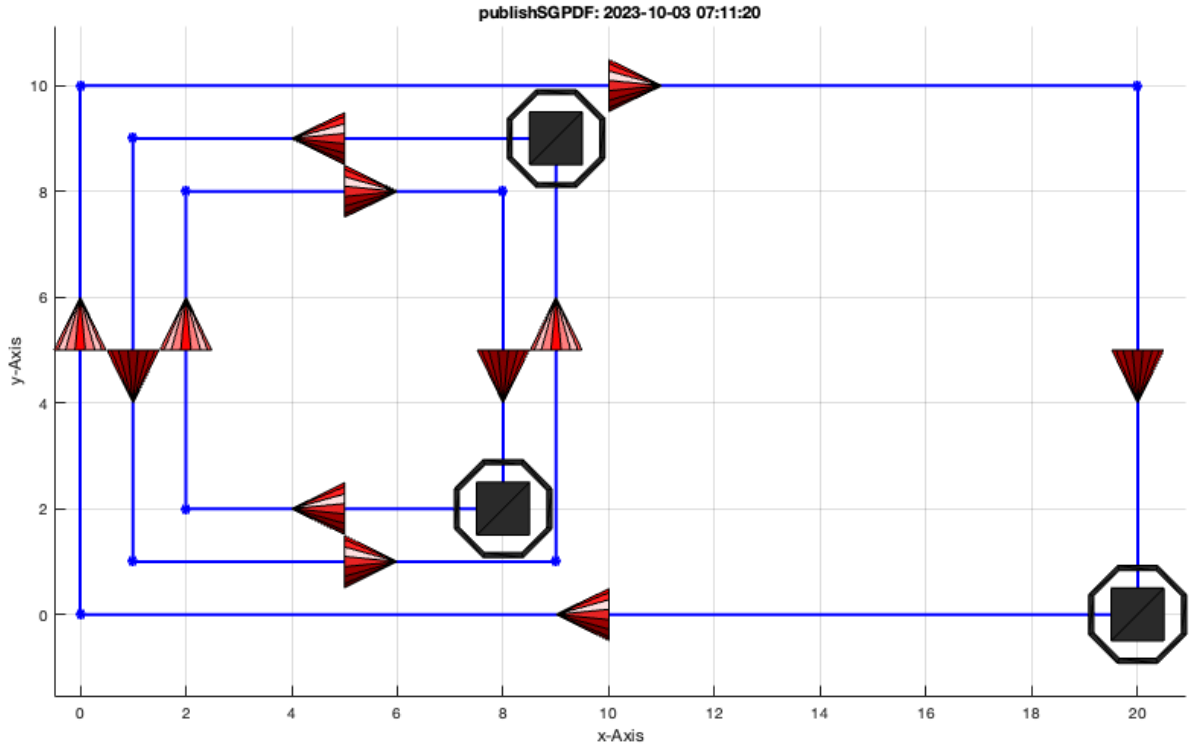


5. Creating embedded contours

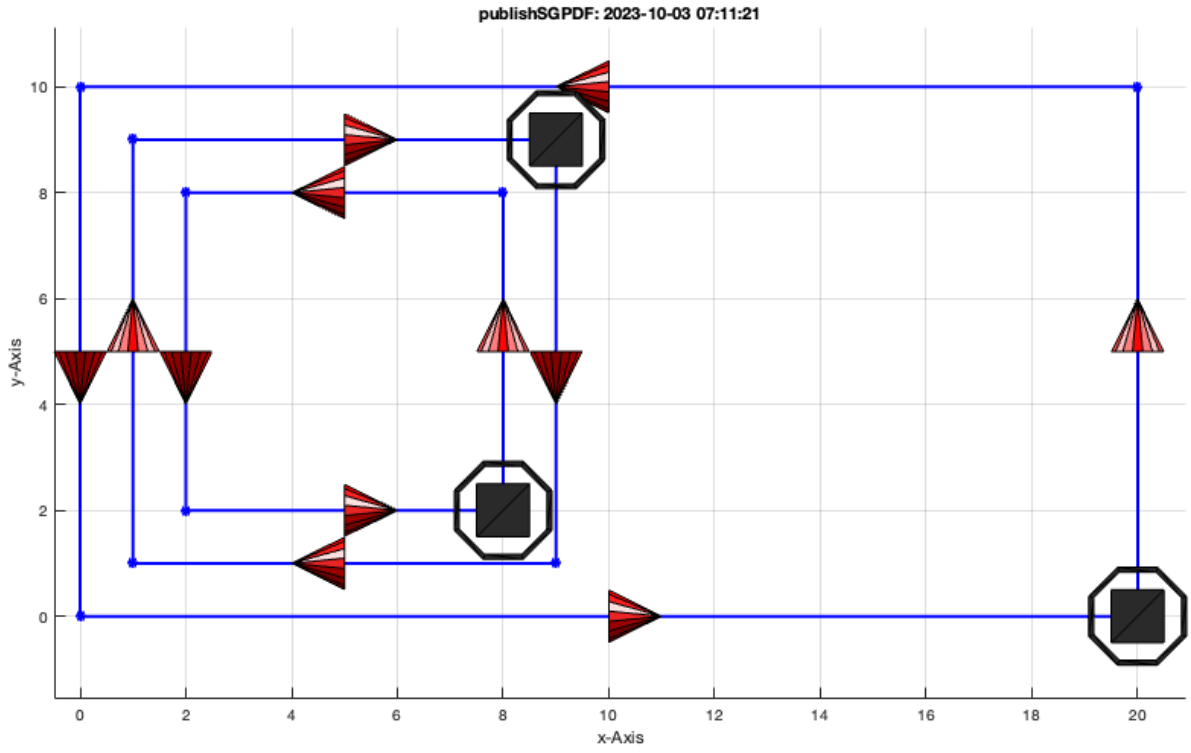
```
CPL=[0 0; 20 0; 20 10; 0 10; NaN NaN; 1 1; 9 1; 9 9; 1 9; NaN NaN; 2 2; 8 2; 8 8; 2 8 ];

close all;PLELofCPL(CPL);
```




```
CPL=CPLuniteCPL(CPL);
close all; PLELofCPL(CPL);
```

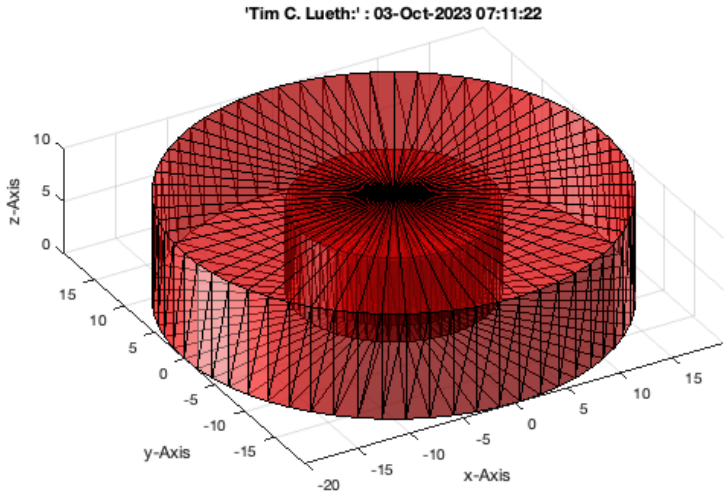



```
CPL=flip(CPL);
close all; PLELofCPL(CPL);
```

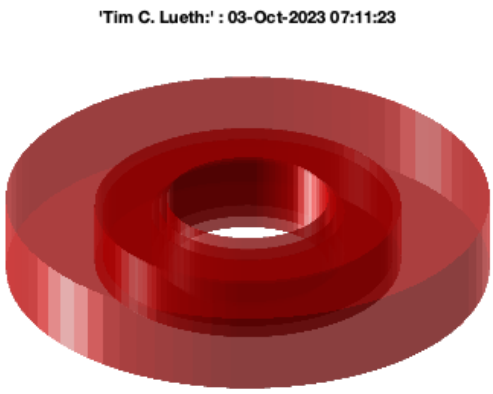


6. Rotate Contours around the z-axis

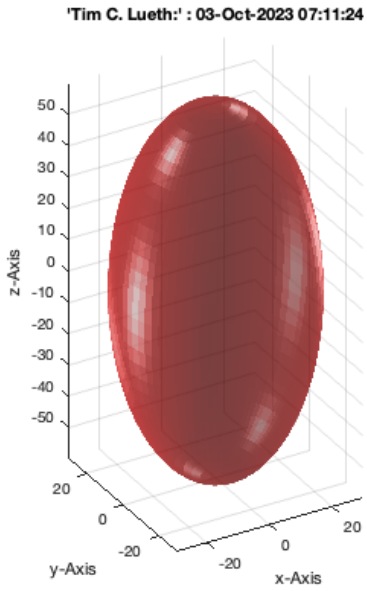
```
VLFLfigure; view(-30,30); grid on;
SG=SGofCPLrot(CPL);
SGplot(SG); VLFLplotlight (0,0.5);
```




```
VLFLfigure; view(-30,30); grid on;
CPL(:,1)=CPL(:,1)+10;
SG=SGofCPLrot(CPL);
SGplot(SG); VLFLplotlight (1,0.5);
```

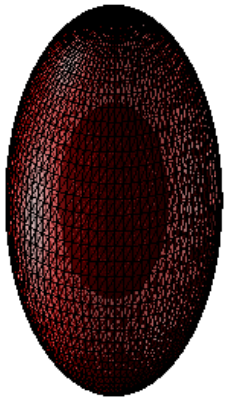



```
VLFLfigure; view(-30,30); grid on;
CPL=PLcircle(30); CPL(:,2)=CPL(:,2)*2;
SG=SGofCPLrot(CPL);
SGplot(SG); VLFLplotlight (1,0.5);
```



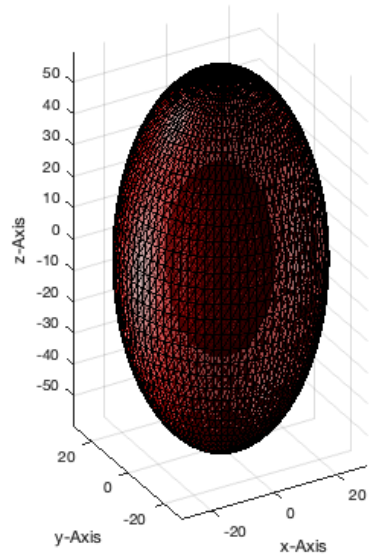

```
VLFLfigure; view(-30,30); grid on;
CPL=PLcircle(30); CPL(:,2)=CPL(:,2)*2;
CPL=[CPL;NaN NaN;CPL*0.5];
SG=SGofCPLrot(CPL);
SGplot(SG); VLFLplotlight (0,0.5);
```

'Tim C. Lueth:' : 03-Oct-2023 07:11:25



```
VLFLfigure; view(-30,30); grid on;
CPL=PLcircle(30); CPL(:,2)=CPL(:,2)*2;

SG=SGcat(SGofCPLrot(CPL),SGswap(SGofCPLrot(CPL*0.5)));
SGplot(SG); VLFLplotlight (0,0.5);
```

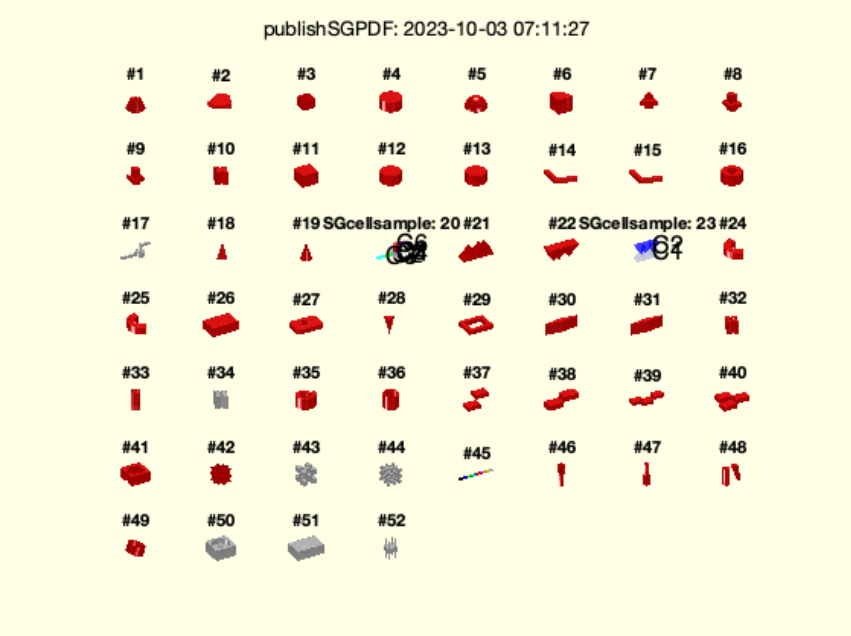
'Tim C. Lueth:' : 03-Oct-2023 07:11:26

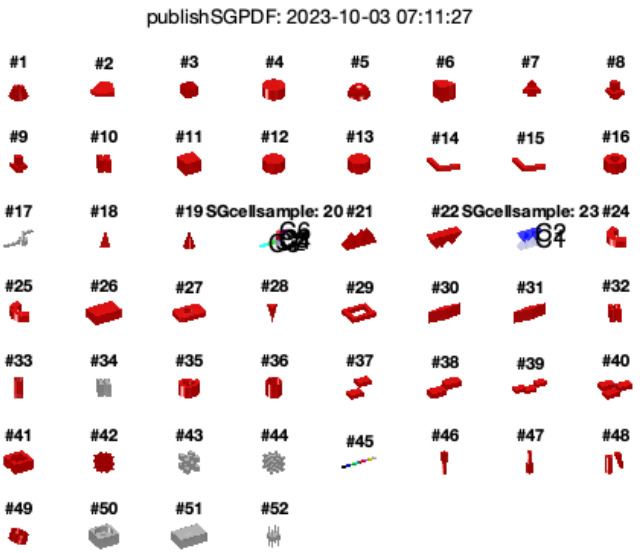


7. Samples of 3D Design

```
SGsample;
```

```
SGsample: 5% 10% 15% 20% 25% 30% 35% 40% 45% 50% 55% 60%
```





Final remarks on toolbox version and execution date

```
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 07:11:56!
Executed 03-Oct-2023 07:11:58 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
phased_array_system_toolbox
```

```
signal_blocks
signal_toolbox
simmechanics
simscape
simulink
statistics_toolbox
==========================================================================================
```

- *Tim Lueth, tested and compiled on OSX 10.7.5 with Matlab 2014b on 2014-11-30*
- *Mattias Traeger, executed and published on 64 Bit PC using Windows with Matlab 2014b on YYYY-MMM-DD*

Published with MATLAB® R2023a