Exercise 14: Manipulation Functions for Closed Polygons and Laser Cutting (SVG)

2015-09-20: Tim C. Lueth, MIMED - Technische Universität München, Germany (URL: http://www.mimed.de)

Contents

- 1. Summary of the already explained aspects of the VLFL-Toolbox
- 2. Fill a contour with copies of pattern
- 3. Writing a contour as SVG-File for laser-cutting
- 4. Calculating the normal vectors of edges and points
- 5. Growing with same number of points
- 6. Growing with correct distance to edges
- 7. Rounded edges inside a contour
- 8. Sort CPLs around its center
- 9. Informations on contours inside of others
- 10. Order contours for the sequential plot with a laser cutter
- Final remarks on toolbox version and execution date

1. Summary of the already explained aspects of the VLFL-Toolbox

In examples 1-13 we've learned about:

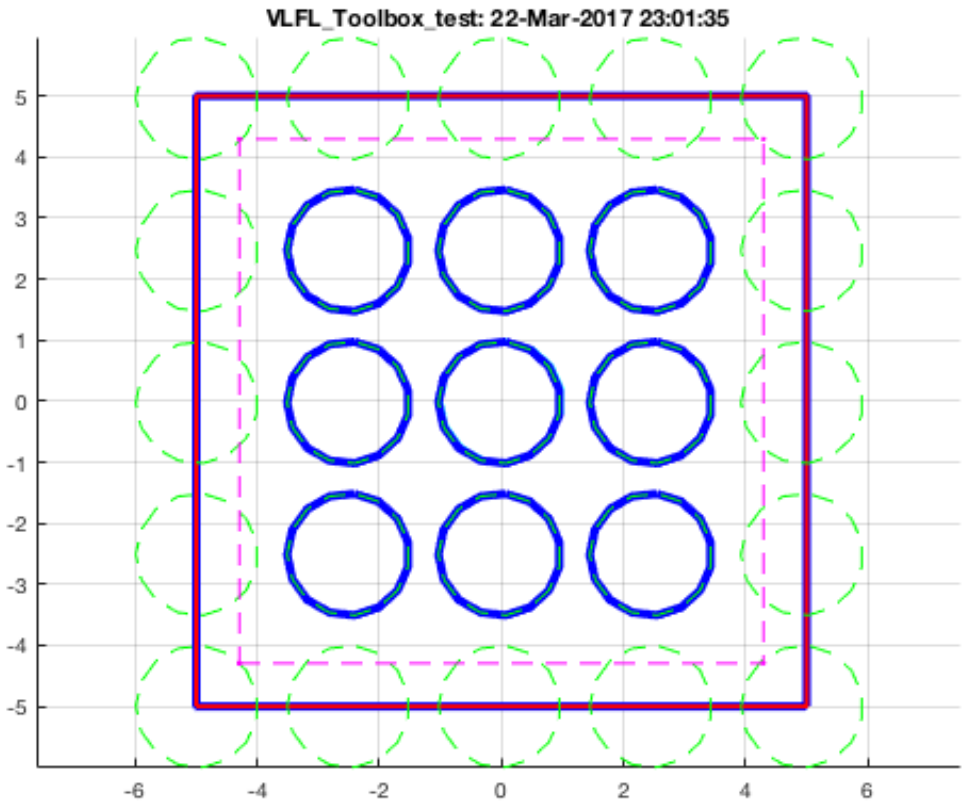
- *Point lists* (PL), *Vertex lists* (VL), *Facet lists* (VL), *Edge lists* (EL)
- Functions to import, check, and export of *STL-File* data
- Functions for generating text strings as solid volumes
- *Closed polygon lists* (CPL) for boolesche operations of contours
- 2.5D extrusion of closed polyogon list (CPL) to *Solid Geometries* (SG)
- Spatial transformation and merging of solid geometry sets
- Relative positioning of solids geometries and boolean operations on solids
- Packaging and boxing of several parts into a complete assemble kit
- Attaching coordinates frames and the creation of kinematik models
- Function for mounting faces of solids and creation of lightweight-solids

2. Fill a contour with copies of pattern

As it was shown already in the function SGplatesofSGML, it often makes sense to fill a contour with another pattern. This can be done by using one of the following functions:

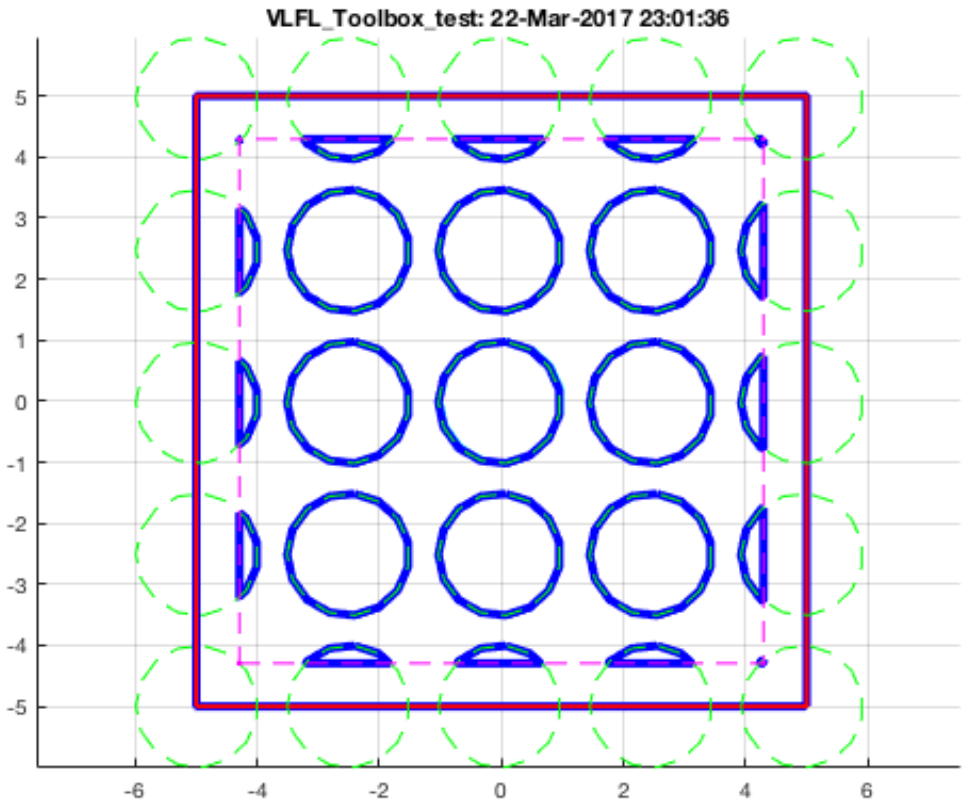
- **CPLfillPattern(CPLA, CPLB,w,d)** - fills a contour CPLA with copies of the pattern CPLB with a distance to the outer contour w and distance between the patterns of d

```
SGfigure; view(0,90); axis on;
CPLfillPattern(PLsquare(10,10),PLcircle(1),1);
```



This can also be done with cutted pattern instead of complete pattern

```
SGfigure; view(0,90); axis on;
CPLfillPattern(PLsquare(10,10),PLcircle(1),1,[],true);
```



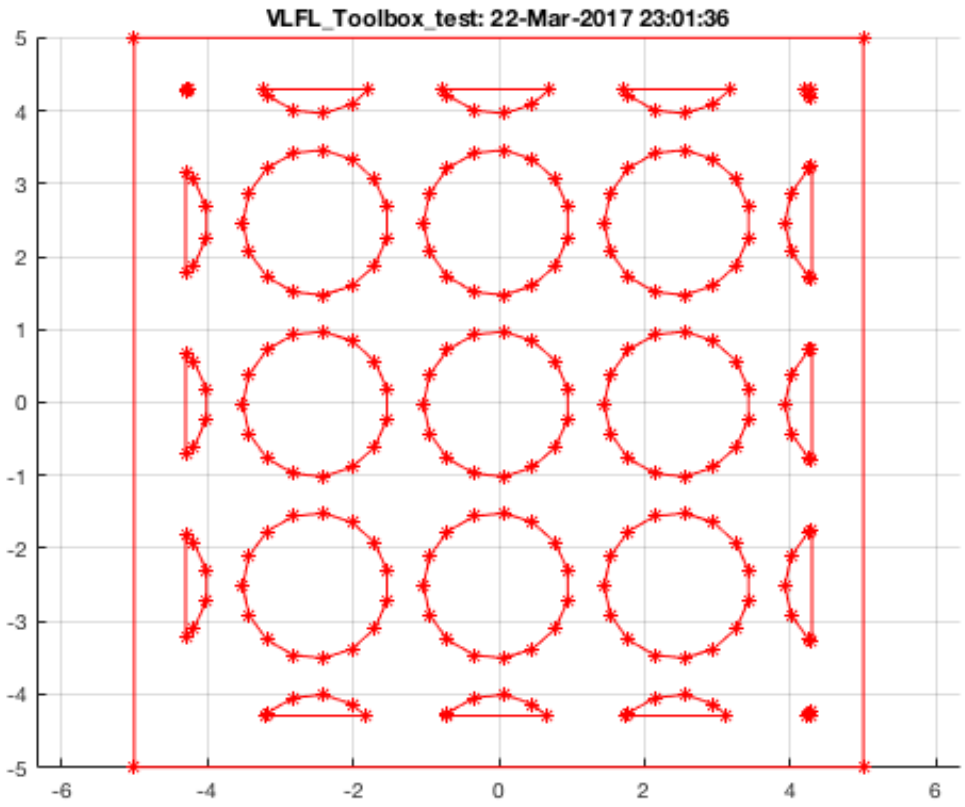
3. Writing a contour as SVG-File for laser-cutting

Especially for laser cutting or platting of contours, the SVG file-format is very popular. Handling of SVG Files is possible using the following functions:

- **CPLwriteSVG (CPL,Filename)** - writes the contours in a SVG-File

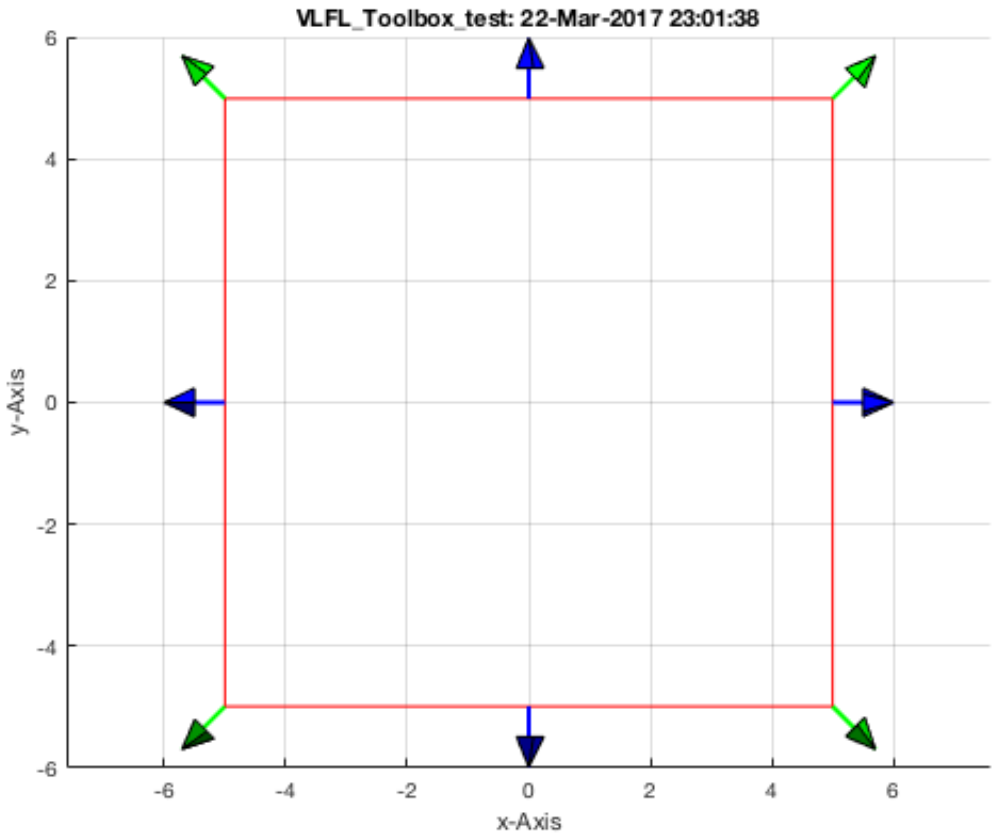
```
SGfigure; view(0,90); axis on;
A=CPLfillPattern(PLsquare(10,10),PLcircle(1),1,[],true);
CPLplot(A);
CPLwriteSVG(A,'VLFL_EXP14');
```

```
WRITING SVG FILE /Users/lueth/Desktop/VLFL_EXP14.SVG in ASCII MODE
completed.
```



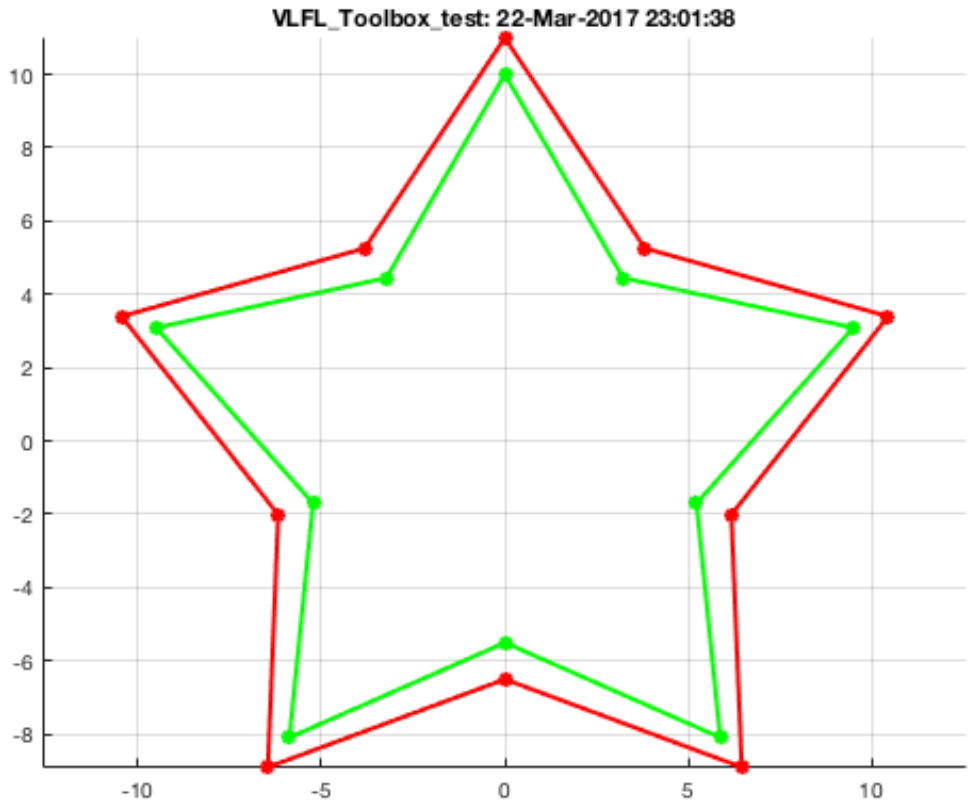
4. Calculating the normal vectors of edges and points

```
SGfigure; view(0,90);
CPLedgeNormal(PLsquare(10,10)); axis on;
```



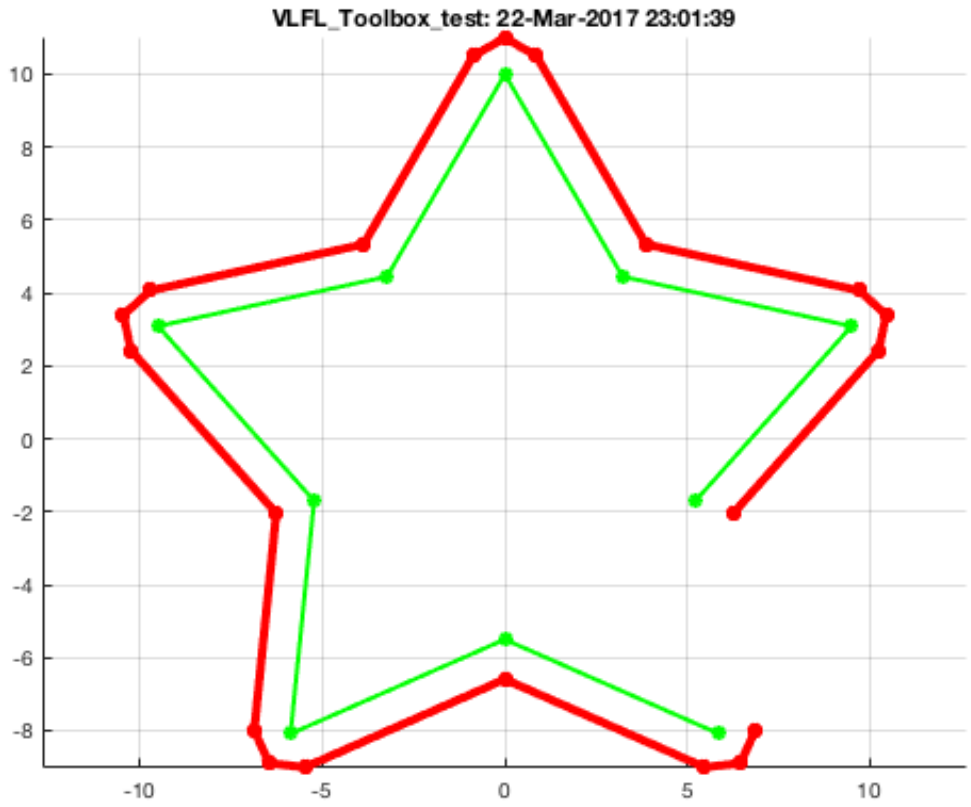
5. Growing with same number of points

```
SGfigure; view(0,90);
CPLgrow(PLstar(10,10),1); axis on;
```



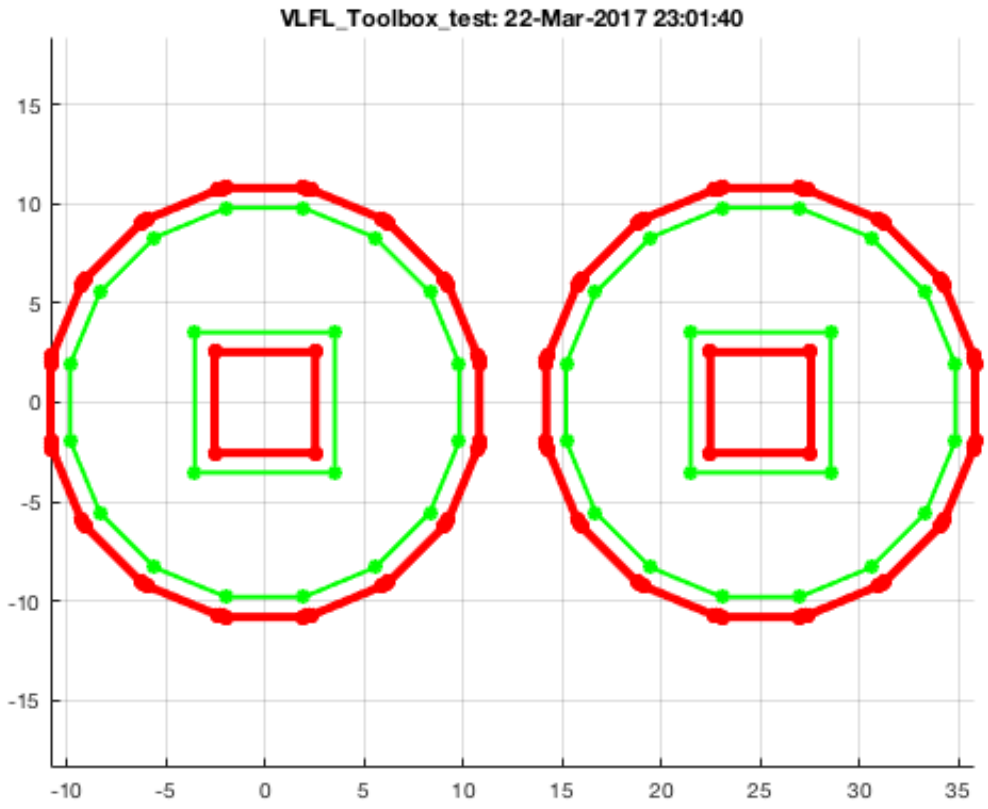
6. Growing with correct distance to edges

```
SGfigure; view(0,90); axis on;
CPLgrowEdge(PLstar(10,10),1);
```



Another example using CPLsample

```
SGfigure; view(0,90);
CPLgrowEdge(CPLsample(12),1); axis on;
```



Growing may have no problems

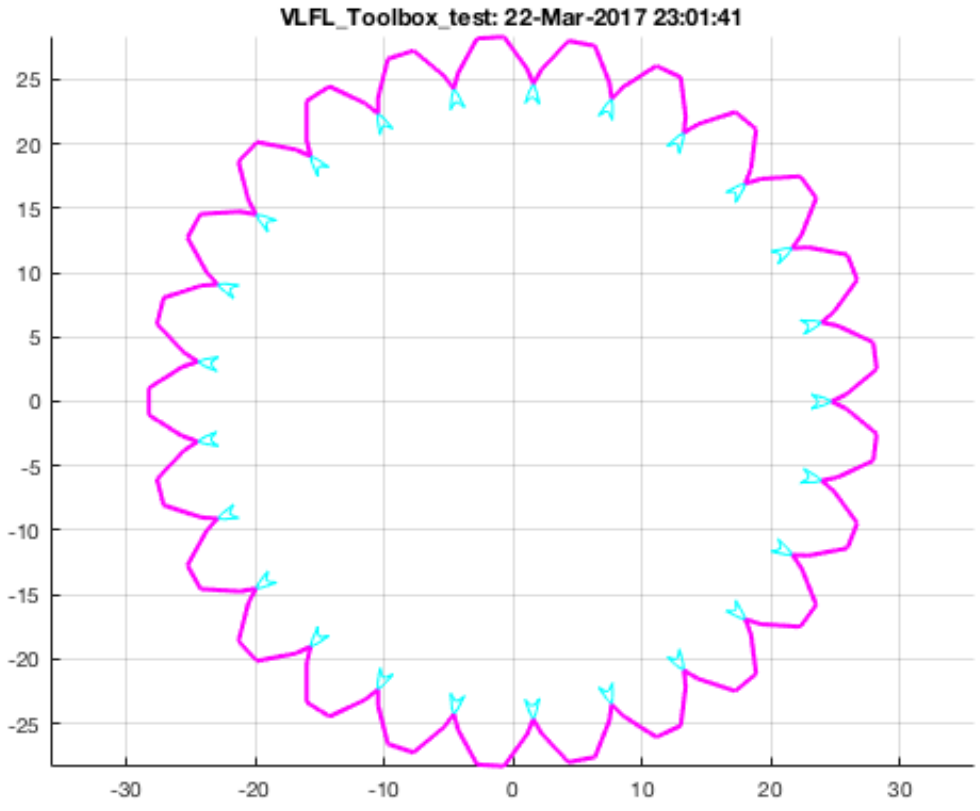
```
SGfigure; view(0,90);
CPLgrow(CPLofPL(PLgearDIN(2,25)),0.5); axis on;

% *Growing may have problems*
SGfigure; view(0,90);
CPLgrow(CPLofPL(PLgearDIN(2,25)),1.5); axis on;

% *Growing problems can be solved using CPLoutercontour*
SGfigure; view(0,90);
CPLoutercontour(CPLgrow(CPLofPL(PLgearDIN(2,25)),1.5));
```

```
Modul: m=2.00
Teeth Nr: z=25
Rolling circle: d=50.00
Foot ratio: rsp=0.70
Tooth size factor: x=1.00
Foot diameter: df=45.00
Head diameter: dh=54.00
Tooth foot width: 3.96
Involute angle: 38.01
Tooth head width: 0.56
Difference tooth gap and tooth head width: 1.14
Modul: m=2.00
Teeth Nr: z=25
Rolling circle: d=50.00
```

```
Foot ratio: rsp=0.70
Tooth size factor: x=1.00
Foot diameter: df=45.00
Head diameter: dh=54.00
Tooth foot width: 3.96
Involute angle: 38.01
Tooth head width: 0.56
Difference tooth gap and tooth head width: 1.14
Modul: m=2.00
Teeth Nr: z=25
Rolling circle: d=50.00
Foot ratio: rsp=0.70
Tooth size factor: x=1.00
Foot diameter: df=45.00
Head diameter: dh=54.00
Tooth foot width: 3.96
Involute angle: 38.01
Tooth head width: 0.56
Difference tooth gap and tooth head width: 1.14
```

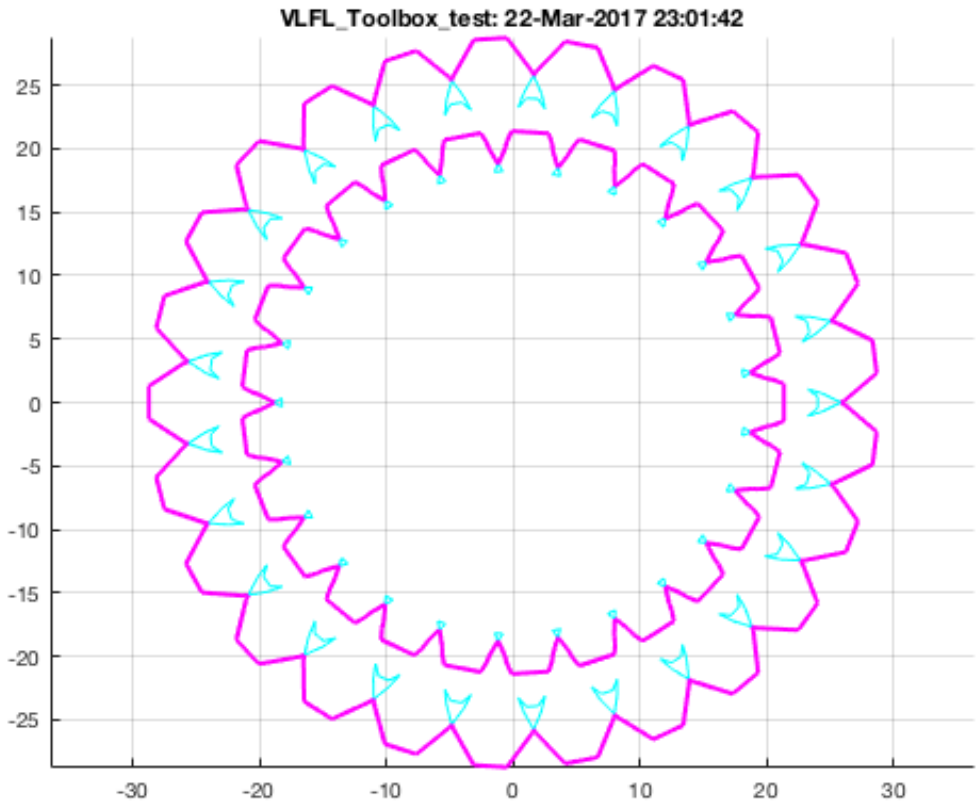


Another example using CPLoutercontour

```
SGfigure; view(0,90);
CPLoutercontour(CPLsample(25),1); axis on;
```

```
Modul: m=2.00
```

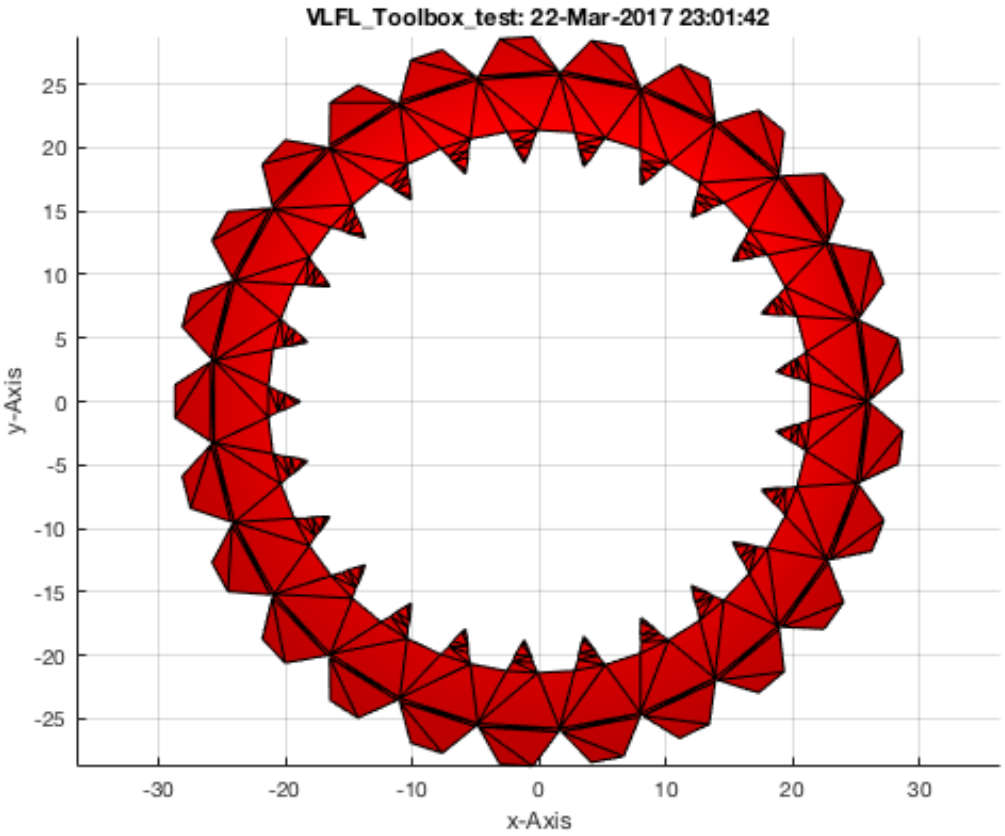
```
Teeth Nr: z=25
Rolling circle: d=50.00
Foot ratio: rsp=0.70
Tooth size factor: x=1.00
Foot diameter: df=45.00
Head diameter: dh=54.00
Tooth foot width: 3.96
Involute angle: 38.01
Tooth head width: 0.56
Difference tooth gap and tooth head width: 1.14
Modul: m=1.50
Teeth Nr: z=25
Rolling circle: d=37.50
Foot ratio: rsp=0.70
Tooth size factor: x=1.00
Foot diameter: df=41.25
Head diameter: dh=34.50
Tooth foot width: 3.63
Involute angle: 34.09
Tooth head width: 0.25
Difference tooth gap and tooth head width: 1.30
```




```
PLFLofCPLdelaunay(CPLoutercontour(CPLsample(25),1));
```

```
Modul: m=2.00
Teeth Nr: z=25
```

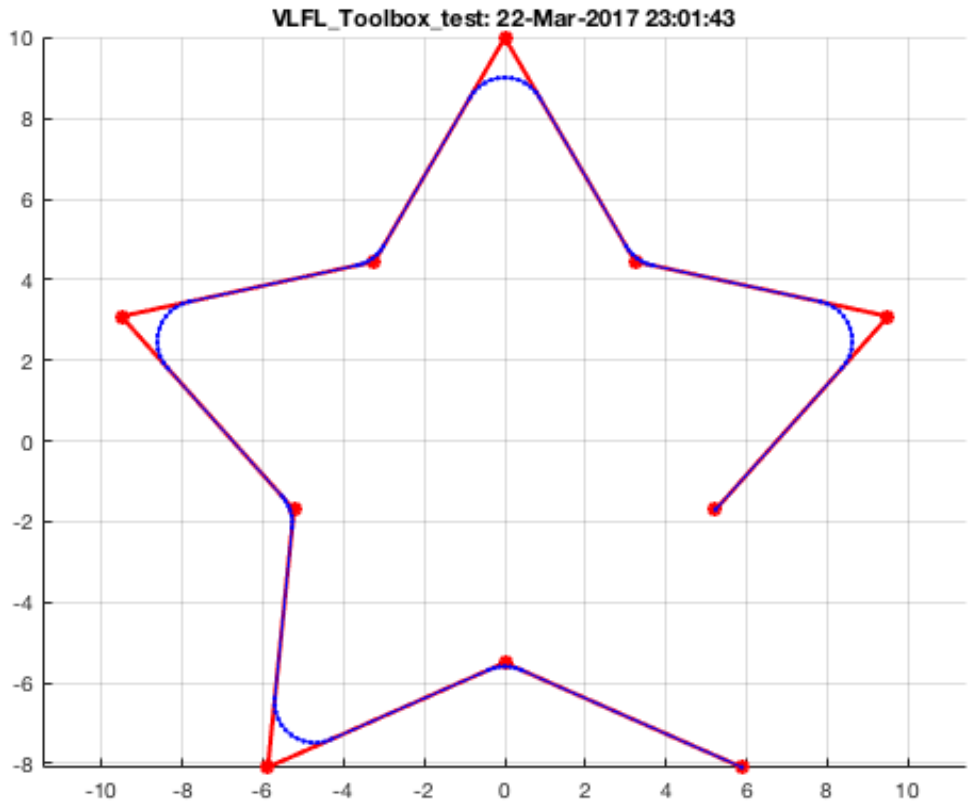
```
Rolling circle: d=50.00
Foot ratio: rsp=0.70
Tooth size factor: x=1.00
Foot diameter: df=45.00
Head diameter: dh=54.00
Tooth foot width: 3.96
Involute angle: 38.01
Tooth head width: 0.56
Difference tooth gap and tooth head width: 1.14
Modul: m=1.50
Teeth Nr: z=25
Rolling circle: d=37.50
Foot ratio: rsp=0.70
Tooth size factor: x=1.00
Foot diameter: df=41.25
Head diameter: dh=34.50
Tooth foot width: 3.63
Involute angle: 34.09
Tooth head width: 0.25
Difference tooth gap and tooth head width: 1.30
```



7. Rounded edges inside a contour

Another method to change the shape of a contour is to round the edges.

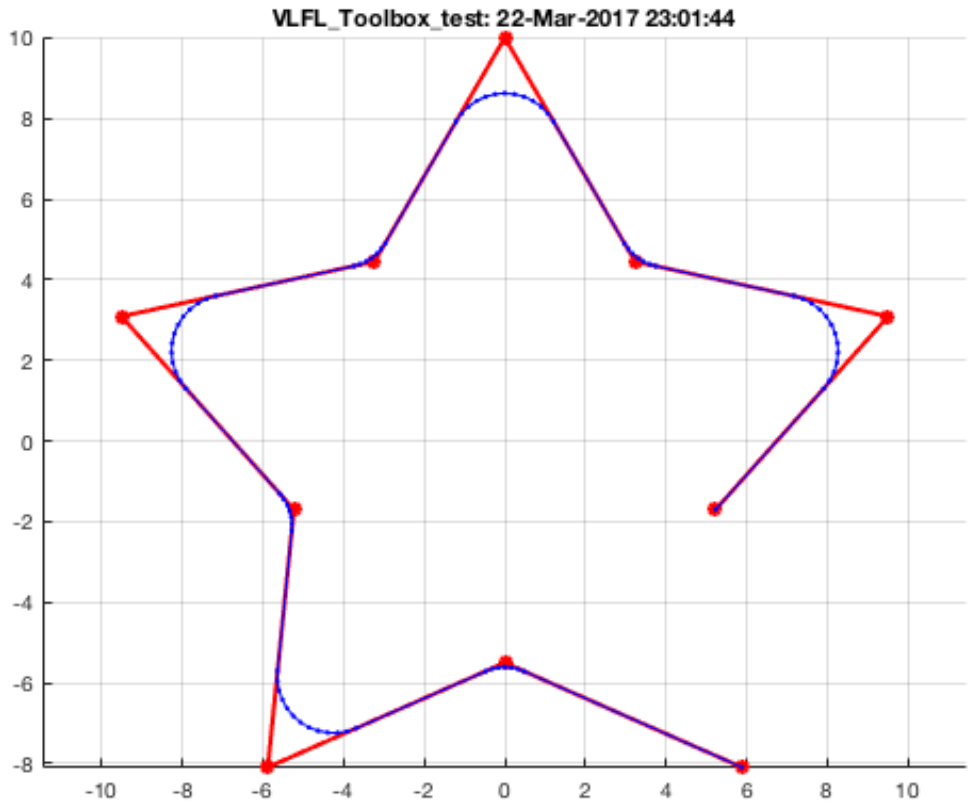
```
SGfigure; view(0,90);
PLradialEdges(PLstar(10,10));axis on;
```



Another example using radius=2

```
SGfigure; view(0,90); axis on;
PLradialEdges(PLstar(10,10),2); axis on;
```

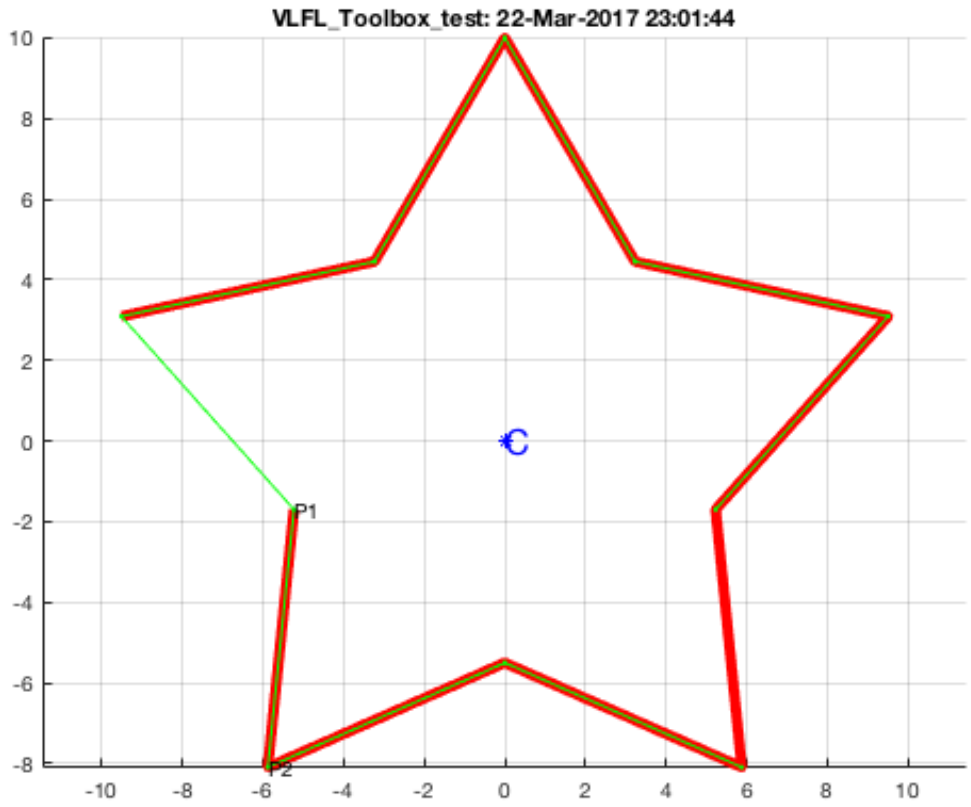
```
Radial egde radius reduced R=2 to 1.4 at [9.5 3.1] w=119.6
Radial egde radius reduced R=2 to 1.2 at [3.2 4.4] w=-47.6
Radial egde radius reduced R=2 to 1.4 at [0.0 10.0] w=119.6
Radial egde radius reduced R=2 to 1.2 at [-3.2 4.4] w=-47.6
Radial egde radius reduced R=2 to 1.4 at [-9.5 3.1] w=119.6
Radial egde radius reduced R=2 to 1.2 at [-5.2 -1.7] w=-47.6
Radial egde radius reduced R=2 to 1.4 at [-5.9 -8.1] w=119.6
Radial egde radius reduced R=2 to 1.2 at [-0.0 -5.5] w=-47.6
```



8. Sort CPLs around its center

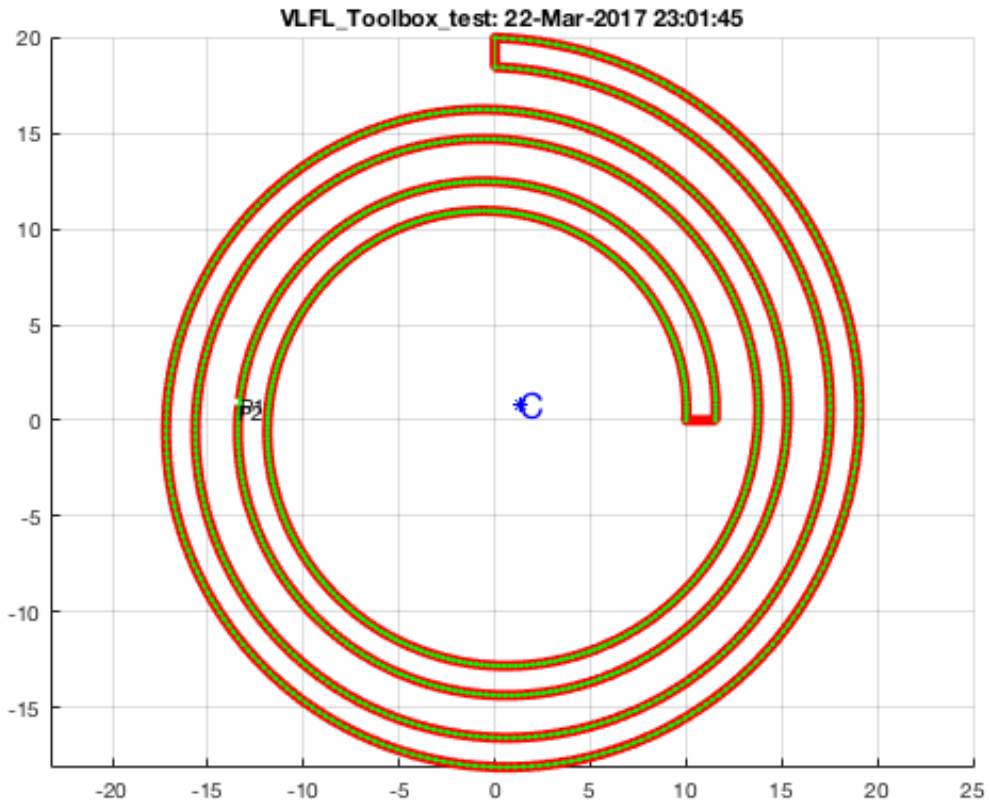
Find the minmal angle value of a star

```
SGfigure; view(0,90);
CPLsortC(PLstar(10,10),'min'); axis on;
```



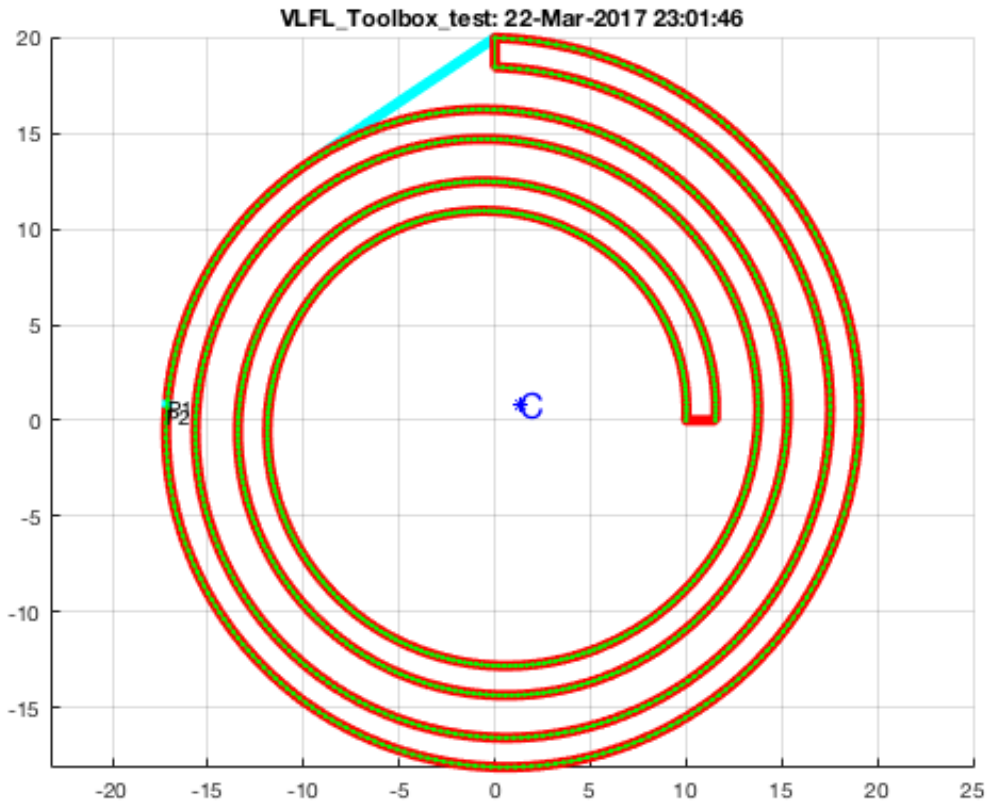
Find the minmal angle value of a spiral

```
SGfigure; view(0,90);
CPLsortC(CPLspiral(10,20,4*pi+pi/2),'min');
```



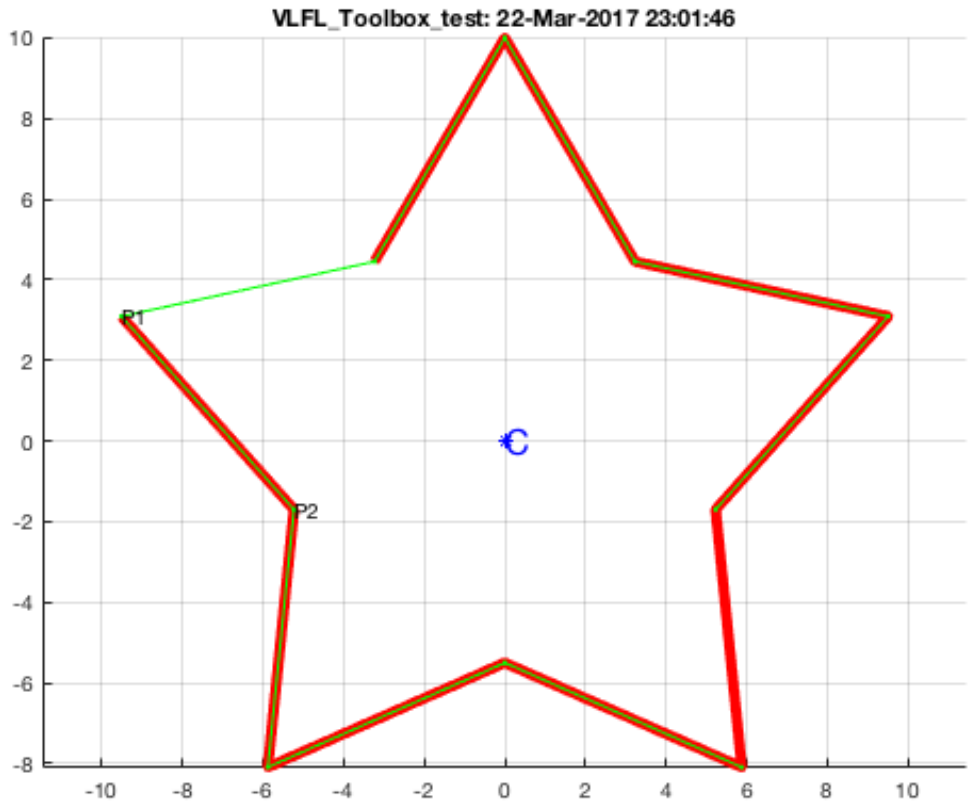
Find the minmal angle value of convex hull of a spiral

```
SGfigure; view(0,90);
CPLsortC(CPLspiral(10,20,4*pi+pi/2),'cmin');
```



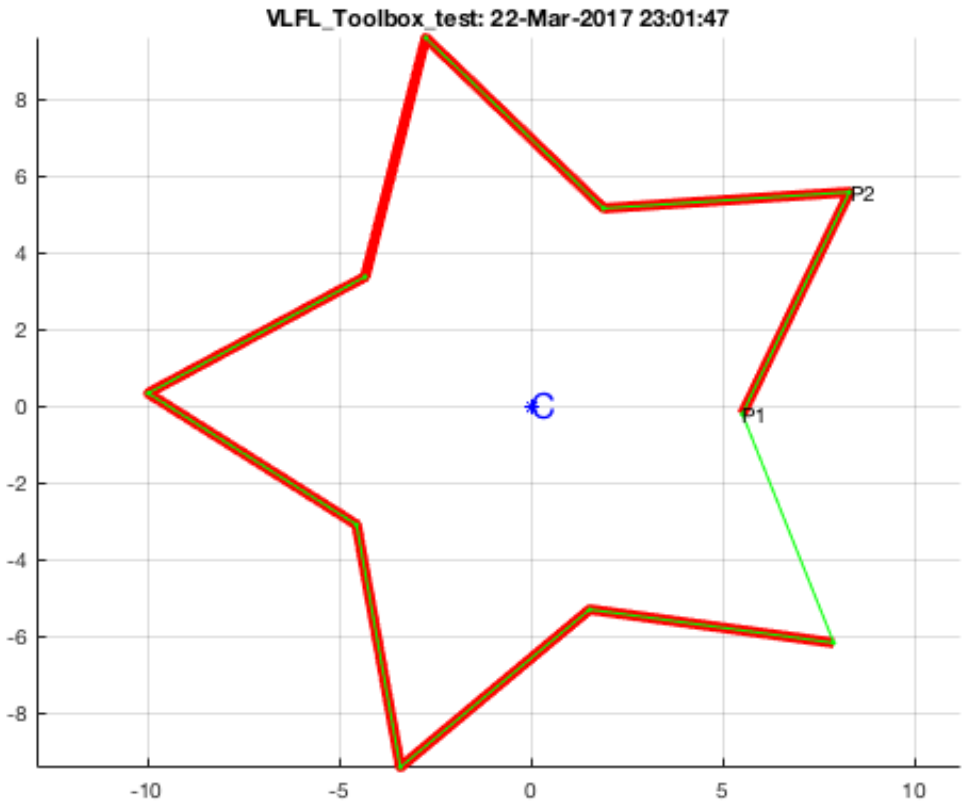
Find the maximum angle value of a star

```
SGfigure; view(0,90);
CPLsortC(PLstar(10,10),'max'); axis on;
```



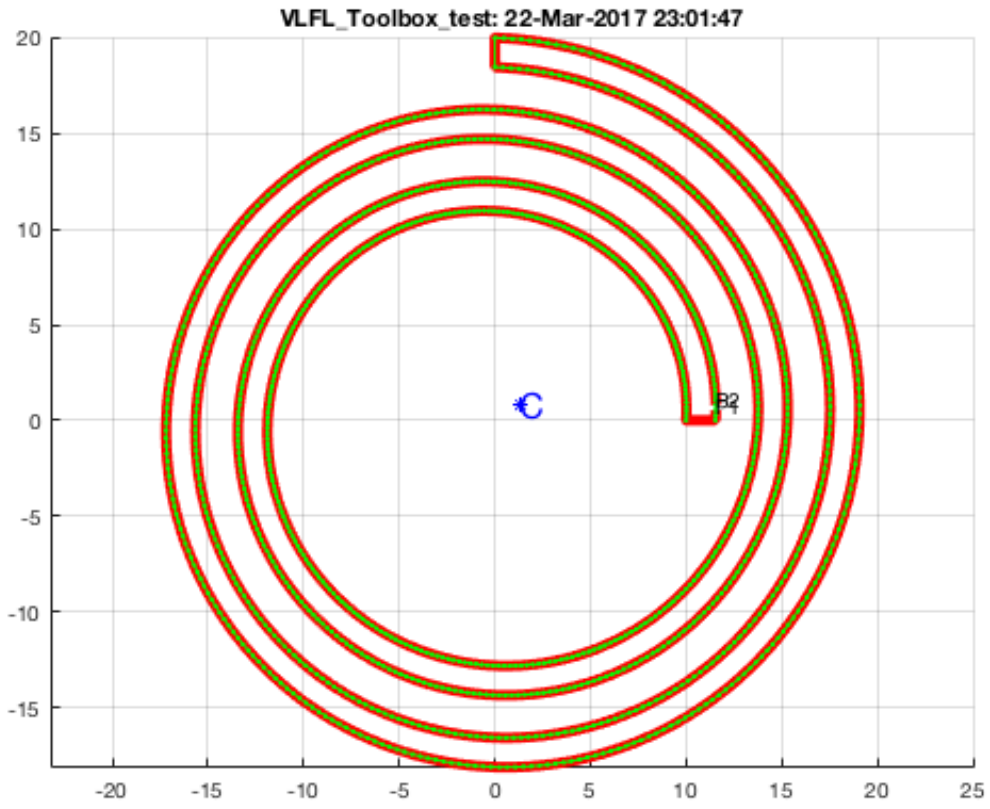
Find the angle value nearest to zero of a star

```
SGfigure; view(0,90);
CPLsortC(PLtrans(PLstar(10,10),rotdeg(160)),'zero'); axis on;
```



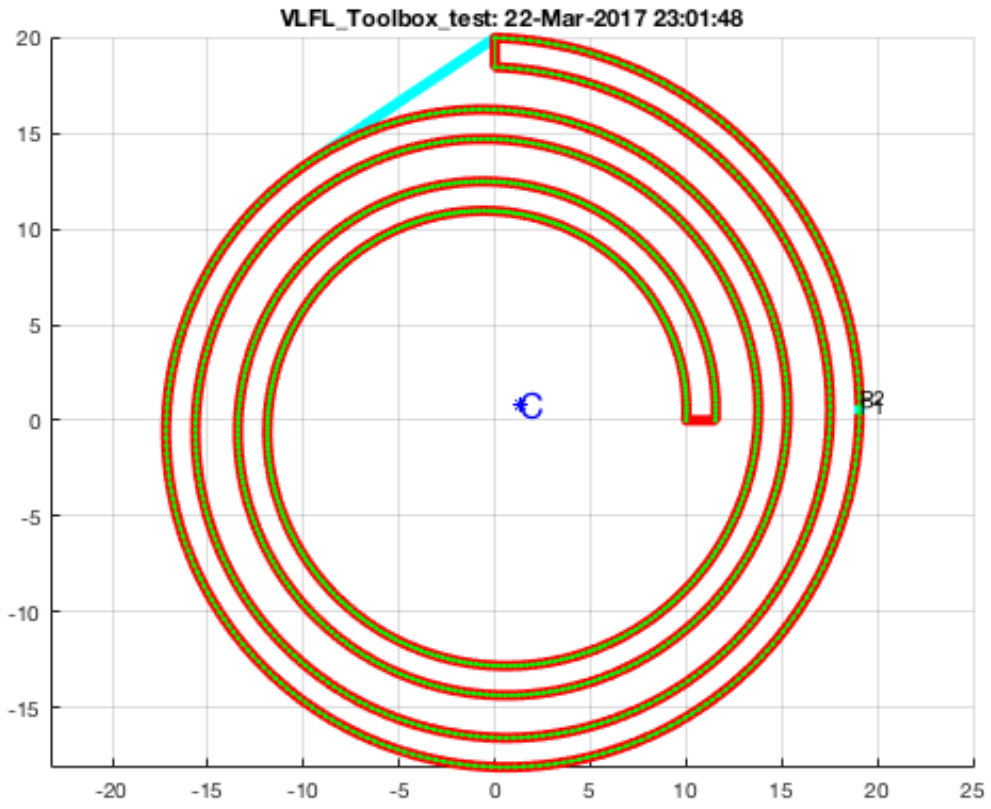
Find the angle value nearest to zero of a spiral

```
SGfigure; view(0,90);
CPLsortC(CPLspiral(10,20,4*pi+pi/2),'zero');
```



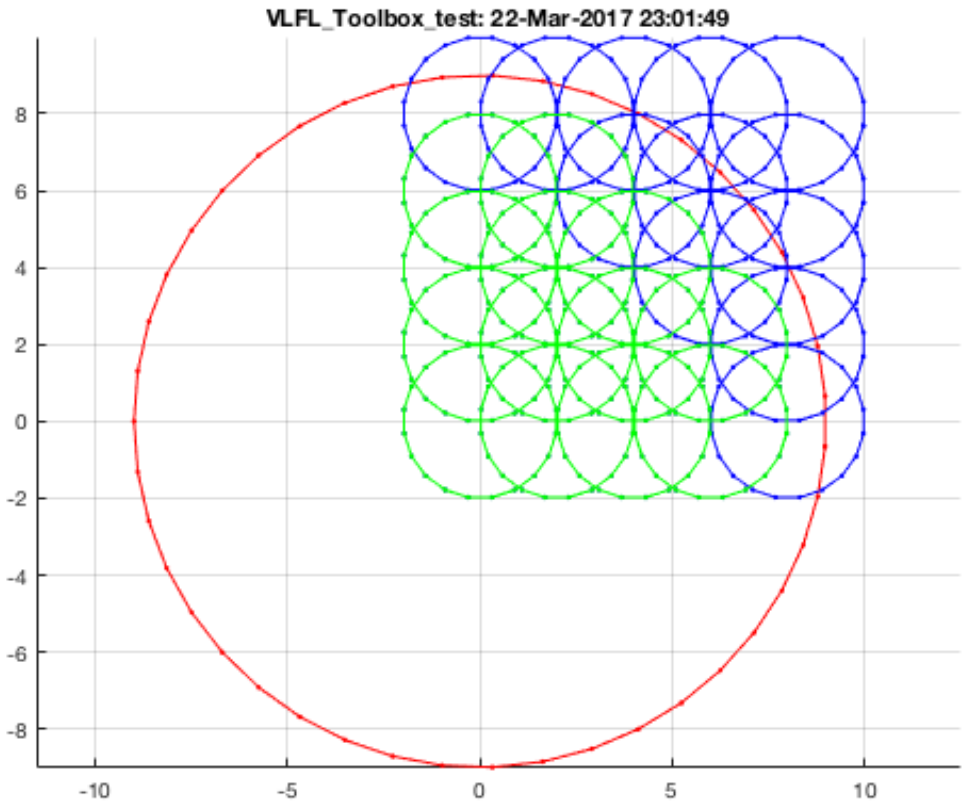
Find the angle value nearest to zero of convex hull of a spiral

```
SGfigure; view(0,90);
CPLsortC(CPLspiral(10,20,4*pi+pi/2),'czero');
```



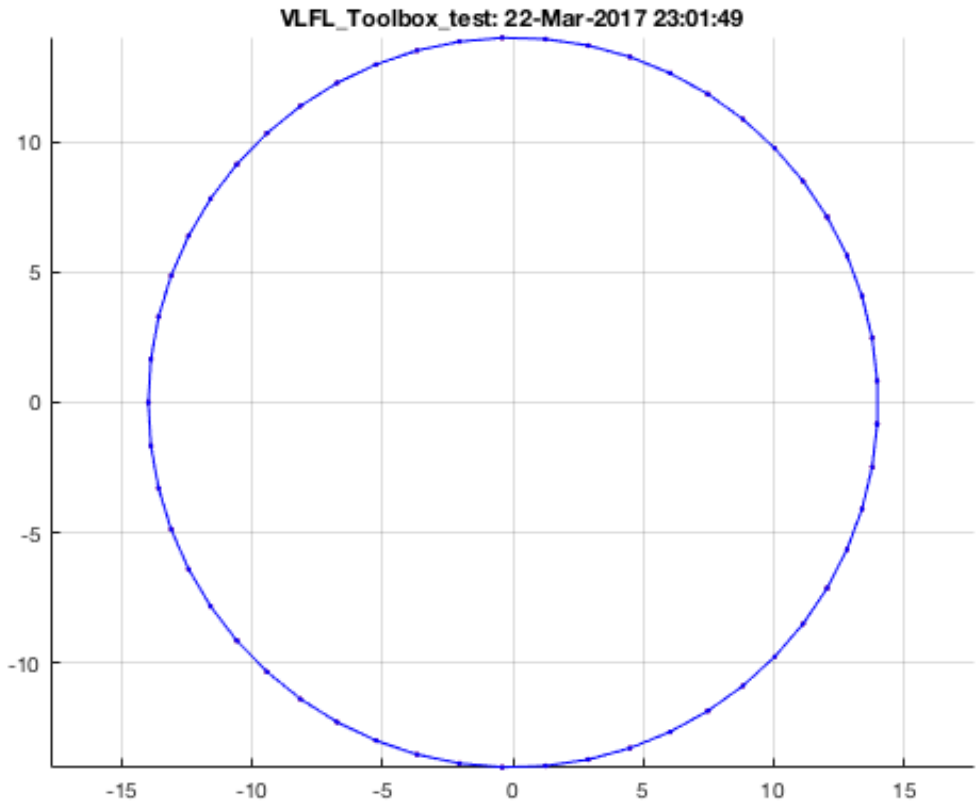
9. Informations on contours inside of others

```
SGfigure; view(0,90);
CPLinsideCPL(PLcircle(9),CPLcopypattern(PLcircle(2),[5 5],[2 2])); axis on;
```



Identical contours are not inside each other

```
CPLinsideCPL(PLcircle(14),CPLsortC(PLcircle(14)));
```



10. Order contours for the sequential plot with a laser cutter

The "level" starts with zero runs from outer to inner. In case of a laser cutter it is necessary to cut the inner contours first.

- ***CPLwriteSVG**** - writes a CPL ans SVG on disk
- ***svgpolylineofCPL**** - plots an SVG file
- ***separateNaN**** - separates CPLs and CVLs
- ***selectNaN**** - creates a new set of selectex CPLs/CVLs
- ***CPLsortinout**** - sorts contours to inner and outer

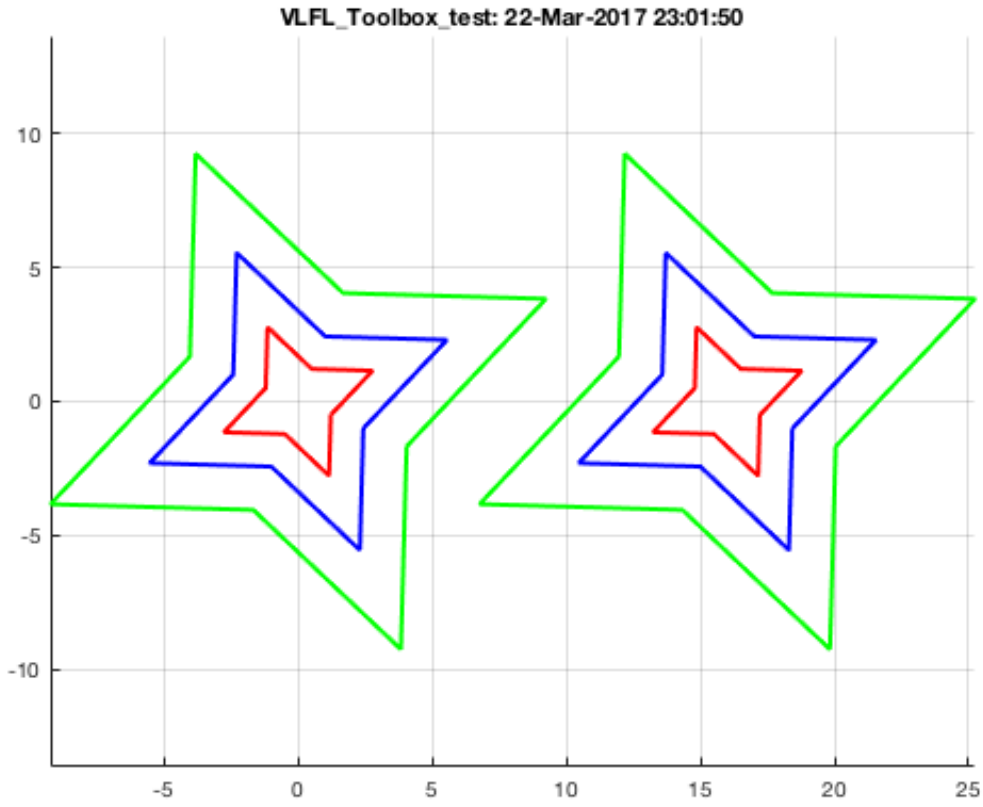
```
SGfigure; view(0,90);
[ci,CC]=CPLsortinout(CPLsample(14))
CPLsortinout(CPLsample(14));
```

```
ci =

     0
     1
     2
     0
     1
     2

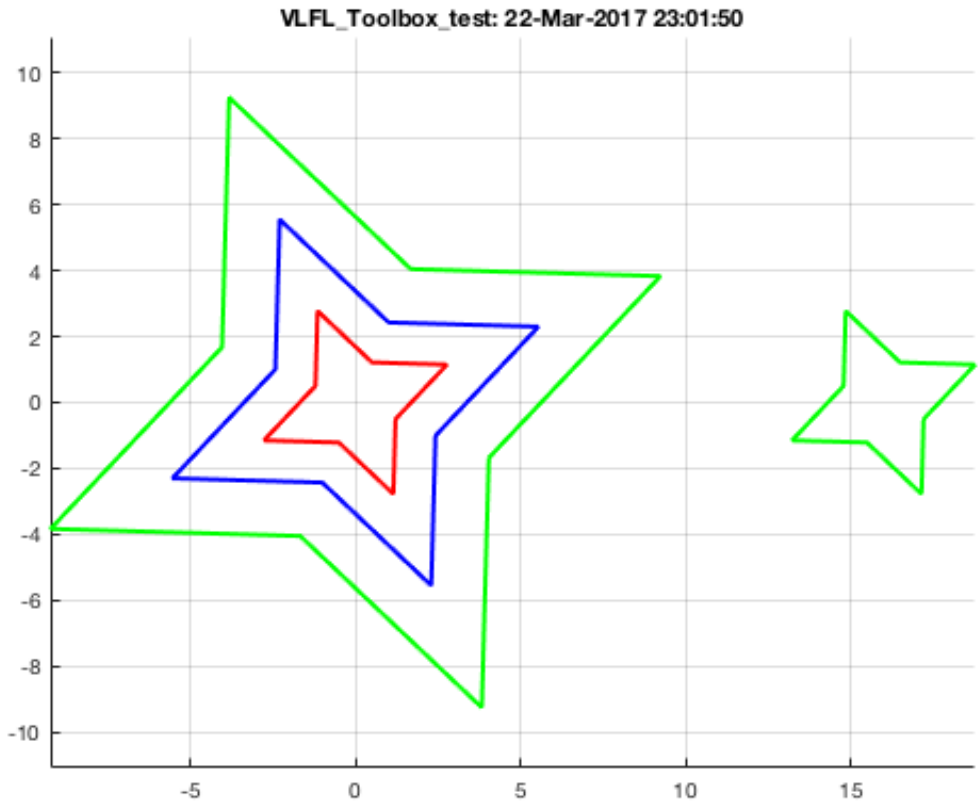

CC =
```

```
   NaN     1     1    -1    -1    -1
    -1   NaN     1    -1    -1    -1
    -1    -1   NaN    -1    -1    -1
    -1    -1    -1   NaN     1     1
    -1    -1    -1    -1   NaN     1
    -1    -1    -1    -1    -1   NaN
```



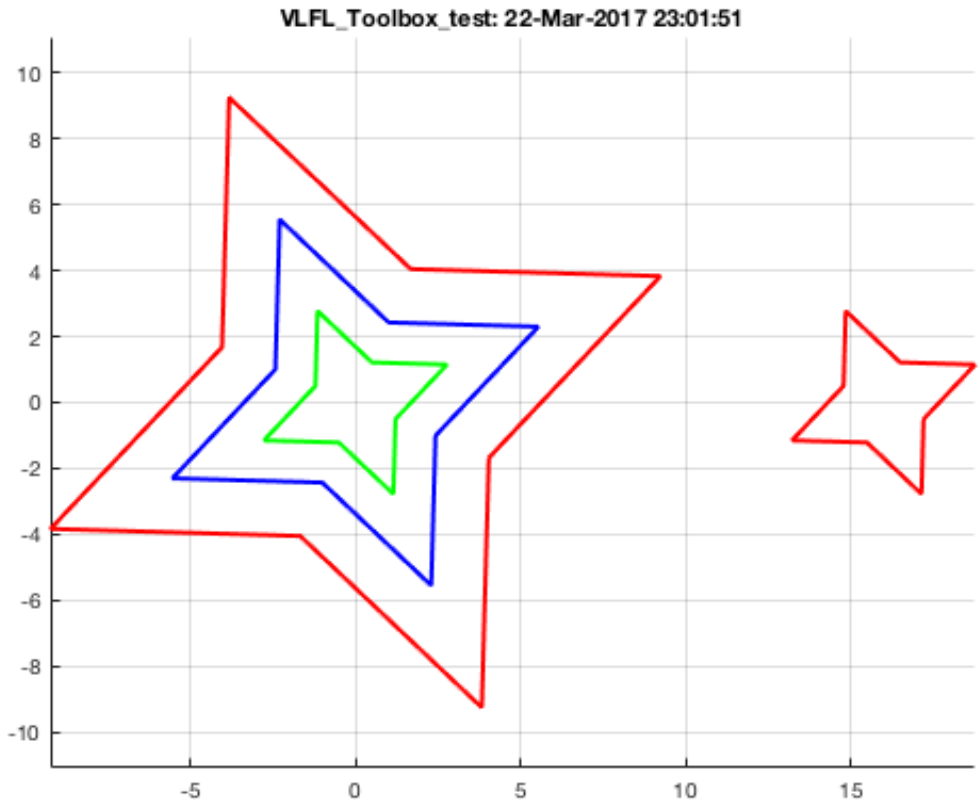
Show a selction from inner to outer for laser cutting

```
CPLsortinout(selectNaN(CPLsample(14),[1,2,3,6]));
```



Now change the order direction from outer to inner

```
CPLsortinout(selectNaN(CPLsample(14),[1,2,3,6]),false);
```



Now write is als a cutter file

```
CPLwriteSVG(CPLsample(14),'VLFL_EXP14_cutter','',true);
```

```
WRITING SVG FILE /Users/lueth/Desktop/VLFL_EXP14_cutter.SVG in ASCII MODE
completed.
```

Final remarks on toolbox version and execution date

```
VLFLlicense
```

```
This VLFL-Lib, Rel. 3.6 (2017-Mar-23), is for limited non commercial educational use only!
Licensee: 3.6 Tim Lueth for academic and education 05-Mar-2017 !
Please contact Tim Lueth, Professor at TU Munich, Germany!
WARNING: This VLFL-Lib (Rel. 3.6) license will exceed at 18-Mar-2018!
Executed 22-Mar-2017 23:01:53 by 'lueth' using Matlab 9.1.0.441655 (R2016b) on a MACI64
======================================== Used Matlab products: ============================
==========
map_toolbox
matlab
simmechanics
simscape
simulink
================================================================================================
```

==========

- *Tim Lueth, tested and compiled on OSX 10.7.5 with Matlab 2014b on 2015-09-20*
- _____, executed and published on 64 Bit PC using Windows with Matlab 2015a on 2015-xx-xx_

Published with MATLAB® R2016b