Tutorial 69: Design of a 4Bar Linkage for a Hook Fixation

2022-02-22: Tim C. Lueth, Professor at Technische Universität München, Germany (URL: http://www.SG-Lib.org) - Last Change: 2022-02-22

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: 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

Motivation for this tutorial: (Originally SolidGeometry 5.1 required)

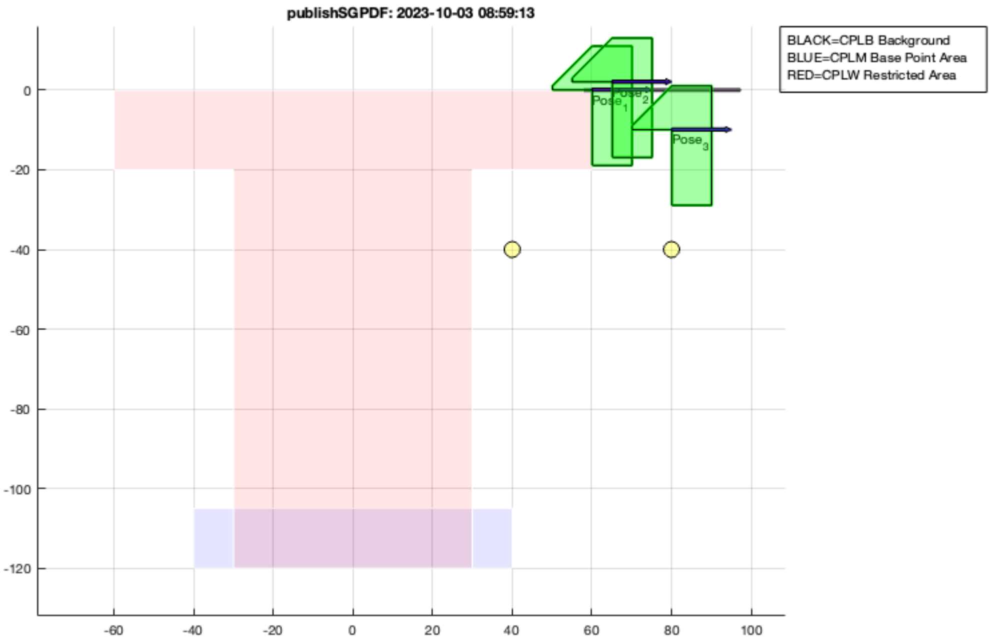
Design a Hook

```
A=PLsquare(120,20);
B=PLsquare(60,100);
CPLW=CPLtransrelCPL(B,A,'center','under','add');
CPLW=CPLW+[0 -10];
D=PLsquare(80,15);

CPLM=CPLtransrelCPL(D,CPLW,'center','alignbottom');
CPLE=PLhook(30,10,[10 1],'',0);
PS.A=[60 0; 65 2; 80 -10] + 0.1;
PS.B=PS.A+[1 0];
PS.RACK=[40 -40; 80 -40];
PS.CPLW=CPLW;
PS.CPLM=CPLM;
PS.CPLE=CPLE;
```

PLOT THE RESULT

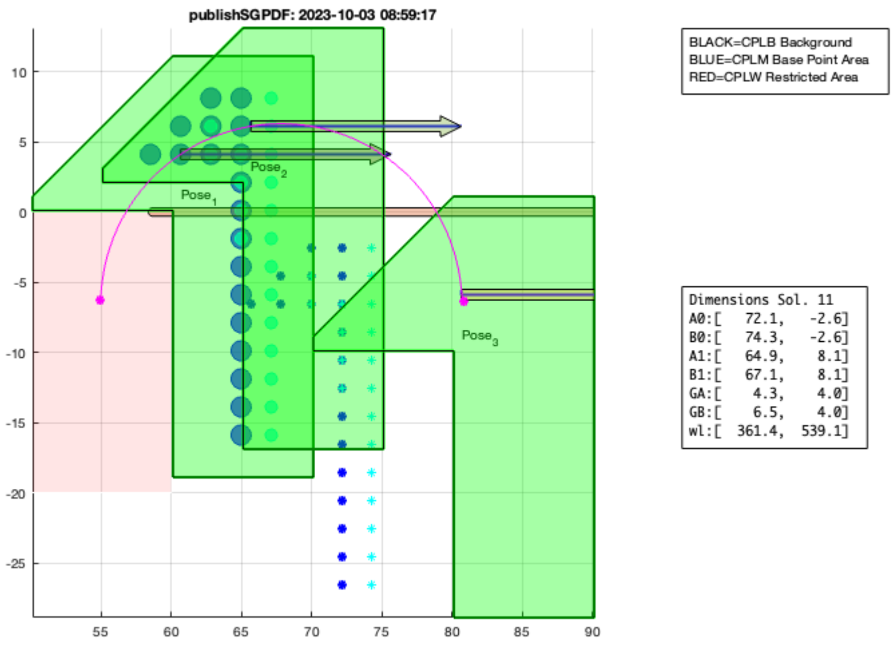
```
SGfigure;
Poseplotspace(PS);
Poseplot(PS);
```



NOW DESIGN A FOURBAR LINKAGE SOLUTIONS

```
fourBarposesyntheses(PS,[1 2 3],[10 3 3],[1 2 3],[1 3]); PS=ans;
```

```
fourBarposesyntheses: Number of Gridpoints should be: 10
PoseaddGPL: CPLE 3.0 mm Buffer for 8 attachment points
fourBarposesyntheses: 3-Pose-Synthesis using poses: [1,2,3]
PoseaddGPL: CPLE 3.0 mm Buffer for 32 attachment points
checkfourbar3Poseattachpermutation: Original solution format "PS.GAL" (3Poses) converted into general pose solution format: "PS.solut"
fourBarposesyntheses: 78 solution were found using 3-Pose-Synthesis.
fourBarposesyntheses: 51 solution were left after forcing a pose order
fourBarposesyntheses: Now forcing an angle limit [1,3]
fourBarposesyntheses: 51 Solutions are angle limited to [1,3]
Poseplotsolution: Plot a limited selection of 10 of 51 solutions
```

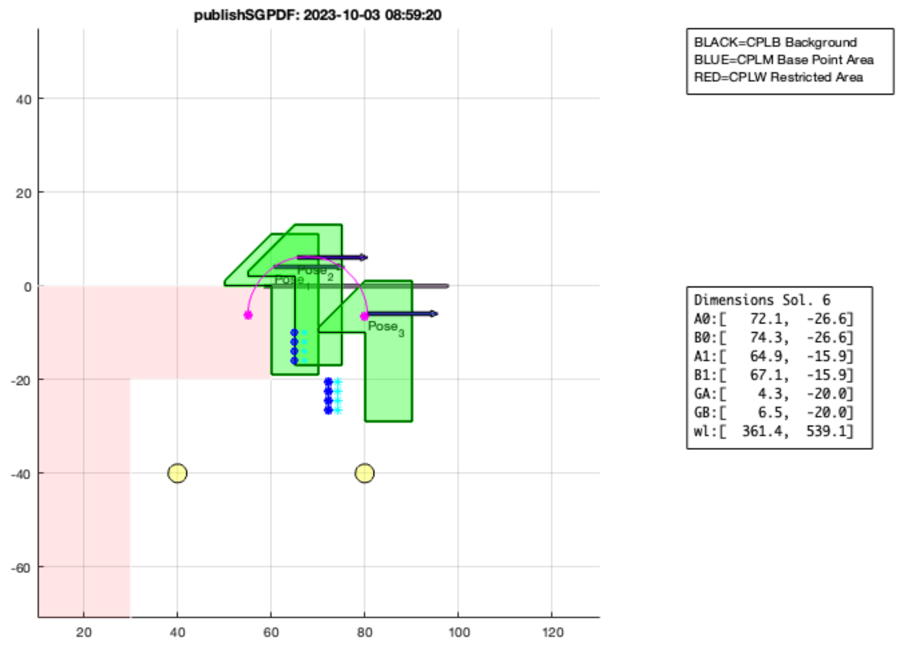


```
Dimensions Sol. 11
A0:[   72.1,   -2.6]
B0:[   74.3,   -2.6]
A1:[   64.9,    8.1]
B1:[   67.1,    8.1]
GA:[    4.3,    4.0]
GB:[    6.5,    4.0]
wl:[  361.4,  539.1]
```

TAKE THE LOWEST BASE POINTS AND USE THE LONGEST DISTANCE BETWEEN BASE POINTS

```
fourBarposesortsolution(PS,'base-posy',[-inf -20],'ground-length','flip'); PS=ans;
```

```
ans =
     '[2.9; 2.9; 2.2; 2.2; 2.2; 2.2]'
Poseplotsolution: Plot a limited selection of 6 of 6 solutions
```

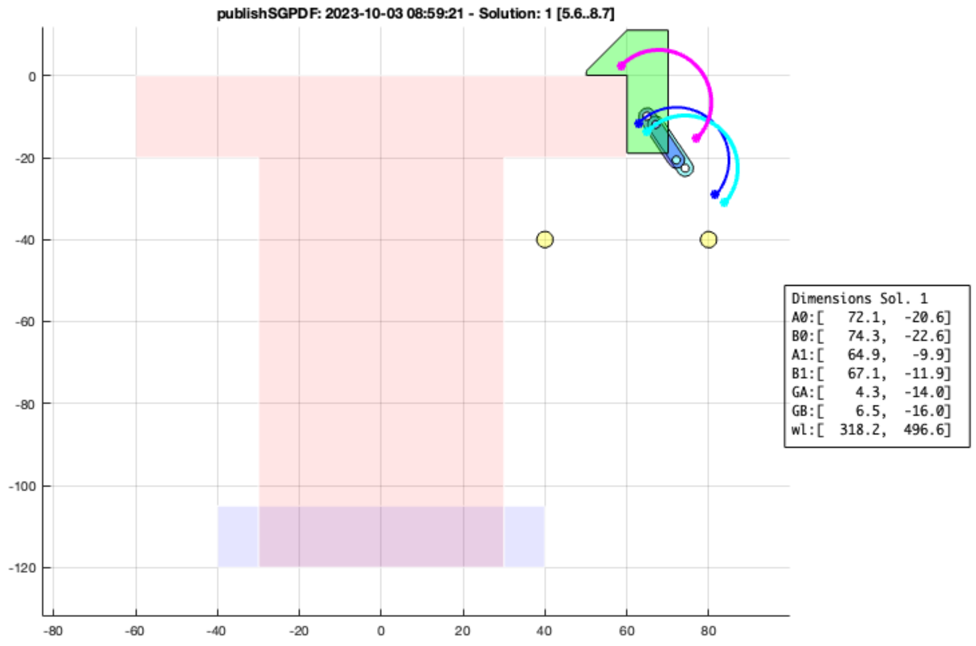


```
Dimensions Sol. 6
A0:[   72.1,  -26.6]
B0:[   74.3,  -26.6]
A1:[   64.9,  -15.9]
B1:[   67.1,  -15.9]
GA:[    4.3,  -20.0]
GB:[    6.5,  -20.0]
wl:[  361.4,  539.1]
```

ANIMATE THE SOLUTION 1

```
SGfigure; Poseplotspace(PS); fourBarposeplotsolution(PS,1,'animate');
```

```
checkfourbarsegment: Warning-Singularity - Crank touches inner workspace limit of 2R-swing!
```

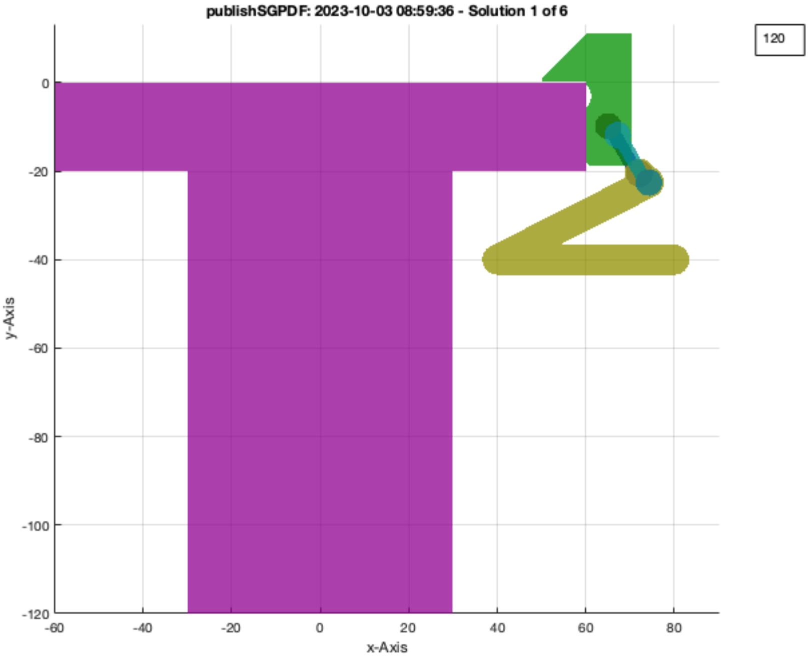


```
Dimensions Sol. 1
A0:[   72.1,  -20.6]
B0:[   74.3,  -22.6]
A1:[   64.9,   -9.9]
B1:[   67.1,  -11.9]
GA:[    4.3,  -14.0]
GB:[    6.5,  -16.0]
wl:[  318.2,  496.6]
```

LAYER THE LINKAGE

```
fourBarposelayering(PS,1,[3 1.6 3],[0 1 -1 2],'wlim',[0 165]); CLLL=ans;
```

```
fourBarposelayering: We use solution #1 of 6 solutions
fourBarposelayering: The links have a radius of 3.0mm and a height of 3.0mm and we use 1.6M screws!
fourBarposelayering: We use Levels [0,1,-1,2,-2] for layering of "Ground", "Effector", "Crank", "Swing", "World"
fourBarposelayering: Crank and swing move on different sides of the ground rack
fourBarposelayering: Crank rotation angle of solution 1 is 2.88(rad) is 165º
CPLrack4PL: WARNING: SOME POINTS ARE NEARER THAN 2*b AND MAY OVERLAP!
CPLrackPLdelaunay: WARNING: SOME POINTS ARE NEARER THAN 2*b AND MAY OVERLAP!
fourBarposelayering: DESIGN THE EFFECTOR
fourBarposelayering: DESIGN THE CRANK
PLshortestpathinCPLcost: CONTOUR BUFFER IS 3.0mm AND SEARCH RESOLUTION IS 0.3mm
fourBarposelayering: DESIGN THE SWING
PLshortestpathinCPLcost: CONTOUR BUFFER IS 3.0mm AND SEARCH RESOLUTION IS 0.3mm
Elapsed time is 2.011125 seconds.
```



DESIGN THE LINKAGE FOR 3D PRINTF AND ASSEMLY

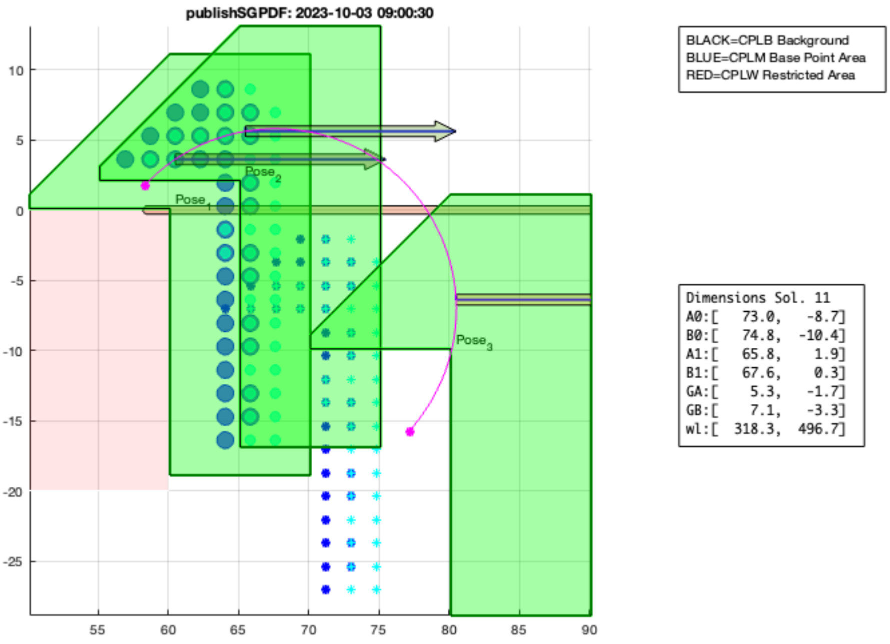
```
fourBarCLLL2SGdesign(CLLL,[3 1.6 3],'assembly','DIN7991');


%%NOW A DIFFERENT LAYERING
PosesampleHook; PS=ans; fourBarposesyntheses(PS,[1 2 3],[16 2.5 2.5],[1 2 3],[1 3]);PS=ans;
```

```
fourBarCLLL2SGdesign: Assembly Method is "DIN7991" using M1.6 x 6mm
Warning: Screw length and solid height may cause boolean trouble
fourBarCLLL2SGdesign:  >>>>>>>>>>>>> WARNING: AT LEAST ONE OF THE SPACE IS TOO SHORT <<<<<<<<<<<<<<<
fourBarCLLL2SGdesign: LAYERS USED FOR ASSEMBLY: [    0,    1,   -1,    2,   -2]
fourBarCLLL2SGdesign: Desing assembly method between solids
fourBarCLLL2SGdesign: Implement assembly method at all solids
SGboolh: "SGboolh('-','A','B')" called during "SGsubtract" failed: Solid A is unchanged
Warning: "SGsubtract(SGG,N)" FAILED in "<a href = "matlab:
SGopentoline('fourBarCLLL2SGdesign.m',277);">fourBarCLLL2SGdesign</a>"
SGboolh: "SGboolh('-','A','B')" called during "SGsubtract" failed: Solid A is unchanged
Warning: "SGsubtract(SPSG,NT)" FAILED in "<a href = "matlab:
SGopentoline('fourBarCLLL2SGdesign.m',292);">fourBarCLLL2SGdesign</a>"
SGboolh: "SGboolh('-','A','B')" called during "SGsubtract" failed: Solid A is unchanged
Warning: "SGsubtract(SPCE,NT)" FAILED in "<a href = "matlab:
SGopentoline('fourBarCLLL2SGdesign.m',295);">fourBarCLLL2SGdesign</a>"
fourBarCLLL2SGdesign: Implement assembly method for rack points
SGwriteMultipleSTL: Writing 6 STL files in <a href = "matlab: openbydoubleclick ('/Users/timlueth/Desktop/STLmult SGofCLLL 2023-10-03-0900')">/Users/ti
fourBarCLLL2SGdesign: ASSEMBLY of 6 parts:
% [2023-Oct-03 09:00:12 by timlueth]
% Required for production (ANYCUBIC-PREDATOR Printer) are the following SG-Lib functions and standard parts:
   6 * DIN965 or DIN7991 M1.6 x 6mm (screw)

fourBarposesyntheses: Number of Gridpoints should be: 16
PoseaddGPL: CPLE 2.5 mm Buffer for 15 attachment points
fourBarposesyntheses: 3-Pose-Synthesis using poses: [1,2,3]
PoseaddGPL: CPLE 2.5 mm Buffer for 58 attachment points
checkfourbar3Poseattachpermutation: Original solution format "PS.GAL" (3Poses) converted into general pose solution format: "PS.solut"
fourBarposesyntheses: 288 solution were found using 3-Pose-Synthesis.
fourBarposesyntheses: 163 solution were left after forcing a pose order
fourBarposesyntheses: Now forcing an angle limit [1,3]
fourBarposesyntheses: 163 Solutions are angle limited to [1,3]
```

```
Poseplotsolution: Plot a limited selection of 10 of 163 solutions
```

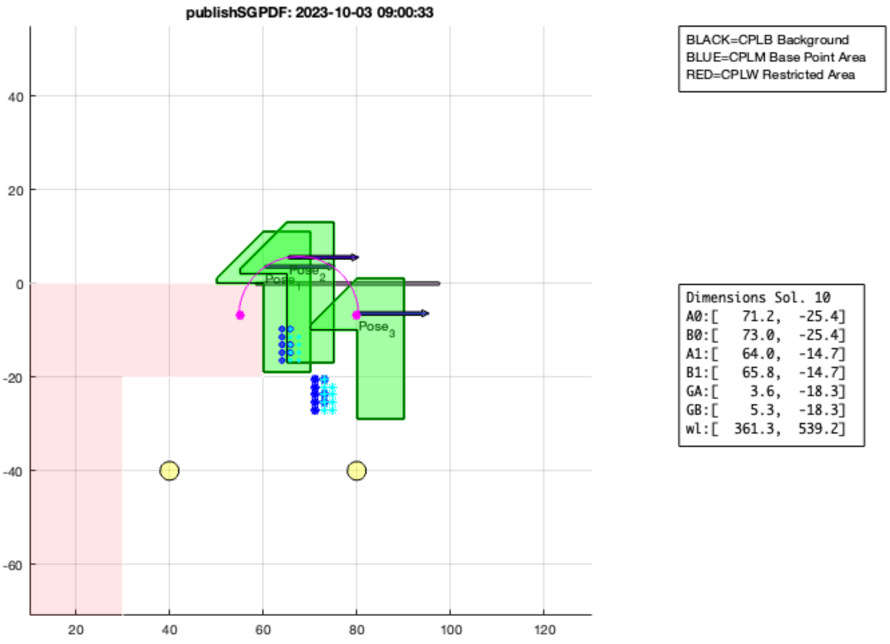


```
Dimensions Sol. 11
A0:[   73.0,    -8.7]
B0:[   74.8,   -10.4]
A1:[   65.8,     1.9]
B1:[   67.6,     0.3]
GA:[    5.3,    -1.7]
GB:[    7.1,    -3.3]
wl:[  318.3,   496.7]
```

Sort the solutions

```
fourBarposesortsolution(PS,'base-posy',[-inf -20],'ground-length','flip'); PS=ans;
```

```
ans =
    '[6.1; 6.1; 4.9; 4.9; 4.9; 3.9; 3.9; 3.9; 3.9; 2.4; 2.4; 2.4; 2.4; 2.4; 2.4; 1.8; 1.8; 1.8; 1.8; 1.8]'
Poseplotsolution: Plot a limited selection of 10 of 20 solutions
```



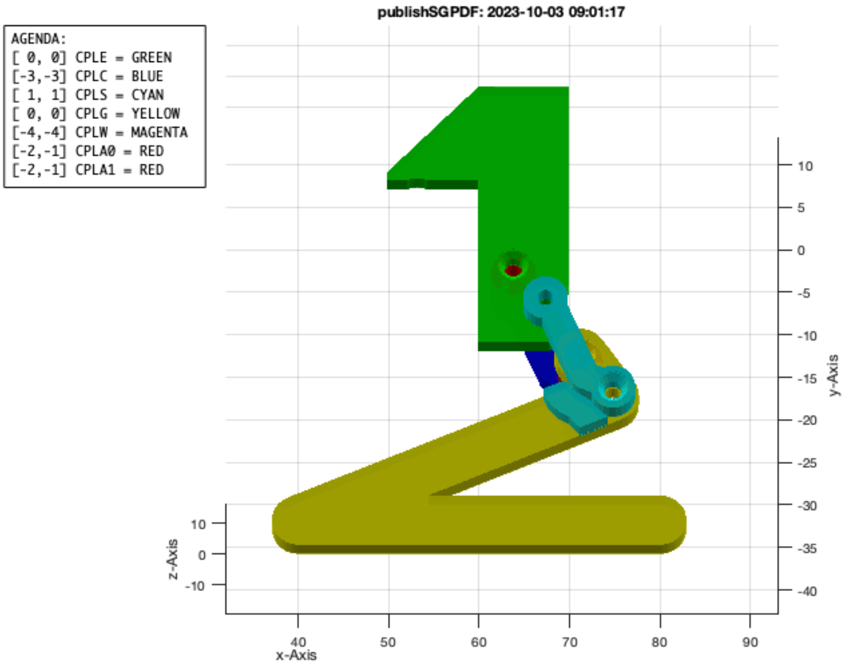
```
Dimensions Sol. 10
A0:[   71.2,   -25.4]
B0:[   73.0,   -25.4]
A1:[   64.0,   -14.7]
B1:[   65.8,   -14.7]
GA:[    3.6,   -18.3]
GB:[    5.3,   -18.3]
wl:[  361.3,   539.2]
```

Layering Effector and Rack in the same layer

```
fourBarposelayering(PS,1,[2.5 1.6 3],[0 0 -3 1],'wlim',[20 165]); CLLL=ans;

fourBarCLLL2SGdesign(CLLL,[2.5 1.6 3],'assembly','DIN7991');
```

```
fourBarposelayering: We use solution #1 of 20 solutions
fourBarposelayering: The links have a radius of 2.5mm and a height of 3.0mm and we use 1.6M screws!
fourBarposelayering: We use Levels [0,0,-3,1,-4] for layering of "Ground", "Effector", "Crank", "Swing", "World"
fourBarposelayering: Crank and swing move on different sides of the ground rack
fourBarposelayering: Crank rotation angle of solution 1 is 2.78(rad) is 159º
fourBarposelayering: DESIGN THE EFFECTOR
fourBarposelayering: DESIGN THE CRANK
PLshortestpathinCPLcost: CONTOUR BUFFER IS 2.5mm AND SEARCH RESOLUTION IS 0.4mm
fourBarposelayering: DESIGN THE SWING
PLshortestpathinCPLcost: CONTOUR BUFFER IS 2.5mm AND SEARCH RESOLUTION IS 0.4mm
Elapsed time is 1.907602 seconds.
fourBarCLLL2SGdesign: Assembly Method is "DIN7991" using M1.6 x 6mm
Warning: Screw length and solid height may cause boolean trouble
fourBarCLLL2SGdesign: LAYERS USED FOR ASSEMBLY: [    0,    0,   -3,    1,   -4]
fourBarCLLL2SGdesign: Desing assembly method between solids
fourBarCLLL2SGdesign: Implement assembly method at all solids
fourBarCLLL2SGdesign: Implement assembly method for rack points
SGwriteMultipleSTL: Writing 6 STL files in <a href = "matlab: openbydoubleclick ('/Users/timlueth/Desktop/STLmult SGofCLLL 2023-10-03-0901')">/Users/ti
fourBarCLLL2SGdesign: ASSEMBLY of 6 parts:
% [2023-Oct-03 09:01:12 by timlueth]
% Required for production (ANYCUBIC-PREDATOR Printer) are the following SG-Lib functions and standard parts:
   6 * DIN965 or DIN7991 M1.6 x 6mm (screw)
```



Published with MATLAB® R2023a