Bastion Geotechnical Pty Ltd

Technical Note

Calculation of Span of Intersections

V1.0 – 191127 – JMS Initial compile.

V1.1 – 191127 – JMS Update to reflect GUI interface and executable software.

V1.2 – 191129 – JMS Update to reflect the csv exported from GEM4D will be imported as-is or if you have the old format, that can be imported as well. Also, noting increase in point size when digitizing the closed polygons will provide some ease.

Context

Geotechnical engineers use the span of underground openings to design ground support and reinforcement. Common methods of estimating span include measuring between two points from a mine plan or using a laser distance measure at the face underground. Typically, these points are not recorded for future use or audits.

Methodology

A methodology has been compiled to provide a location-based output for reporting intersection spans. The methodology presented here is demonstrated with the software GEM4D (www.basrock.net) and Python (www.python.org). Alternatives to GEM4D include Maptek Vulcan, Geovia Surpac and Vela Datamine amongst others.

1. Digitizing Intersection Polygons (GEM4D)

Figure 1: Typical underground drive strings. Plan view.

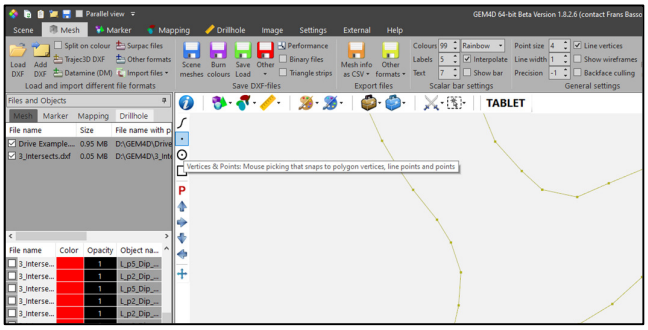


Figure 2: Displaying line vertices and setting snap mode to vertices and points. For easier digitizing, set the Point Size to >=10.

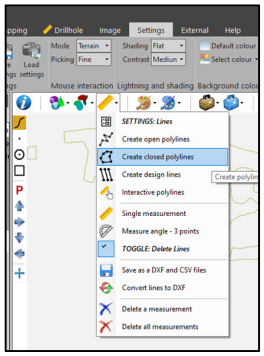


Figure 3: Selection menu for creating closed polygons.

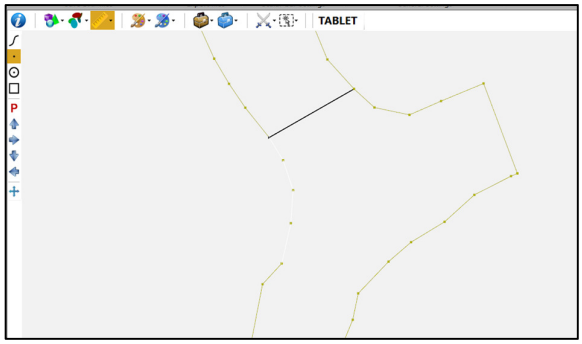


Figure 4: Digitizing the polygon. Right-click on the penultimate point when done. Please note to digitize the polygon to an adequate distance away from the intersection such that the maximum inscribed circle can be calculated. Visual inspection as per Figure 19 will provide an indication as to whether this has been completed adequately.

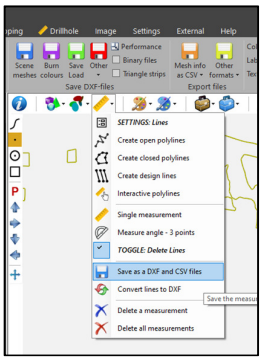


Figure 5: Selection menu for saving the digitized polygons as dxf and csv.

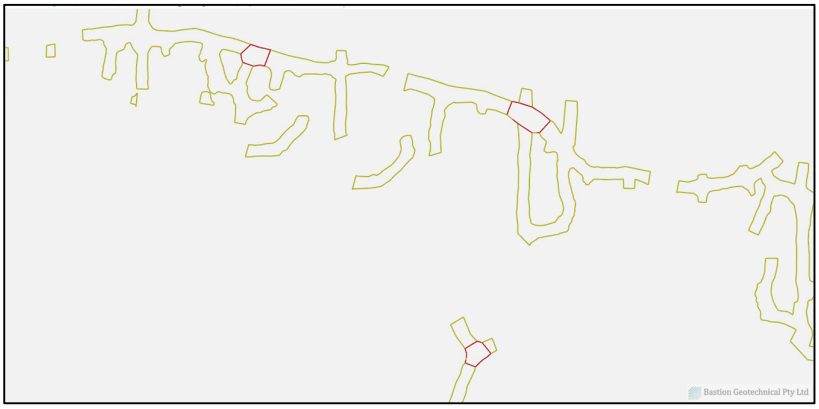
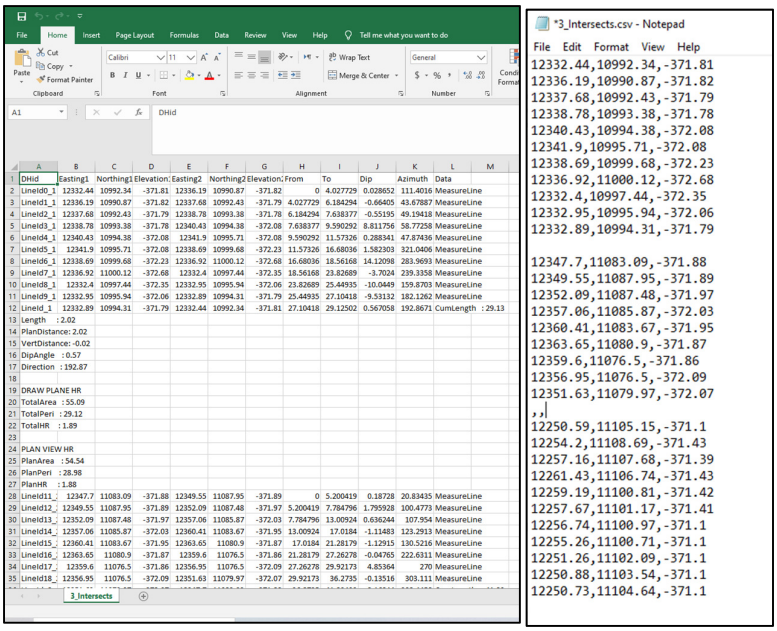


Figure 6: Three digitized polygons (red colour) loaded into GEM4D.

2. A note of the Input CSV File (Excel)



DHid	Easting1	Northing1	Elevation1	Easting2	Northing2	Elevation2	From	To	Dip	Azimuth	Data
LineId0_1	12332.44	10992.34	-371.81	12336.19	10990.87	-371.82	0	4.027729	0.028652	111.4016	MeasureLine
LineId1_1	12336.19	10990.87	-371.82	12337.68	10992.43	-371.79	4.027729	6.184294	-0.66405	43.67887	MeasureLine
LineId2_1	12337.68	10992.43	-371.79	12338.78	10993.38	-371.78	6.184294	7.638377	-0.55195	49.19418	MeasureLine
LineId3_1	12338.78	10993.38	-371.78	12340.43	10994.38	-372.08	7.638377	9.590292	8.811756	58.77258	MeasureLine
LineId4_1	12340.43	10994.38	-372.08	12341.9	10995.71	-372.08	9.590292	11.57326	0.288341	47.87436	MeasureLine
LineId5_1	12341.9	10995.71	-372.08	12338.69	10999.68	-372.23	11.57326	16.68036	1.582303	321.0406	MeasureLine
LineId6_1	12338.69	10999.68	-372.23	12336.92	11000.12	-372.68	16.68036	18.56168	14.12098	283.9693	MeasureLine
LineId7_1	12336.92	11000.12	-372.68	12332.4	10997.44	-372.35	18.56168	23.82689	-3.7024	239.3358	MeasureLine
LineId8_1	12332.4	10997.44	-372.35	12332.95	10995.94	-372.06	23.82689	25.44935	-10.0449	159.8703	MeasureLine
LineId9_1	12332.95	10995.94	-372.06	12332.89	10994.31	-371.79	25.44935	27.10418	-9.53132	182.1262	MeasureLine
LineId_1	12332.89	10994.31	-371.79	12332.44	10992.34	-371.81	27.10418	29.12502	0.567058	192.8671	CumLength : 29.13
Length : 2.02											
PlanDistance: 2.02											
VertDistance: -0.02											
DipAngle : 0.57											
Direction : 192.87											
DRAW PLANE HR											
TotalArea : 55.09											
TotalPeri : 29.12											
TotalHR : 1.89											
PLAN VIEW HR											
PlanArea : 54.54											
PlanPeri : 28.98											
PlanHR : 1.88											
LineId11_1	12347.7	11083.09	-371.88	12349.55	11087.95	-371.89	0	5.200419	0.18728	20.83435	MeasureLine
LineId12_1	12349.55	11087.95	-371.89	12352.09	11087.48	-371.97	5.200419	7.784796	1.795928	100.4773	MeasureLine
LineId13_1	12352.09	11087.48	-371.97	12357.06	11085.87	-372.03	7.784796	13.00924	0.636244	107.954	MeasureLine
LineId14_1	12357.06	11085.87	-372.03	12360.41	11083.67	-371.95	13.00924	17.0184	-1.11483	123.2913	MeasureLine
LineId15_1	12360.41	11083.67	-371.95	12363.65	11080.9	-371.87	17.0184	21.28179	-1.12915	130.5216	MeasureLine
LineId16_1	12363.65	11080.9	-371.87	12359.6	11076.5	-371.86	21.28179	27.26278	-0.04765	222.6311	MeasureLine
LineId17_1	12359.6	11076.5	-371.86	12356.95	11076.5	-372.09	27.26278	29.92173	4.85364	270	MeasureLine
LineId18_1	12356.95	11076.5	-372.09	12351.63	11079.97	-372.07	29.92173	36.2735	-0.13516	303.111	MeasureLine

```
12332.44,10992.34,-371.81
12336.19,10990.87,-371.82
12337.68,10992.43,-371.79
12338.78,10993.38,-371.78
12340.43,10994.38,-372.08
12341.9,10995.71,-372.08
12338.69,10999.68,-372.23
12336.92,11000.12,-372.68
12332.4,10997.44,-372.35
12332.95,10995.94,-372.06
12332.89,10994.31,-371.79

12347.7,11083.09,-371.88
12349.55,11087.95,-371.89
12352.09,11087.48,-371.97
12357.06,11085.87,-372.03
12360.41,11083.67,-371.95
12363.65,11080.9,-371.87
12359.6,11076.5,-371.86
12356.95,11076.5,-372.09
12351.63,11079.97,-372.07
,,
12250.59,11105.15,-371.1
12254.2,11108.69,-371.43
12257.16,11107.68,-371.39
12261.43,11106.74,-371.43
12259.19,11100.81,-371.42
12257.67,11101.17,-371.41
12256.74,11100.97,-371.1
12255.26,11100.71,-371.1
12251.26,11102.09,-371.1
12250.88,11103.54,-371.1
12250.73,11104.64,-371.1
```

Figure 7: The csv input file can be imported as-exported from GEM4D (left-hand image) or processed to have the co-ordinates of the vertices for each polygon and the ,, converted to a white space carriage return (the second ,, is yet to be modified) (right-hand image). Please note that the polygon needs to be contiguously digitized in either a clockwise or counter-clockwise manner. A trial was undertaken using four strings representing the corners of an intersection, not digitized in sequential order and the maximum inscribed circle was noted to be very much smaller than that required.

3. Processing with Python Executable

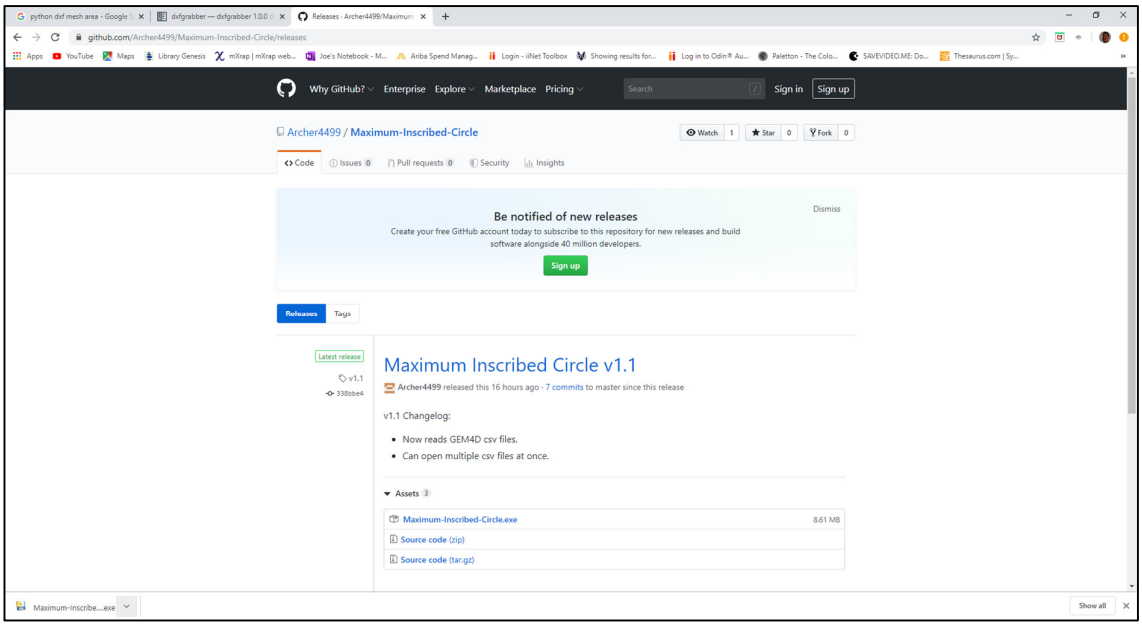


Figure 8: Download the Python Maximum Inscribed Circle executable (https://github.com/Archer4499/Maximum-Inscribed-Circle/releases) and place into a memorable location. Run it when ready, observing all relevant IT policy instructions and constraints emplaced by Microsoft.

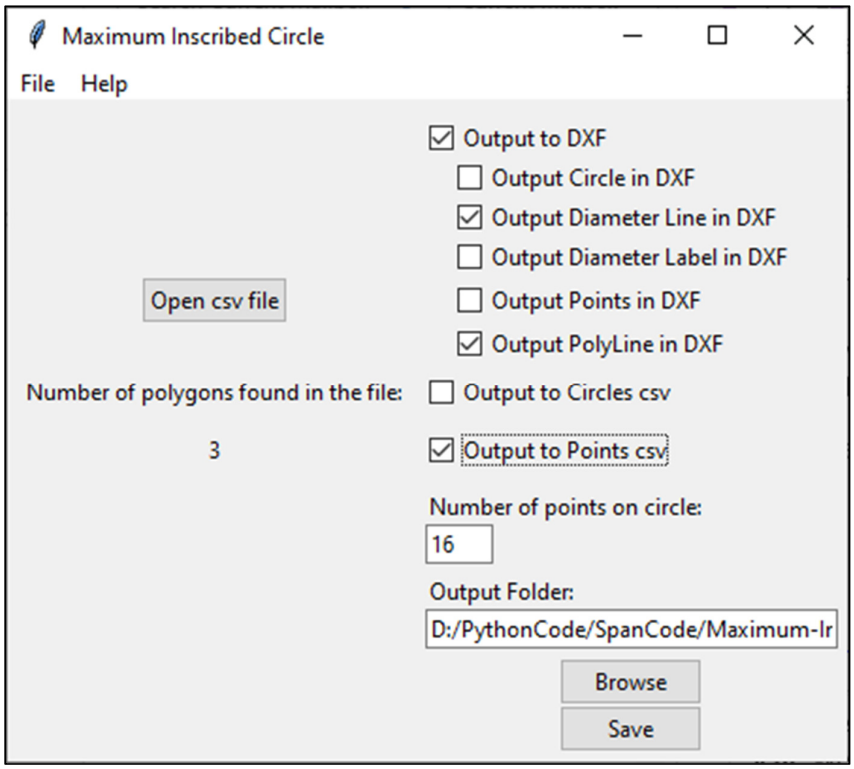


Figure 9: Use the Open csv file button to browse to the GEM4D exported or processed input csv file. Set the checkboxes as above for GEM4D usage. Points in DXF is optional for GEM4D. Output Circle in DXF and Output Diameter Label in DXF is not supported by GEM4D but may be supported in other packages (eg AutoCAD, Vulcan etc.). This is untested. Use the Browse button to browse to the folder where you want the files saved.

4. Importing the output files (GEM4D)

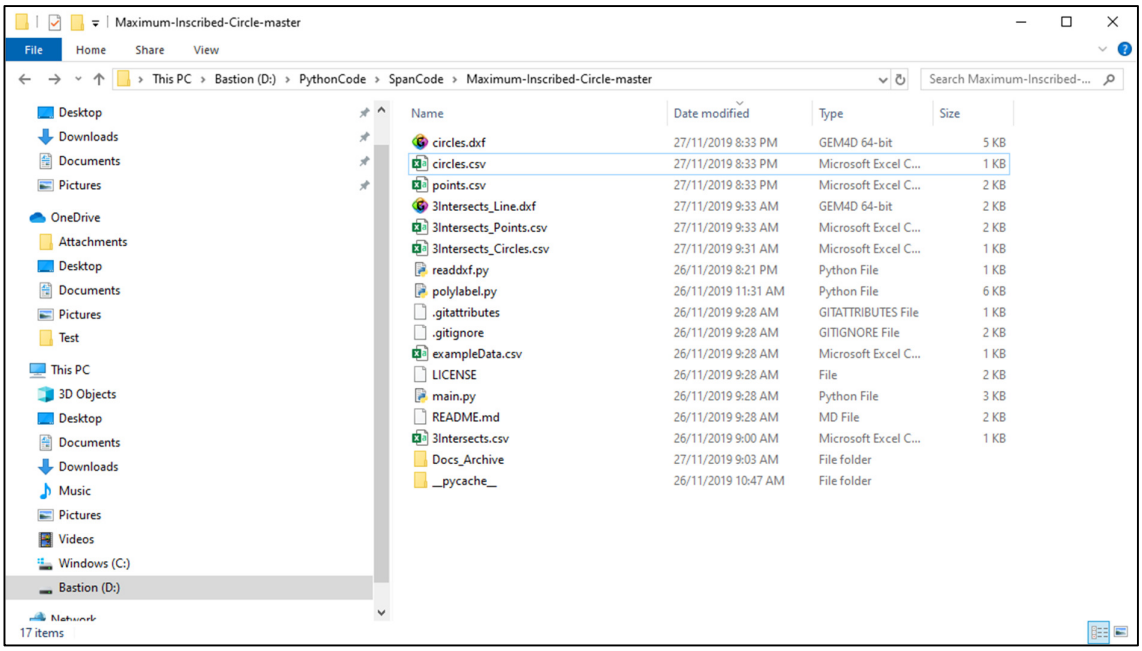


Figure 10: Locate the files in the working folder, these are now ready for import to GEM4D or other packages.

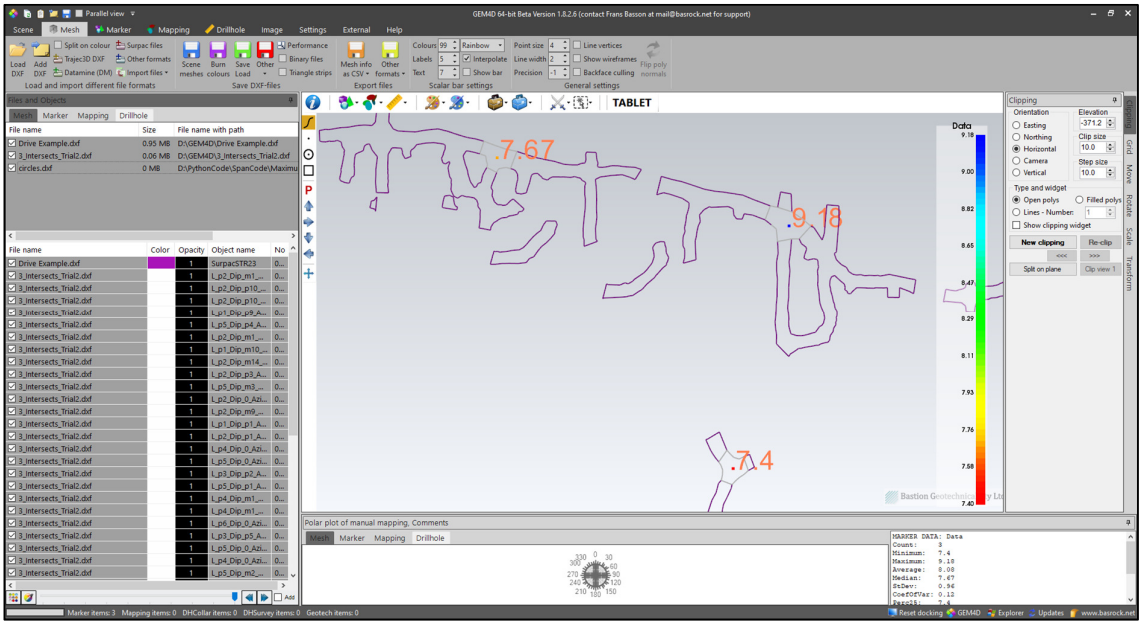


Figure 11: circles.csv file dropped into GEM4D as markers, DataColumn set to 1 and labelled Text scale 7.

Technical Note

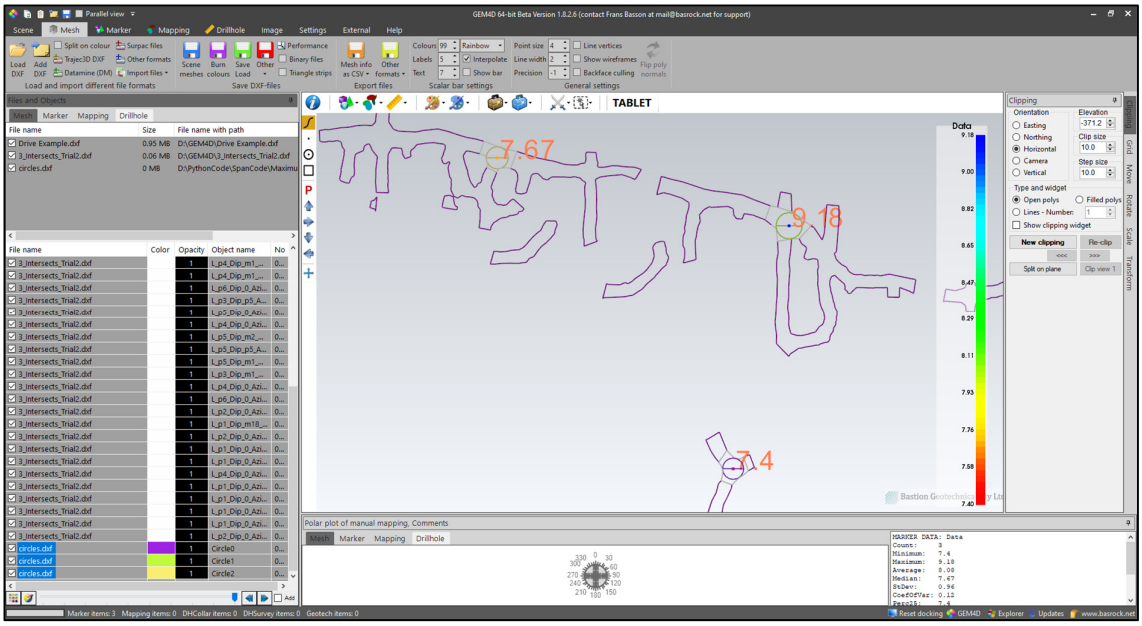


Figure 12: circles.dxf file dropped into GEM4D as meshes. Note the markers from the previous figure are retained. Also note that the labels do not import into GEM4D.

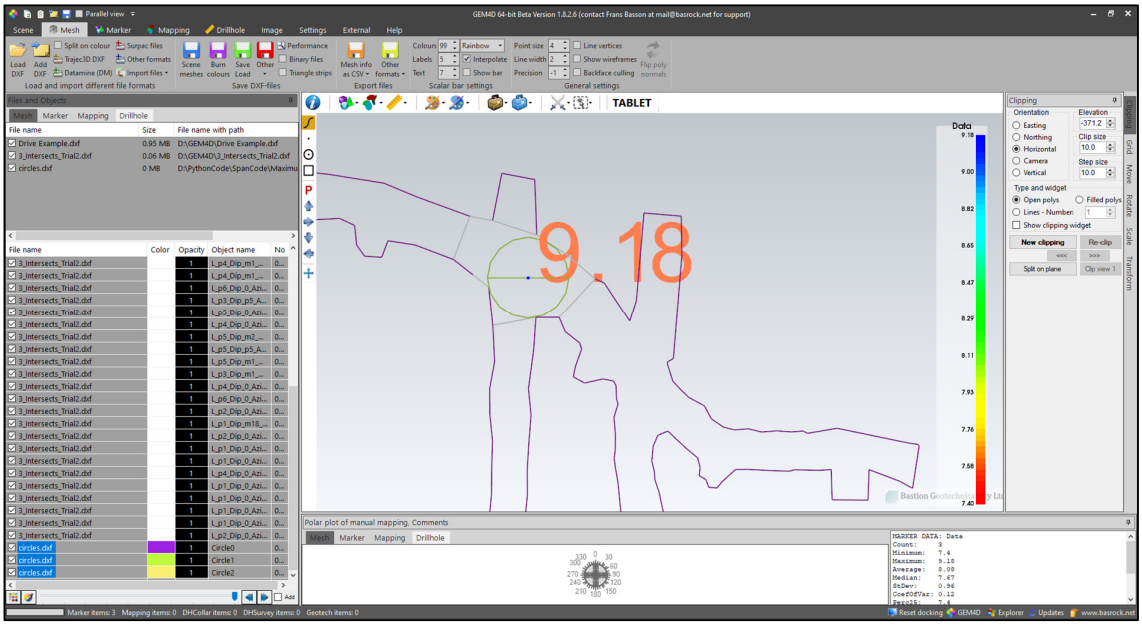


Figure 13: Final GEM4D view. Label derived from the circles.csv marker file.

Recommendations for Improvement

- Import dxf of closed polygons into Python. Removes the need to undertake actions shown in Figures 7, 8, and 9.
- ~~GUI interface and development of executable. Removes the need for the user to install Python or become familiar with how to run Python script or problems parsing arguments. Figures 10, 11, 12, 13 and 14.~~
- ~~Option to export one, some or all three output files. Removes the need to run the script up to three times if all three output files are required. Figures 12, 13 and 14.~~
- ~~Output points.csv as a POLYLINE dxf file.~~