How to run the AR-HF Forward Modeling Code

1. Introduction

1.1. Purpose

The forward modeling code computes all possible two-dimensional shapes generated from a solid by either of two processes. One process projects a silhouette of the solid onto a plane, which is termed Plane of Projection. The other process, termed Plane of Section, examines the intersection of planes with the solid. For each possible two-dimensional shape, two variables, aspect ratio (AR) and Heywood factor (HF), are computed. These data are then used to determine the probability of any given pair of (AR, HF) value being generated from a given solid.

1.2. Operational Overview

The solid is drawn using 3D modeling software and stored in objective file format (OFF). The MATLAB routine, off2arhf, reads one or more OFF files in a directory. Other variables the user specifies whether Plane of Projection of Plane of Section models are to be computed, the number of points on sphere, maximum number of slices, and whether on not a data formatted for gnuplot is to be output.

1.3. Theory

For a discussion of the theory and algorithm logic, see Rickman, Lohn-Wiley and Knicely, 2014, Probabilistic Particle Shape Measurement.

1.4. Credits

This package incorporates work by Brian Hannan and Blake Lohn-Wiley. The work was done over the summer of 2014, as summer interns at NASA/Marshall Space flight Center, while under the direction of Doug Rickman. This code is an evolution of software written by Joshua Knicely and Blake Lohn-Wiley under the direction of Doug Rickman during the summer of 2013. Subsequently D. Rickman added capability to write files needed by gnuplot.

2. Required Software

- MATLAB
- SketchUp (www.sketchup.com) or another CAD package that can export in OFF format
- the forward_modeling_code folder
- gnuplot (http://www.gnuplot.info) if plots are desired

3. Setup

Download the forward modeling folder and all of its subdirectories to your computer.

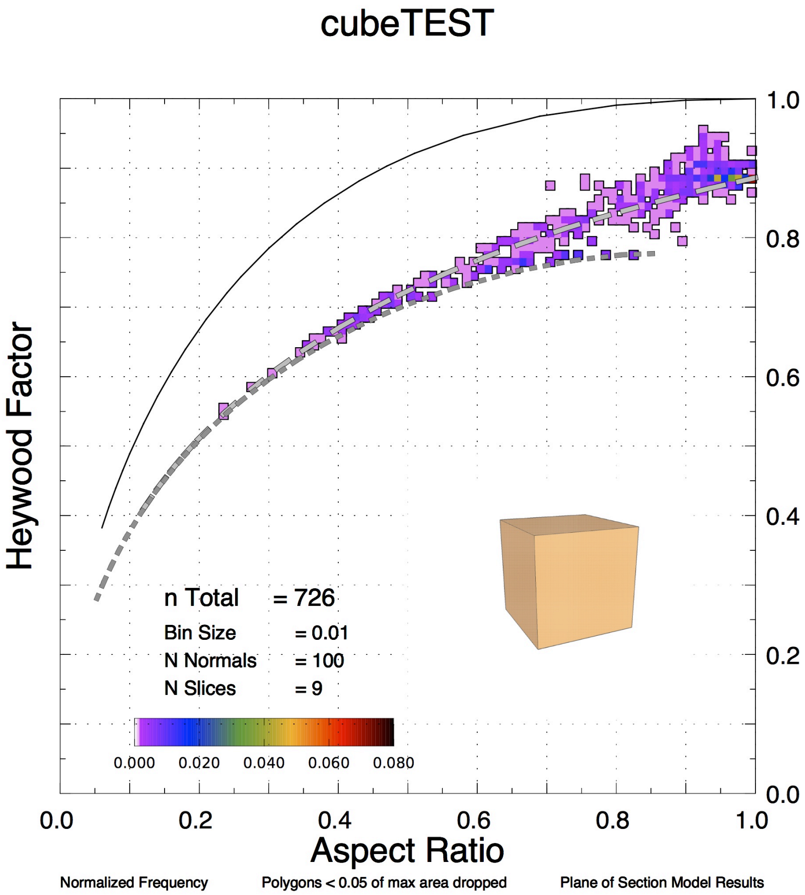
Once you have the forward modeling code on your machine, add this directory to the MATLAB search path. To do this launch MATLAB. Then go to File->Set Path...- >Add with Subfolders... and select the forward modeling folder. Select Open and then click Save.

3.1. Testing

In the provided directory “Debugging” are the files:

- cubeTEST.skp
- cubeTEST.png
- cubeTEST.off
- cubeTEST_sec_100norms_9slices_PofSect.txt
- cubeTEST_sec_100norms_9slices_PofSect.plt
- cubeTEST_sec_100norms_9slices_PofSect.pdf

The .skp file was used to generate the .png and the .off files. The .off was the source file used to create a Plane of Section model with 100 normals and 9 slices. The output of the model was the .plt and .txt files. These, with the .png file was used to make the following illustration. Note, to use the .plt file as is, it is necessary to place it, the .txt and the .png files in the directory indicated by the path1 and path2 statements at the beginning of the .plt file.



4. Using off2arhf

Within the MATLAB user environment, the user runs a routine called “off2arhf” to create MAT file(s) containing AR-HF data for one or more OFF files. A minimum of three input arguments are required for plane of projection. Plane of section requires a minimum of four input arguments.

The required input arguments are listed below.

- ***directoryString*** A string (in single quotes) containing the full path to a directory. This must be the first input argument.
- ***secOrProj*** A case-insensitive string that indicates whether plane of projection or plane of section data is desired. This must be the second input argument.
  - Set ***secOrProj*** equal to 'proj' or 'projection' to generate plane of projection data.
  - When ***secOrProj*** equals 'sec' or 'section', plane of section data is generated.
- ***numSpherePoints*** The integer number of points to be distributed on the sphere (the number of normals). This must be the third input argument.
- ***numSlicesPerNormal*** The integer number of slices per normal. If plane of section data is desired, ***numSlicesPerNormal*** must be the fourth input argument. This input argument should only be supplied for plane of section.

The optional input arguments, which can be supplied in any order following the required arguments, are:

- ***'FileName', fName*** A property name-value pair input. When this name-value pair is provided, off2arhf will generate AR-HF data only for the OFF file ***fName*** located in the directory ***directoryString***. ***fName*** is a string (in single quotes) that provides the name of the OFF file. ***fName*** must contain the -.off extension. The single quote marks are required.
- ***'OutputGnu'*** When the case-insensitive ***'OutputGnu'*** flag is supplied, PLT and TXT files will be generated for gnuplot figure generation. The single quote marks are required.

For testing, run off2arhf on a file in the testfolder directory (located inside forward_modeling_code/data_output_files/). To simplify the following examples a variable is first defined which contains the value of ***directoryString***.

```
mydir = '/[YOUR PATH HERE]/data_output_files/testfolder/'
```

The following command will run plane of section code with 100 normals and 10 sections per normal for the OFF file named block.off in the directory specified in mydir.

```
off2arhf(mydir,'sec',100,10,'FileName','block.off')
```

To generate plane of projection data for numSpherePts=100 normals on the same OFF file, use the command

```
off2arhf(mydir,'proj',100,'FileName','block.off')
```

The command below tells off2arhf to generate plane of section data and to additionally generate the PLT and TXT files required by gnuplot.

```
off2arhf(mydir,'sec',100,10,'FileName','block.off','OutputGnu')
```

To automate the generation plane of section data for all OFF files in the mydir directory, use

```
off2arhf(mydir,'sec',100,10)
```

5. Creating Solids

Many 3D design packages can be used to define solids. The following discusses doing this with SketchUp.

5.1. Exporting SketchUp drawings in OFF format

After installing SketchUp, download the SketchUp plugin, off_exporter.rb. This may be downloaded from http://www.cs.princeton.edu/courses/archive/spr08/cos426/assn2/off_exporter.rb Place off_exporter.rb in the SketchUp 2014->SketchUp->Plugins folder. More info on

locating this folder can be found at http://help.sketchup.com/en/article/38583 To export a SketchUp drawing, select Plugins->Export Off.

5.2. Drawing the Solid

The forward modeling code requires that the solid be represented by a closed surface and assumes that there are no interior faces. (An ”interior face” is a polygon that passes through the volume enclosed by the boundary of the closed surface.) However, interior faces are sometimes formed in a solid by SketchUp. This will cause an error in the plane of section code because a non-ideal (non-manifold) triangle mesh will be formed during import. Specifically, ideal meshes have at most 2 incident faces per edge). It is therefore necessary to check for interior faces before exporting the SketchUp drawing.

Figure 1 shows a beveled cube that was drawn in SketchUp. This is contained in the file the file Cube_Beveled Initial.skp. Figure 2 shows the solid with several of the exterior faces hidden, so internal structure may be seen. Faces that are selected may be hidden by right clicking and using the Hide option. To unhide a hidden face, turn on the “Hidden Geometry” option in the view menu. The hidden face my be then be selected and a right click will show the Unhide option. After deleting any interior faces, the drawing can be exported using the OFF format.

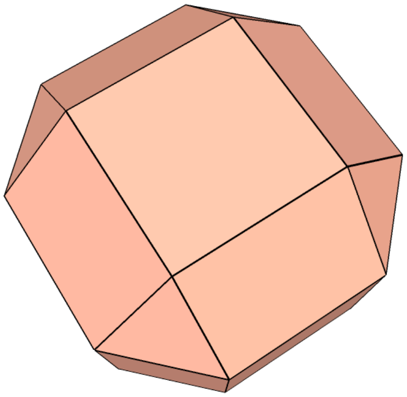


Figure 1: Initial SketchUp drawing showing exterior faces only.

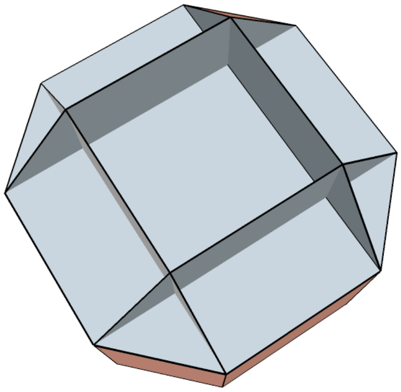


Figure 2. View after hiding several exterior faces.

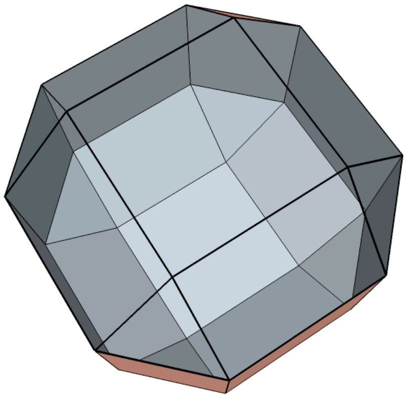


Figure 3. View after removal of interior faces.

5.3.Image file

It can be useful to have a reference image showing the solid. This may be made in SketchUp using the File – Export - 2D Graphic menu. Export should be as a .png file. Under options for the .png set the output file width to 1000. This will make the image suitable for inclusion in a plot made by gnuplot.

5.4.OFF file format

Objective File Format files begin with the keyword OFF. The next line holds three integers: the number of vertices, the number of edges, and the number of faces. A list of the Cartesian coordinates of all vertices follows (one per line). Next, faces are listed (one per line). For a triangle mesh, a given face will be listed as three indices that point to the three vertices that form the triangular face.

6. A note on distributing points on the sphere

PointsOnSphereEqually throws an error the first time (and only the first time) it runs in a new MATLAB session. The function requires the creation of a java object to run, and it doesn't seem to want to do this on the first try. Brian has temporarily stopped using it to avoid problems with automated data generation.

Brian commented the call to PointsOnSphereEqually at lines 97-98 and replaced it with PointsOnSphere at lines 99-100 until the error can be avoided. If you would like to use PointsOnSphereEqually, modify off2geomstruct so that it no longer calls PointsOnSphere and run PointsOnSphereEqually once before generating your AR- HF data.

Inside PointsOnSphereEqually, a path is set that points to the location of the a JAR file. Be sure to modify this path before use. The line that requires modification is line 46 of PointsOnSphereEqually.m (in the common_code folder).

7. matGeom

This code uses some functions from the matGeom package version 1.1.9 (released Feb. 2014). Visit the matGeom page, http://matgeom.sourceforge.net/wiki/index.php/MatGeom

The matGeom functions used are

- centroid.m
- createLine3d.m
- createPlane.m
- distanePointLine3d.m
- distancePointPlane.m
- intersectEdgePlane.m
- meshEdges.m
- normalizeVector3d.m
- readMesh_off.m
- vectorCross3d.m
- vectorNorm3d.m

These functions are included in the forward_modeling_code folder.

8. Running gnuplot

In order to produce high quality graphics gnuplot, version 4.6, is used. For this reason off2arhf writes two ascii files, one with data and the other with a command script to run gnuplot. By running the script from the command line, gnuplot will produce a PDF of the plot. The names of all three files is determined by the name of the OFF file with the extension ".txt", ".plt", and ".pdf" respectively.

8.1. .TXT file

In the .txt file there are comment lines, defined by a leading "#", containing information about the data. Then there are six sets of data. The following is an example of the comment lines for a dipyramidal trigonal prism.

```
# Plane of Section Model Results  -  Formatted for gnuPLOT
# Corresponding gnuPLOT script file =
  "Dipyramidal_Trigonal_Prism_1_1_PofSect.plt"
#
# Dipyramidal Trigonal Prism 1 1
# Model execution time = 19.8475 seconds
# Number of Surface Normals : 100
# Slice Spacing : 10
# Polygons smaller than 0.05 of the largest polygon are dropped.
# Number of data points = 639
# Binning Interval = 0.01
# Max frequency in a bin                     = 0.025039
# Frequency at which cumulative ~50% occurs   = 0.0031299
# Frequency at which cumulative ~75% occurs   = 0.0015649
# Frequency at which cumulative ~95% occurs   = 0.0015649
# Cum Freq for all bins with 5 or more points = 0.10016
#
# Cumulative Frequency within frequency 0.0015649, which is contour 1 =
  0.95149
# Cumulative Frequency within frequency 0.0062598, which is contour 2 =
  0.16745
# Cumulative Frequency within frequency 0.010955, which is contour 3 =
  0.034429
# Cumulative Frequency within frequency 0.015649, which is contour 4 =
  0.034429
# Cumulative Frequency within frequency 0.020344, which is contour 5 =
  0.034429
#
# Data Block 1: Ellipse, rectangle and triangle lines
# Data Block 2: Binned and normalized data, structured for contouring
# Data Block 3: Binned and normalized data with cumulative frequency >= 95%
# Data Block 4: Binned and normalized data with cumulative frequency >= 75%
# Data Block 5: Binned and normalized data with cumulative frequency >= 50%
# Data Block 6: Individual Aspect Ratio and Heywood points.
```

8.2. .PLT file

The details of the .plt file are beyond the scope of this document. An example .plt is included here with a few additional comments. For syntax and functional significance of individual commands please see the gnuplot documentation.

Please note, the .plt file continues to evolve as we develop our work. This version may or may not match exactly with the .plt generated by your installation.

Comment – In a gnuplot script, where a "#" is found defines the start of a comment. The comment continues to the end of the line.

```
# - - - - - Introduction - - -  - - - - - - - - - - - - - - - - -
# Plane of Section Model ResultsFor the solid Dipyramidal Trigonal Prism 1 1
# Graph Relative Frequency of Aspect Ratio vs Heywood Factor using gnuplot.
# The path is set to /Users/doug2/Documents/MATLAB/.  Edit the "path1" and
  "path2 lines to change this.
# This script designed by Doug Rickman, Jan 30, 2014, mod: July 3, 2014
# set terminal pdf enhanced dashed dl 3 size 7,7
```

```
set terminal pdfcairo enhanced color dashed dashlength 3 size 7,7   # The
  size affects the dimensions of the plot.
```

Comment – Macros are used to simplify both manual editing of the script and its original creation. This way file names and paths can be defined at the front of the script. Then actual commands can refer to the definitions made using the macros.

```
# - - - - - Define Macro Terms. Set Input and Output Files - -
set macros
dquote    = '"'
path1     = "/Users/doug2/Documents/"
path2     = "MATLAB/"
outfile   = "Dipyramidal_Trigonal_Prism_1_1_PofSect.pdf"
infile    = "Dipyramidal_Trigonal_Prism_1_1_PofSect.txt"
set output @dquote@path1@path2@outfile@dquote

# - - - - - Define Line Styles -- - - - - - - - - - - - - - - - -
set style line 1  lt 0  lw 3  lc rgbcolor 'black'   # dot
set style line 2  lt 1  lw 3  lc rgbcolor 'black'   # solid
set style line 3  lt 2  lw 10 lc rgbcolor 'grey70'  # long dash
set style line 4  lt 3  lw 10 lc rgbcolor 'grey50'  # short dash
set style line 5  lt 4  lw 3  lc rgbcolor 'black'   # long short long short
set style line 6  lt 5  lw 3  lc rgbcolor 'black'   # long short short long
set style line 13 lt 2  lw 12 lc rgbcolor 'black'   # long dash
set style line 14 lt 3  lw 11 lc rgbcolor 'black'   # short dash
```

Comment – The plots are actually 3D surfaces, though they are not shown in a way that this is obvious. The general controls configure various parameters that may be used by subsequent plot operations.

```
# - - - - - Set General Controls - - - - - - - - - - - - - - - -
set grid                        # grid and cntrparam are used with contours
set cntrparam bspline
set cntrparam order 10
set view equal xyz              # View establishes the orientation and visual
  geometry of the plot
set view 0,0,1.5,1              # This affects the dimensions of the plot.
unset xlabel                    # Remove any existing X, Y, Z axes labels
unset ylabel
unset zlabel
unset key                       # Do not display a key
unset zrange                    # Free the range of Z values
unset colorbox                  # Turn off the color box
set xrange [0:1]                # Set the X range of the plot
set yrange [0:1]                # Set the Y range of the plot
set zrange [0:100]              # Set the Z range of the plot
unset ztics                     # Turn off tics on the Z axis
set mxtics 2                    # Define minor tics frequency on X axis
set mytics 2                    # Define minor tics frequency on Y axis
set multiplot                   # Allow multiple plots in the same space
set size 0.95, 0.95             # Set the size of the plot
set origin 0.0, 0.01            # Set the starting location of the plot
```

Comment – The labels are automatically generated in off2arhf based on the user defined options.

```
# - - - - - Define the Labels - - - - - - - - - - - - - - - - - -
```

```
set label 1 'Dipyramidal Trigonal Prism 1 1'           font 'Helvetica,30'
   at  0.5 ,  1.1 , 0 center
set label 2 'n Total   {\011}= 639'  font 'Helvetica,20' at  0.15,  0.23 , 0
set label 3 'Bin Size  {\011}= 0.01' font 'Helvetica,17' at  0.15,  0.18
set label 4 'N Normals{\011}= 100'   font 'Helvetica,17' at  0.15,  0.14
set label 5 'N Slices  {\011}= 10'   font 'Helvetica,17' at  0.15,  0.10
set label 6 'Aspect Ratio'           font 'Helvetica,28' at  0.5 ,  -0.09, 0
   center
set label 7 'Heywood Factor'         font 'Helvetica,28' at -0.05 ,  0.5 , 0
   center rotate by 90
set label 8 'Normalized Frequency'   font 'Helvetica,12' at screen 0.10,
   0.02, 0 left
set label 9 'Polygons < 0.05 of max area dropped' font 'Helvetica,12' at
   screen 0.45,  0.02, 0 center
set label 10 'Plane of Section Model Results' font 'Helvetica,12' at screen
   0.64,  0.02, 0 left

# - Create grid lines and tics every 0.2 intervals on both axes -
set xtics 0.2 format "%.1f" font 'Helvetica,20' offset -0.35,-0.3
set ytics 0.2 format "%.1f" font 'Helvetica,20' offset -0.35,-0.3

set surface                     # Show the data as a surface
unset contour                   # Do not show the data as contours
```

Comment – The .plt file can be used to make a large number of different plots. Which plots are produced is controlled by uncommenting and commenting the following lines.

```
# - - - - - Choose Which Data To Plot And How - - - - - - - - - -
# - Later Plots Overwrite Earlier Plots BUT DO NOT ERASE THEM - -
# - -  - Plots may be reordered by changing their sequence. - - -

# To plot the individual data points turn on the following line.
# splot @dquote@path1@path2@infile@dquote  index 5 using 1:2:3 with points
   pointtype 7 lc rgbcolor 'dark-green'  pointsize 0.2 # filled circles

# In the 95, 75 and 50% plots, the data are plotted twice, to create an
   outline effect.
# The bins are plotted as points using a hollow square symbol!!!
# The point size of each of a pair of plots is scaled to match the bin size
   used and the dimensions of the plot.
# The first plot of a pair is larger and creates an black outline effect for
   the data cloud.
# The plotted symbols are shifted so they align graphically on the bins in
   the plot.
# If something changes the dimensions of the plot, the size of the symbols
   must be changed.
```

Comment – The data produced by MATLAB are actually bin boundaries. Gnuplot is using the boundaries as though they are points. Therefore, it is necessary to translate the MATLAB values to the center of the bin as plotted by gnuplot. This is done by shifting the MATLAB data one half the width of the bin, which in this case is 0.005.

```
# To plot the bins containing 95% of the data.
splot @dquote@path1@path2@infile@dquote  index 2 using
   ($1+0.005):($2+0.005):($3) with points pointtype 5 lc rgbcolor 'black'
   pointsize 0.8    # Hollow squares, used to outline
```

```
splot @dquote@path1@path2@infile@dquote  index 2 using
   ($1+0.005):($2+0.005):($3) with points pointtype 5 lc rgbcolor
   'lemonchiffon'  pointsize 0.60    # solid squares

# To plot the bins containing 75% of the data
splot @dquote@path1@path2@infile@dquote  index 3 using
   ($1+0.005):($2+0.005):($3) with points pointtype 5 lc rgbcolor 'royalblue'
   pointsize 0.78 # solid squares
splot @dquote@path1@path2@infile@dquote  index 3 using
   ($1+0.005):($2+0.005):($3) with points pointtype 5 lc rgbcolor 'grey60'
   pointsize 0.60 # solid squares

# To plot the bins containing 50% of the data
splot @dquote@path1@path2@infile@dquote  index 4 using
   ($1+0.005):($2+0.005):($3) with points pointtype 5 lc rgbcolor 'light-red'
   pointsize 0.78 # solid squares
splot @dquote@path1@path2@infile@dquote  index 4 using
   ($1+0.005):($2+0.005):($3) with points pointtype 5 lc rgbcolor 'grey20'
   pointsize 0.60 # solid squares

# To plot the binned data as contours turn on the following 9 lines.
# unset surface
# set contour
# set cntrparam levels incremental 0.00156, 0.00469, 0.025
# set cbrange [0.0:0.025039]
# set label 11 'Contour (B,I,E)= 0.00156, 0.00469, 0.025'   font
   'Helvetica,17' at  0.15,  0.06
# splot @dquote@path1@path2@infile@dquote  index 1 using
   ($1+0.005):($2+0.005):($3) w l lw 3 palette
# set surface
# unset contour
```

Comment – In the following, the color assigned to a bin is controlled by the value of the bin, the palette, and the scaling from bin values to the palette range. The scaling is done in the "set cbrange" command. The values used by cbrange default from zero to the maximum observed value for the data set being plotted. It can be useful to change the maximum to a slightly larger, evenly divisible value and then change the value of cbtics to a value that is exactly 1/2 to 1/4 of the new maximum. For this example, the value of 0.025039 might be changed to 0.026 and the 0.002 changed to 0.013.

```
# To plot the normalized, binned data as a greyscale or a colored surface
   first turn on 1 of the following 2 lines.
# set palette defined (  0 "white", 0.1 "grey", 5 "black")   # Use this line
   for a greyscale plot.
# set palette defined (  0 "white", 0.1 "light-magenta", 0.1 "dark-magenta",
   1 "blue", 2 "forest-green", 3 "orange", 4 "red", 5 "black") # Use this line
   for colored plot.
# Second, turn on the following 5 lines.
# set colorbox horizontal user origin 0.18,0.17 size 0.28,0.03
# set cbrange [0.0:0.025039]
# set cbtics 0.002 format "%.3f" # Adjust the tic interval from 0.002 as
   needed
# splot @dquote@path1@path2@infile@dquote  index 1 using
   ($1+0.005):($2+0.005):($3>0.00000 ? $3 : NaN) with points pointtype 5 lc
   rgbcolor "black"  pointsize 0.78
```

```
# splot @dquote@path1@path2@infile@dquote  index 1 using
   ($1+0.005):($2+0.005):($3>0.00000 ? $3 : NaN) with points pointtype 5
   palette  pointsize 0.68

# - - - - - Do the Ellipse, Rectange and Triangle Lines - -  - -
# To plot the lines for ellipses, rectangles and triangles turn on the
   following lines.
set format xy""
set xtics 0.1
set ytics 0.1
splot @dquote@path1@path2@infile@dquote   index 0 using 1:2:7 w l ls 2  #
   Ellipse
splot @dquote@path1@path2@infile@dquote   index 0 using 3:4:7 w l ls 13 #
   Rectangle
splot @dquote@path1@path2@infile@dquote   index 0 using 3:4:7 w l ls 3  #
   Rectangle
splot @dquote@path1@path2@infile@dquote   index 0 using 5:6:7 w l ls 14 #
   Isoceles triangle
splot @dquote@path1@path2@infile@dquote   index 0 using 5:6:7 w l ls 4  #
   Isoceles triangle
unset surface
set contour
```

Comment – The following will place a picture of the solid, saved in the file “my_solid.png”, in the lower right quadrant of the plot. The picture may be generated in SketchUp. The following parameters work well if the .png file is approximately 1000 pixels wide.

```
set style rectangle back fc rgb "white" fs solid border
set object 101 rectangle from screen 0.45, 0.2, 0 to screen 0.75, 0.45, 0
set cbrange [*:*]
splot "my_solid.png" binary filetype=auto origin=(0.44,0.1,0) dx=0.0005
   dy=0.0005 with rgbimage

unset multiplot                  # This causes all of the plots defined above
   to be rendered into the output file.
exit
```

8.3.Installation of gnuPLOT

It is necessary to have gnuplot properly installed and with the correct options. The script in the .plt file uses the terminal “pdfcairo” and employees the enhanced options in order to generate dashed line. This option must be selected during the installation process.

On the Mac, under OS X 9.4, we have found it easiest to install gnuplot using MacPorts, https://www.macports.org/ports.php?by=name&substr=gnuplot The variants used are pangocairo and pdflib

Further guidance about installing gnuplot is beyond the scope of this document.

9. Information

For additional information please contact

Doug Rickman
Earth Science Office MSFC/NASA
320 Sparkman Drive
Huntsville, Alabama 35805

256-961-7889
doug.rickman@nasa.gov

10. Dependencies for off2arhf

		Level								
	Level	1	2	3	4	5	6	7	8	9
1	1	off2arhf								
2	2		off2geomstruct							
3	3			PointsOnSphere						
4	3			centroid						
5	3			meshEdges						
6	3			readMesh_off						
7	3			mindist						
8	4				distmat					
9	3			convexity3d						
10	4				minConvexHull					
11	5					angleSort3d				
12	6						createPlane			
13	7							normalizePlane		
14	8								normalizeVector3d	
15	8								planeNormal	
16	9									vectorCross3d
17	8								projPointOnPlane	
18	7							vectorNorm3d		
19	6						planePosition			
20	5					isCoplanar				
21	4				triangulateFaces					
22	3			trans2origin						
23	2		generateGnuplotFiles							
24	3			write_gnuPLOT_data						

25	4	ARHF_Ellipse
26	4	ARHF_Rectangle
27	4	ARHF_Triangle
28	5	triangle_area
29	3	write_gnuPLOT_file
30	3	bin_data
31	4	ContourLimits
32	2	processGeomStruct
33	3	run_arhf
34	4	planeofprojection
35	5	aspectRatio
36	6	distancePointLine3d
37	6	polynormal
38	7	distFromLine
39	7	maxVerticesDist
40	5	convexity
41	6	rotate2xy
42	7	quatRotVerts
43	8	quatCalc
44	8	quatRotVec
45	5	heywood
46	5	solidity
47	4	planeofsection
48	5	xsecmesh
49	6	distancePointPlane
50	6	intersectEdgePlane
51	7	createLine3d
52	6	matroundsf
53	6	truncateverts
54	2	getdirconts

Toolboxes:

1 Mapping Toolbox (4.0) [map]

2 MATLAB (8.2) [matlab]

3 Statistics Toolbox (8.3) [stats]