RFLTools

Lesson 2: Sightlines

This lesson assumes you are familiar with creating and working with RFL alignment control, if this is not the case I strongly suggest you review Lesson 1.

For Lesson 2 we will be expanding on the linework created in Lesson 1. Most linework created using the sightline tools will be drawn on the current layer so for this lesson I will create a new layer, “Sightline”, and set its color to cyan. Also, I’ve set ‘PDMODE’ to be ‘34’ and ‘PDSIZE’ to be ‘0.25’ so that points will be visible on the screen.

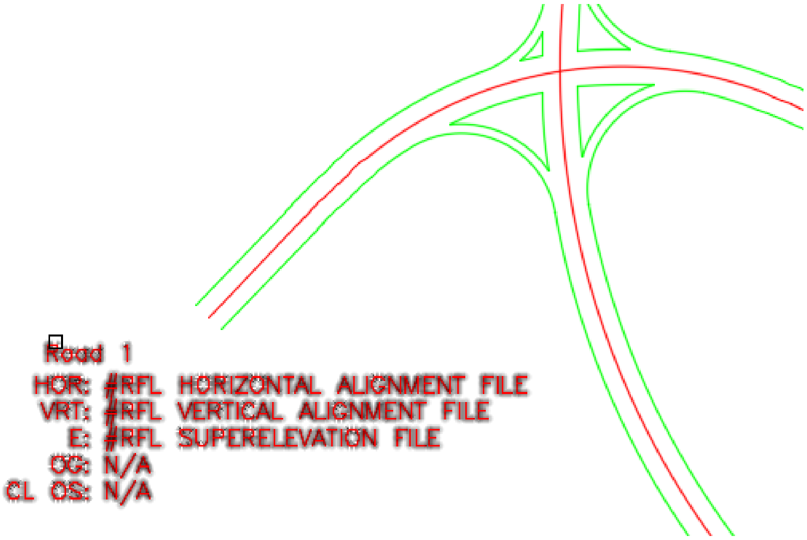
1. Simple Plan in 2D

1.1 Method 1: Fixed Offsets

Commands Used:

- RAB
- SIGHTLINE

To begin set the current alignment using RAB and select ‘Road 1’


Road 1
HOR: #RFL HORIZONTAL ALIGNMENT FILE
VRT: #RFL VERTICAL ALIGNMENT FILE
E: #RFL SUPERELEVATION FILE
OG: N/A
CL OS: N/A

Enter ‘SIGHTLINE’ at the command prompt. You will first be asked which direction you want the analysis, forward or reverse (up chainage or down) – select Forward.

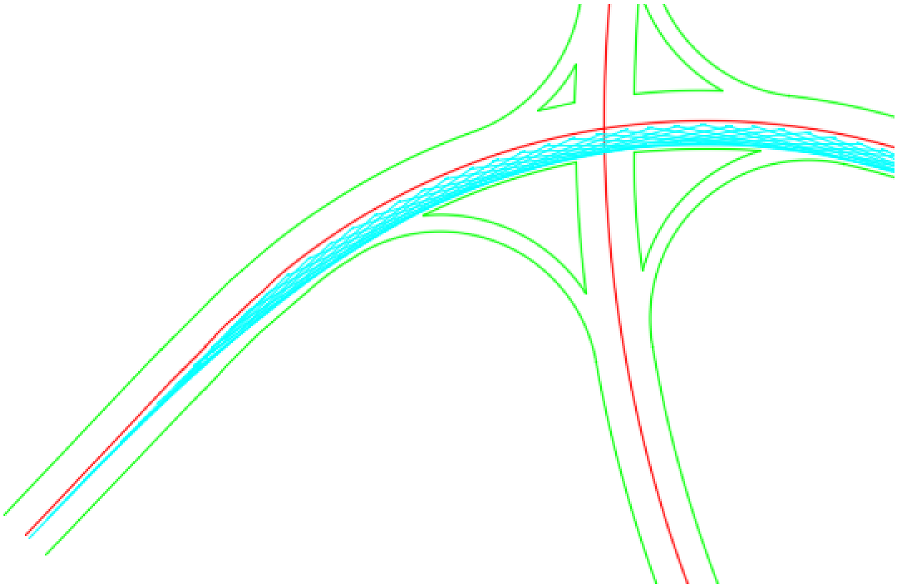
Next you will be asked over what chainage range you wish to complete the analysis for. Here I will use the entire alignment so press return.

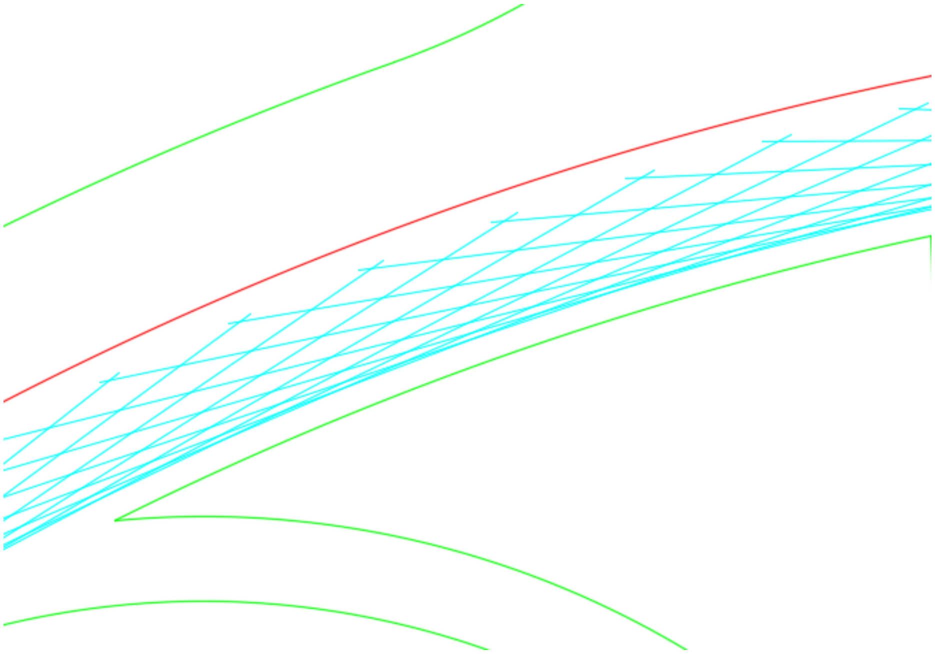
This method assumes a fixed lane offset from control. For this method I'll assume we're looking at the inside lane and assume the lane width is 3.7m, so enter 1.85 (half of 3.7).

Next you will be prompted for the step size. This value specifies the station increment between sightline start points - I'll use 10.

Finally, enter the sight distance. This will be the distance (point to point) of the sightlines. I'll use 120 for this example.

SIGHTLINE will draw a series of lines over the specified chainage range with each line drawn precisely at the specified distance forming essentially a hatch showing the zone of influence





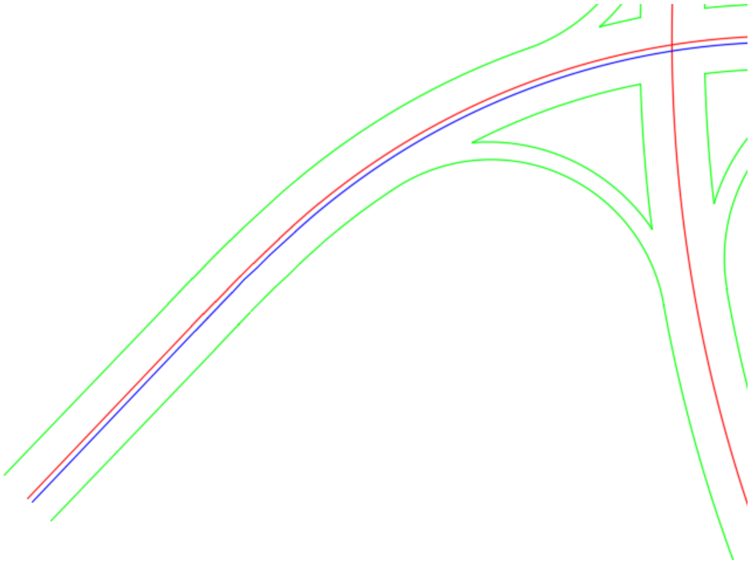
1.2 Method 2: Polyline Selection

Commands Used:

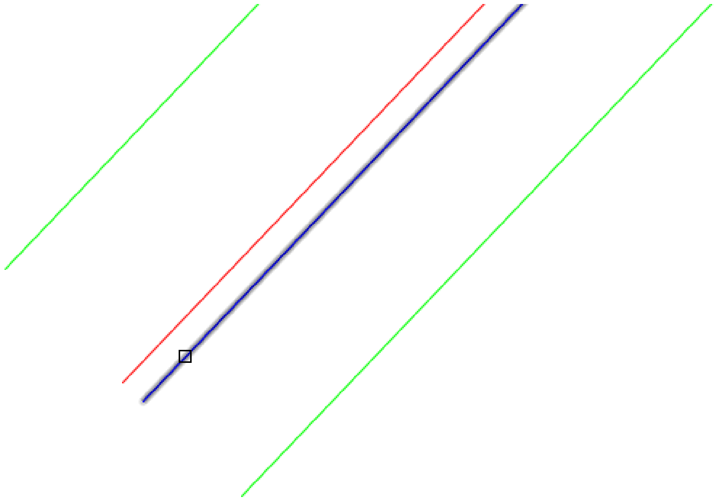
- SIGHTLINE3D

This method uses the more advanced tool, ‘SIGHTLINE3D’, but the tool has a simple 2D mode.

To begin you must first have defined a travel path. I’ve duplicated Method 1 by offsetting the control alignment by 1.85 and joined all together to a single polyline

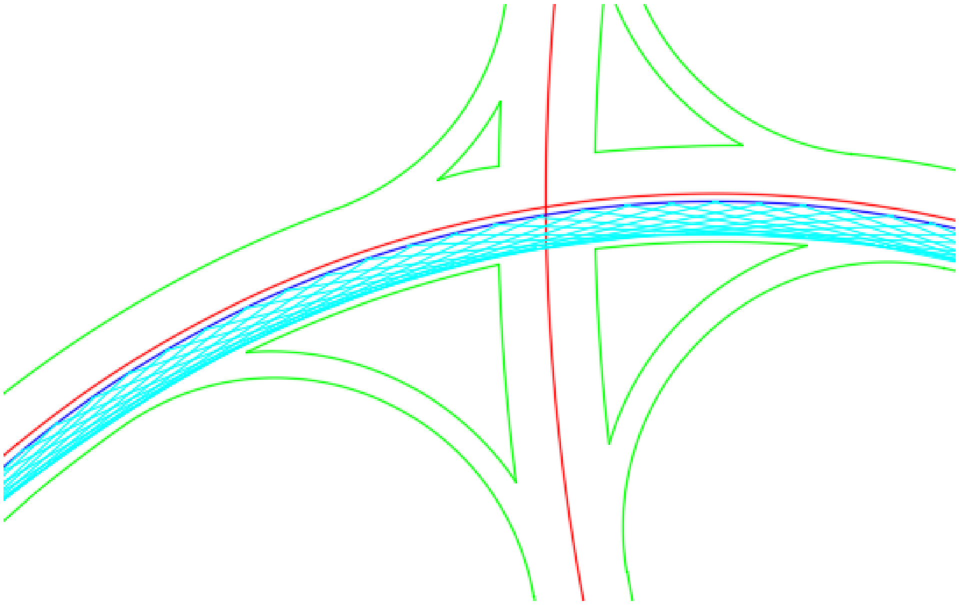


Enter ‘SIGHTLINE3D’ at the command prompt. You will be asked to select the polyline near the start (this will establish which direction the sightline analysis is done)



Next is the sight length, I’ll use 120 again. Use the default for iterative analysis (this will be explained later). And enter 10 for the step size.

Press ‘<Return>’ to complete in 2D and ‘<Return>’ for no barrier analysis (explained later). You should see the same analysis as was completed in Method 1



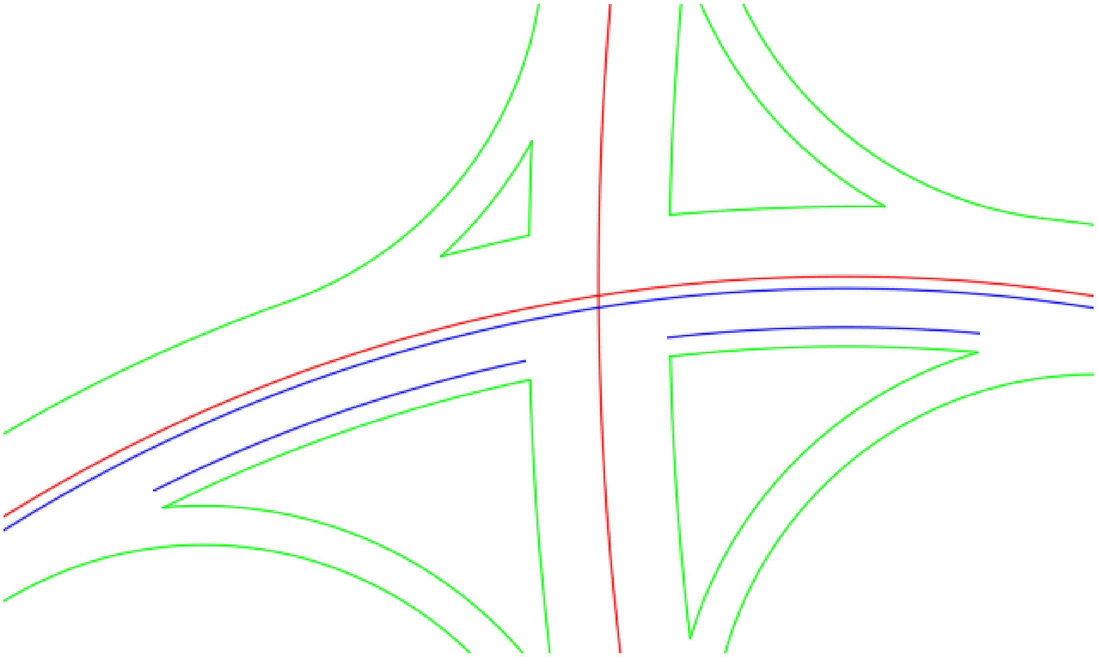
2. Advanced Plan in 2D

Commands Used:

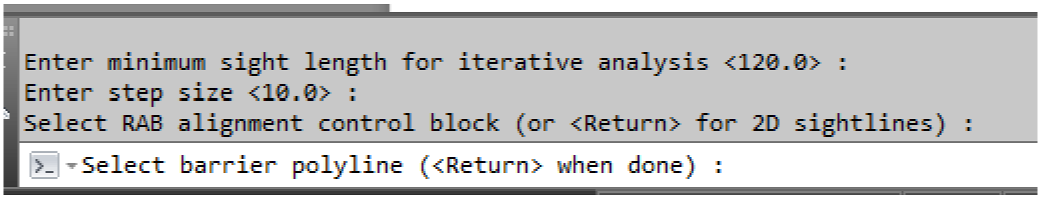
- SIGHTLINE3D

In this example we would like to know the impacts the sightlines have on barriers (or other 2D polylines). Note that the barrier objects must be polylines.

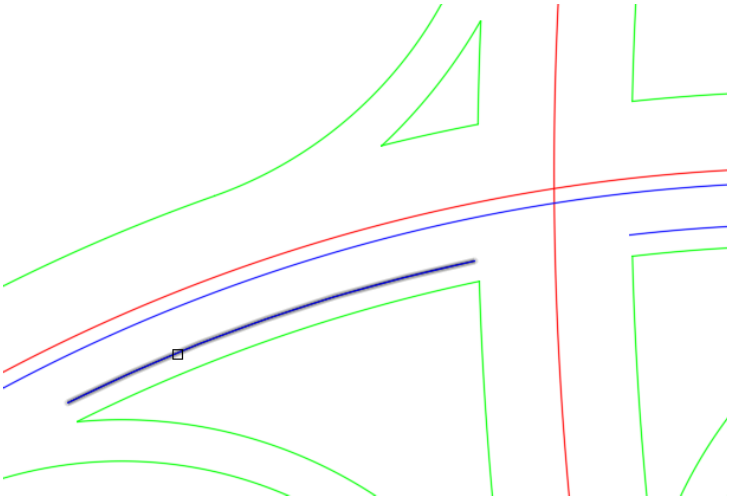
In this example I've created a couple of polylines that I know will be crossing the sightlines



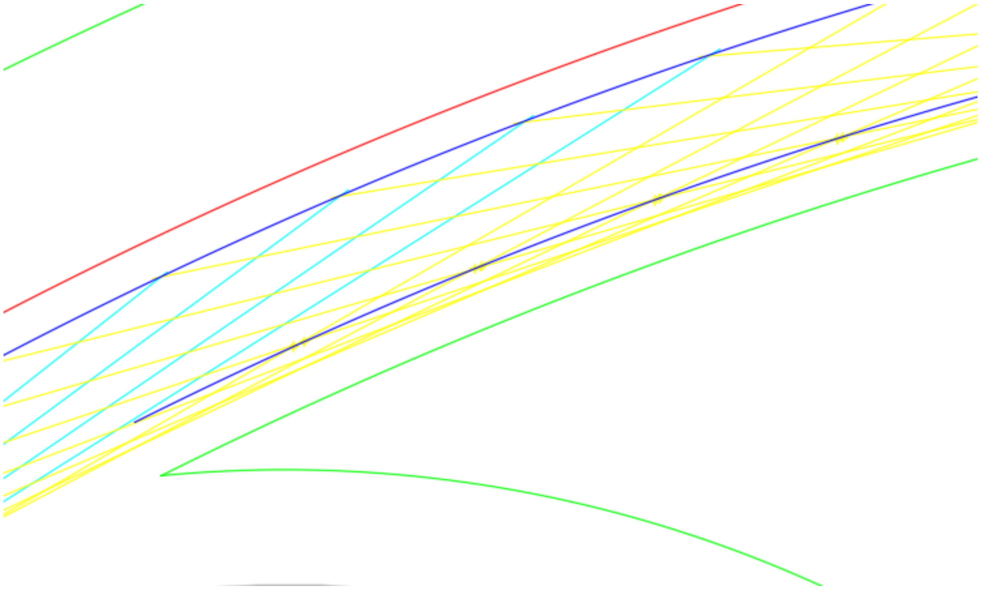
We'll now complete the analysis as in Method 2 above but when we are prompted to select a barrier we'll select these new polylines



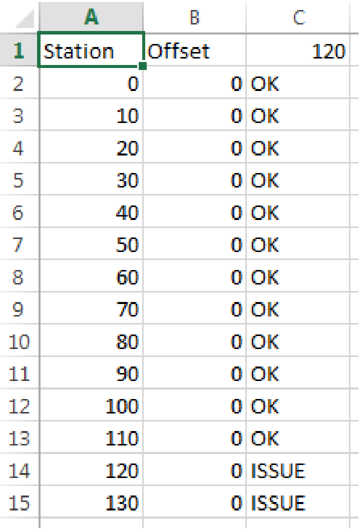
```
Enter minimum sight length for iterative analysis <120.0> :
Enter step size <10.0> :
Select RAB alignment control block (or <Return> for 2D sightlines) :
Select barrier polyline (<Return> when done) :
```



You'll notice that when there is an intersection between the sightline and the barrier the color changes to yellow (actually truecolor '255,255,0') and a point is placed at the intersection. This gives a visual method of determining which sightlines have issues and the points where the issues occur



In addition, SIGHTLINE3D will create a .CSV file where you can see which chainages are impacted. The file will be located in your 'Documents' folder as 'SIGHTLINE3D.CSV' (note that this file is overwritten each time you run SIGHTLINE3D – so move/rename as necessary)



	A	B	C
1	Station	Offset	120
2	0	0	OK
3	10	0	OK
4	20	0	OK
5	30	0	OK
6	40	0	OK
7	50	0	OK
8	60	0	OK
9	70	0	OK
10	80	0	OK
11	90	0	OK
12	100	0	OK
13	110	0	OK
14	120	0	ISSUE
15	130	0	ISSUE

If no alignment is currently in memory the stations and offsets will be relative to the selected path polyline, if there is an alignment defined they will be relative to that alignment.

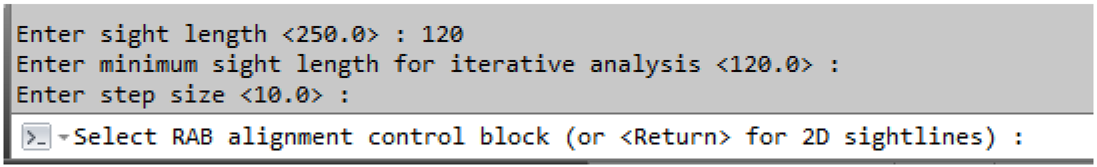
3. Simple Plan in 3D

Commands Used:

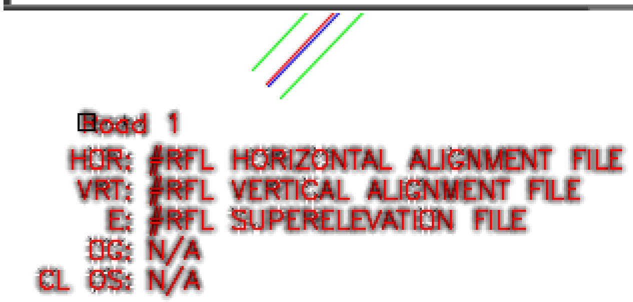
- SIGHTLINE3D
- SELEV

As in the previous example but we would like to know if the sightlines and 'see over' the barrier. The assumption here is that all the object elevation are controlled by the alignment/profile/superelevation within the RAB block, i.e. they are placed on the roadway and top of barrier is at a fixed height above the road. If this is not the case you must analyze based on a surface model with will be discussed later.

Proceed as described in the previous example but select the RAB alignment block



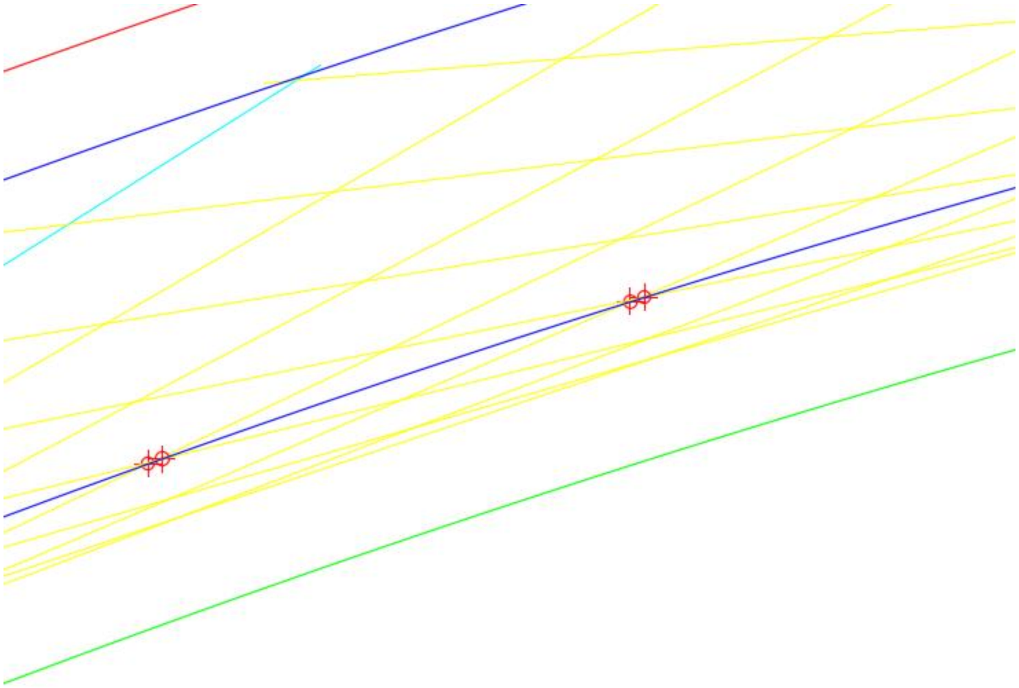
```
Enter sight length <250.0> : 120
Enter minimum sight length for iterative analysis <120.0> :
Enter step size <10.0> :
>_ ▾ Select RAB alignment control block (or <Return> for 2D sightlines) :
```


Road 1
HOR: #RFL HORIZONTAL ALIGNMENT FILE
VRT: #RFL VERTICAL ALIGNMENT FILE
E: #RFL SUPERELEVATION FILE
OG: N/A
CL OS: N/A

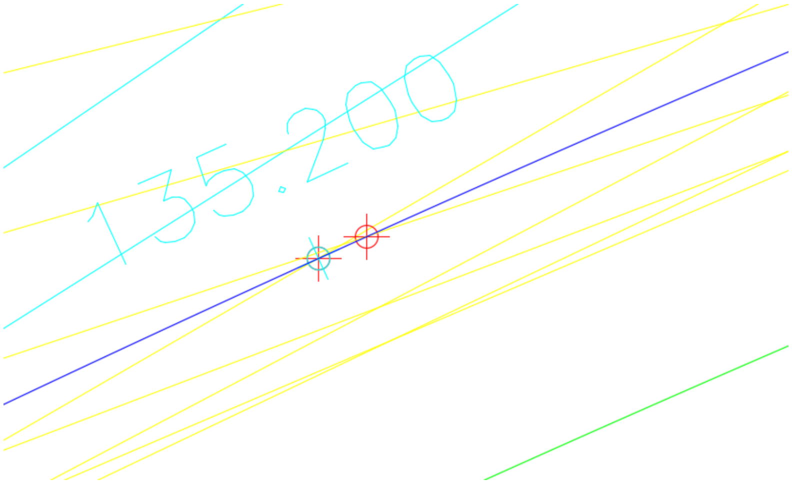
This time you will be prompted to enter an eye height and target height. Here I will use 1.05 and 0.15 respectively.

Select 'No' to compare to surface and then select the barrier polylines. This time you will be prompted to enter the height of the barrier, I will use 0.69.

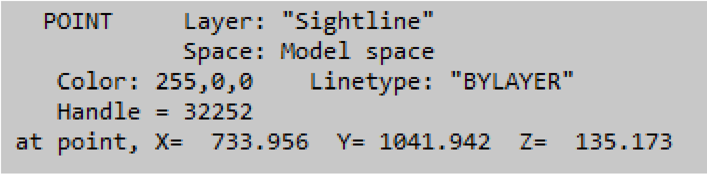
The analysis will proceed as before except the points drawn will be red (color '255,0,0') when there is a conflict (eye line below top of barrier) or green (color '0,255,0') when no conflict (eye line above barrier)



You can check by using 'SELEV' to insert point blocks at the conflict points. 'SELEV' will place a 'SPOTELEVATION' block identifying the elevation of the road. In my example I see 135.200



The top of the barrier will be 0.69 above this point which calculates to 135.890. You can list the red point and it will have been placed at the eye line elevation, in this example it shows 135.173



```
   POINT      Layer: "Sightline"
              Space: Model space
    Color: 255,0,0    Linetype: "BYLAYER"
    Handle = 32252
 at point, X=  733.956  Y= 1041.942  Z=  135.173
```

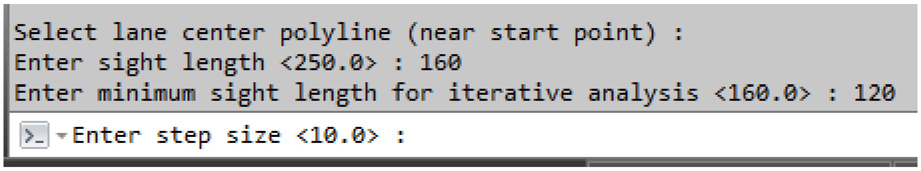
135.173 is below the top of barrier at 135.890 and hence it was drawn red.

4. Iterative Plan in 3D

Commands Used:

- SIGHTLINE3D

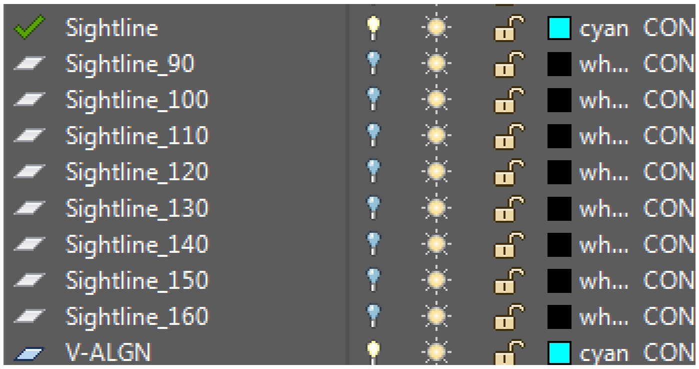
In this example I would like to calculate sight lines along a path over a range of sight distances. In this example I will analyze barrier conflicts but over a range from 160 down to 90. Complete as in “3. Simple Plan in 3D” but instead enter 160 for the sight length and 90 for the minimum sight length



```
Select lane center polyline (near start point) :
Enter sight length <250.0> : 160
Enter minimum sight length for iterative analysis <160.0> : 120
Enter step size <10.0> :
```

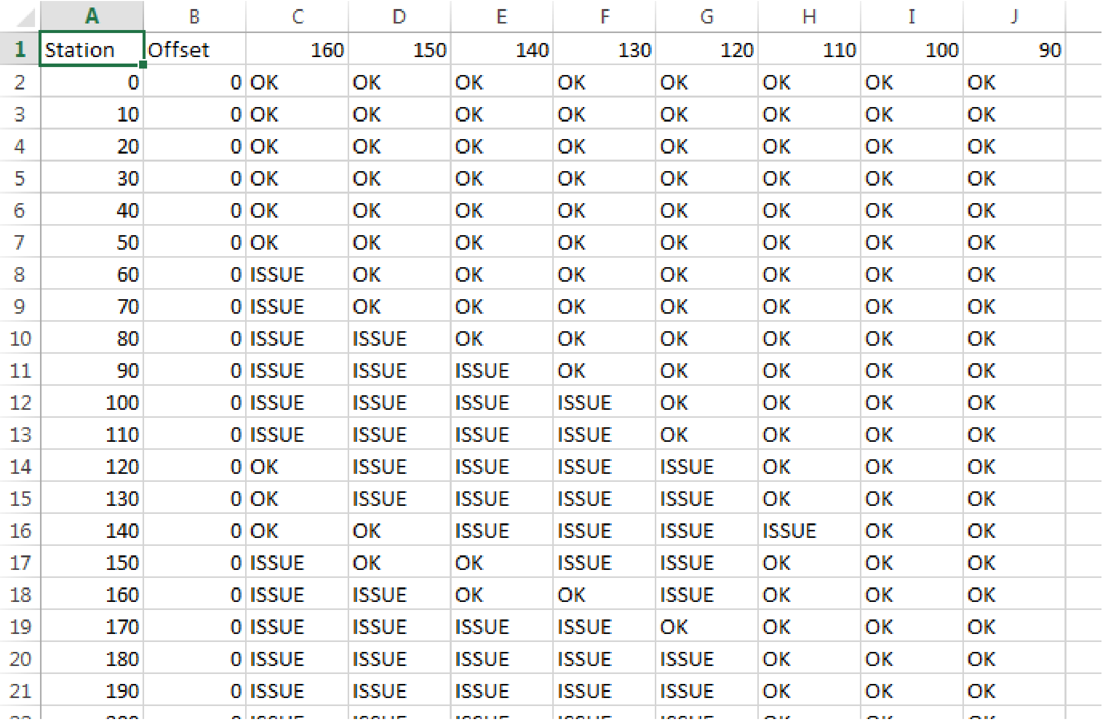
Again use 10 for the step size. Complete as before without the RAB alignment block and select to two barrier polylines.

This time you will not see anything drawn. However, there will be a series of layers created each turned off. Each new layer is prefixed with the current layer (‘Sightline’ in my case) and suffixed with the sight distance



✓	Sightline				cyan	CON
	Sightline_90				wh...	CON
	Sightline_100				wh...	CON
	Sightline_110				wh...	CON
	Sightline_120				wh...	CON
	Sightline_130				wh...	CON
	Sightline_140				wh...	CON
	Sightline_150				wh...	CON
	Sightline_160				wh...	CON
	V-ALGN				cyan	CON

You can turn on which ever you wish to review the results. Additionally, ‘SIGHTLINE3D.CSV’ now contains the summary for the entire analysis



	A	B	C	D	E	F	G	H	I	J
1	Station	Offset	160	150	140	130	120	110	100	90
2	0	0	OK	OK	OK	OK	OK	OK	OK	OK
3	10	0	OK	OK	OK	OK	OK	OK	OK	OK
4	20	0	OK	OK	OK	OK	OK	OK	OK	OK
5	30	0	OK	OK	OK	OK	OK	OK	OK	OK
6	40	0	OK	OK	OK	OK	OK	OK	OK	OK
7	50	0	OK	OK	OK	OK	OK	OK	OK	OK
8	60	0	ISSUE	OK	OK	OK	OK	OK	OK	OK
9	70	0	ISSUE	OK	OK	OK	OK	OK	OK	OK
10	80	0	ISSUE	ISSUE	OK	OK	OK	OK	OK	OK
11	90	0	ISSUE	ISSUE	ISSUE	OK	OK	OK	OK	OK
12	100	0	ISSUE	ISSUE	ISSUE	ISSUE	OK	OK	OK	OK
13	110	0	ISSUE	ISSUE	ISSUE	ISSUE	OK	OK	OK	OK
14	120	0	OK	ISSUE	ISSUE	ISSUE	ISSUE	OK	OK	OK
15	130	0	OK	ISSUE	ISSUE	ISSUE	ISSUE	OK	OK	OK
16	140	0	OK	OK	ISSUE	ISSUE	ISSUE	ISSUE	OK	OK
17	150	0	ISSUE	OK	OK	ISSUE	ISSUE	OK	OK	OK
18	160	0	ISSUE	ISSUE	OK	OK	ISSUE	OK	OK	OK
19	170	0	ISSUE	ISSUE	ISSUE	ISSUE	OK	OK	OK	OK
20	180	0	ISSUE	ISSUE	ISSUE	ISSUE	ISSUE	OK	OK	OK
21	190	0	ISSUE	ISSUE	ISSUE	ISSUE	ISSUE	OK	OK	OK
22	200	0	ISSUE	ISSUE	ISSUE	ISSUE	ISSUE	OK	OK	OK

5. Advanced Plan in 3D

This will analyze sightlines against a Civil 3D surface. For this example I will check sightlines along an existing road.

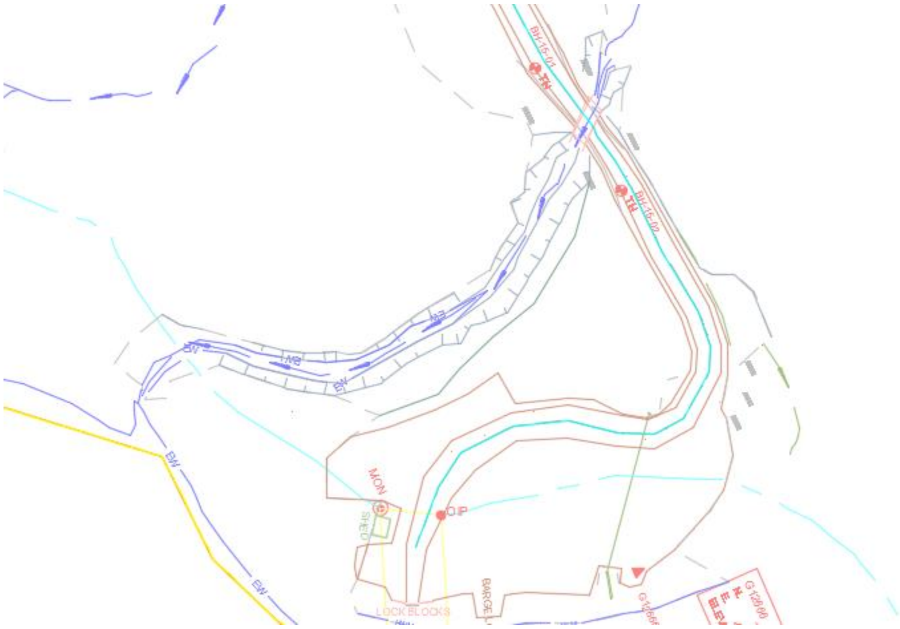
Commands Used:

- 3DP2ALIGN
- DRAPEPOLY
- SIGHTLINE3D

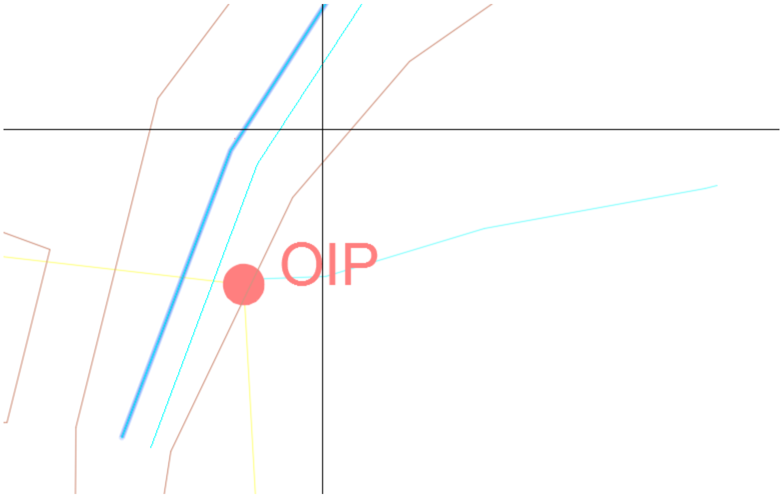
I'm going to use the base from my Tuck Inlet project. You can use base provided you have a surface which covers the entire road to be analyzed and you have a polyline along the roadway.

On the Tuck project we have an existing gravel road averaging about 5 m to 6 m width. The base has a polyline following the centerline of this road.

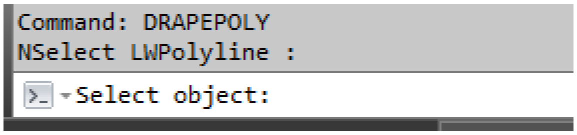
To begin I will create a "Sightline" layer and copy the centerline onto it. I will then ensure I have a continuous polyline over the length of this road



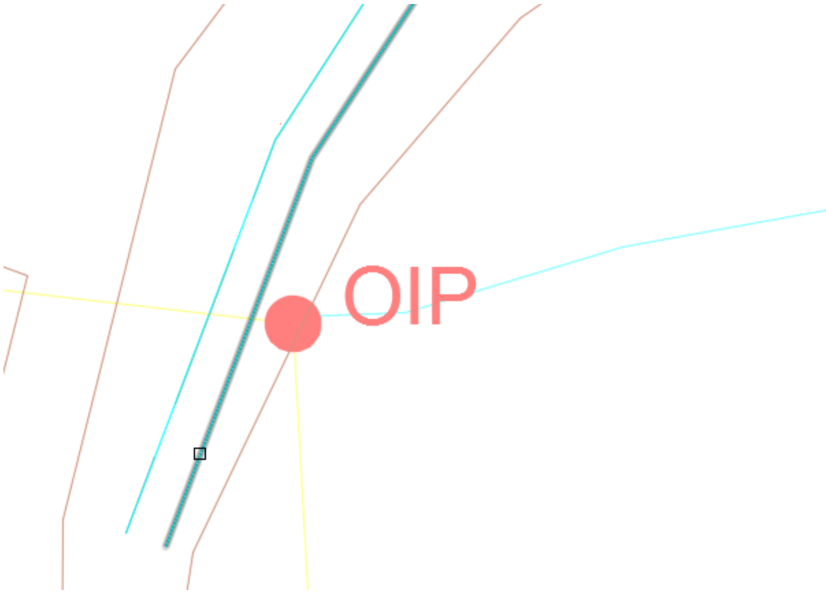
To create the sightline path I will offset this polyline 1.5 m



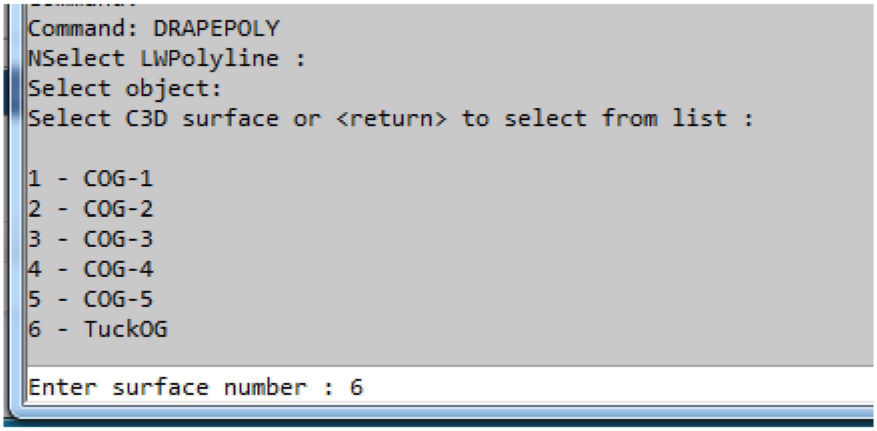
Once you have to sight path polyline use DRAPEPOLY to produce a 3D Polyline along the path



```
Command: DRAPEPOLY
NSelect LWPolyline :
Select object:
```



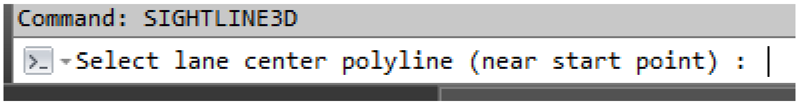
Once selected you will be prompted to select the Civil 3D surface



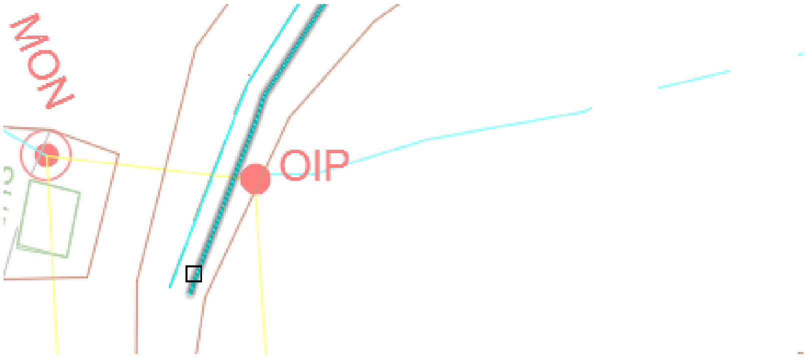
```
Command: DRAPEPOLY
NSelect LWPolyline :
Select object:
Select C3D surface or <return> to select from list :

1 - COG-1
2 - COG-2
3 - COG-3
4 - COG-4
5 - COG-5
6 - TuckOG

Enter surface number : 6
```

DRAPEPOLY will create a 3D Polyline along the path.

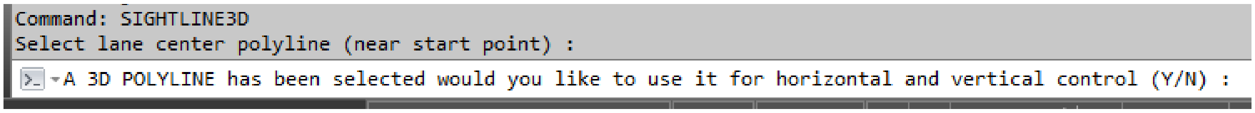
We can now proceed with the sightline analysis by entering ‘SIGHTLINE3D’ at the command prompt and selecting the newly created 3D polyline



```
Command: SIGHTLINE3D
Select lane center polyline (near start point) :
```



SIGHTLINE3D has determined that you have selected a 3D polyline and will now ask if you would like to use it for horizontal and vertical control



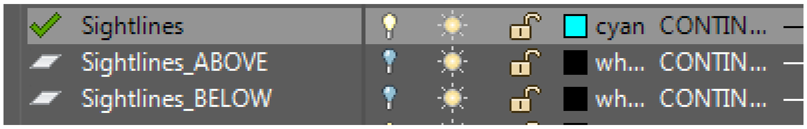
```
Command: SIGHTLINE3D
Select lane center polyline (near start point) :
A 3D POLYLINE has been selected would you like to use it for horizontal and vertical control (Y/N) :
```

Enter ‘Yes’ and a temporary alignment and profile will be generated from the vertices of this 3D polyline.

For my example I will use 90 for sight distance, non-iterative, 1.05 for eye, 0.15 for target and 10 for step. When you are prompted to compare to surface select ‘Yes’ and choose the surface you used for generating the 3D polyline. In this example I will not be checking against a barrier.

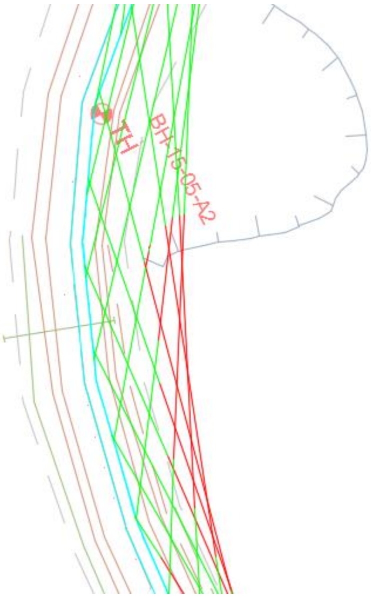
Be prepared that this tool is fairly slow running – especially for longer alignments and complex surfaces. Based on the 3D polyline, SIGHTLINE3D will determine a 3D sightline and for each 3D sightline a profile will be generated.

When SIGHTLINE3D has finished there will be 2 new layers '*_ABOVE' and '*_BELOW' which will be turned off. Turning them on will reveal the conflicts between the sightlines and the surface



✓	Sightlines	🟦 cyan	CONTIN...
▱	Sightlines_ABOVE	■ wh...	CONTIN...
▱	Sightlines_BELOW	■ wh...	CONTIN...

When a sightline falls below the surface it will be colored red (255,0,0) and when above it will be colored green (0,255,0)



Note that this command is largely a 'Work-in-Progress' and I am working towards making it run faster. If you require this tool please let me know as I have recreated a number of my base programming in Visual Studio and these can be loaded and used in place of the .LSP versions - .DLL coding runs significantly faster than .LSP

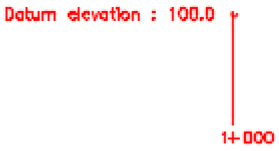
6. Simple Profile

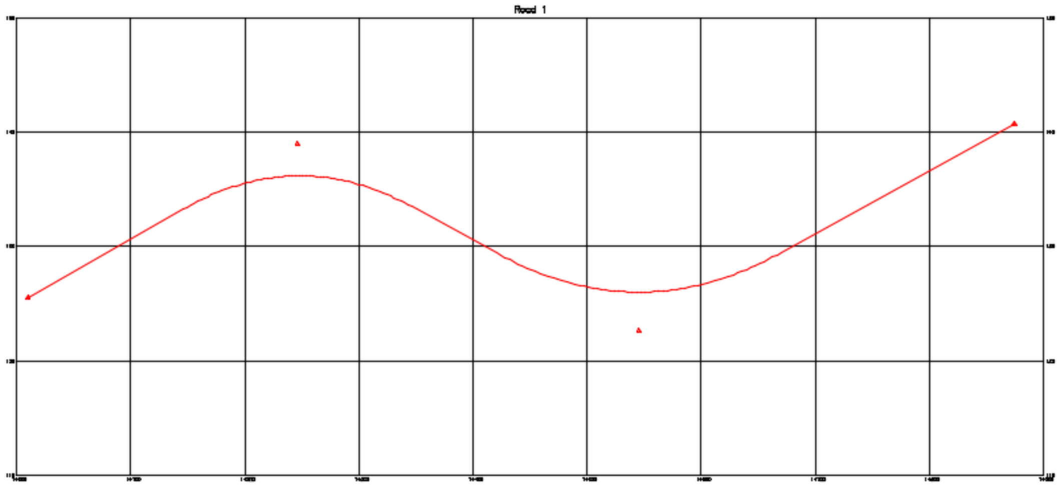
Commands Used:

- RAB
- SIGHTLINEPROF

In this example we will draw sightlines on a profile. If you do not have a profile defined and drawn please refer back to 'Lesson 1'.

To complete this you require the profile definition block and (preferred) a drawn profile



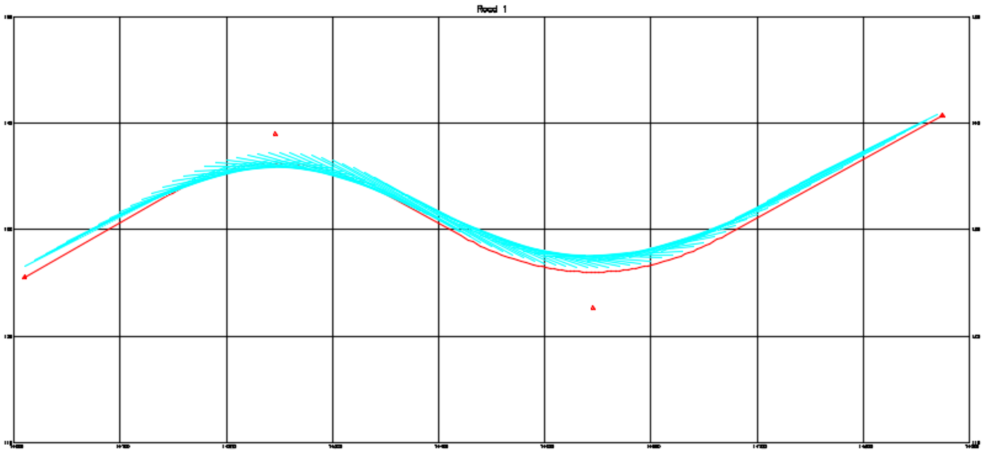


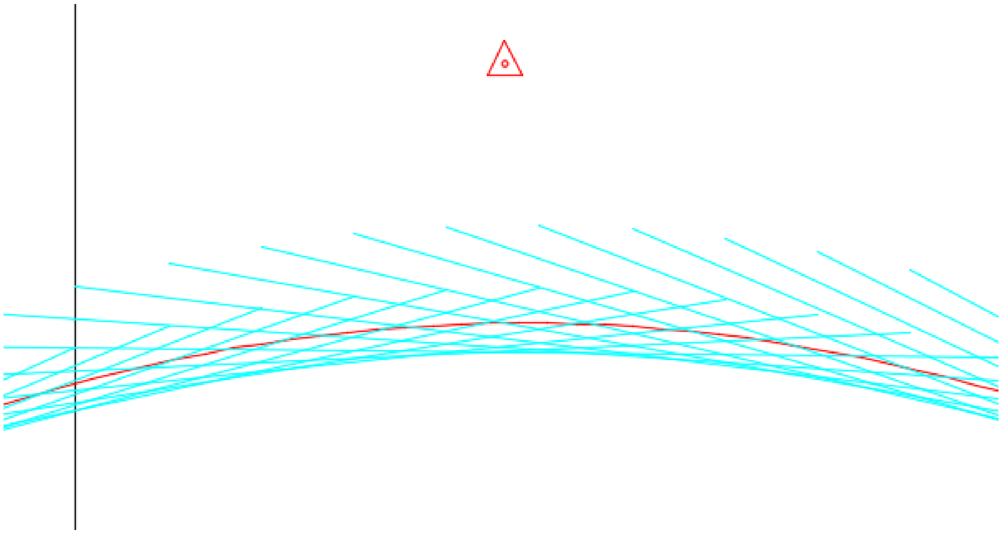
Ensure you have the correct profile loaded into memory – enter 'RAB' and select the Road 1 block. Enter 'SIGHTLINEPROF' at the command prompt.

First select the profile definition block to set up the profile location. You can proceed forward (up chainage) or reverse (down chainage) – I'll proceed forward. Now enter the start/end chainages or <return> for profile start and end – I'll use the profile start and end.

Next enter the eye and target height, I'll use 1.05 and 0.38. Next step size, I'll use 10. And finally the sight distance, I'll use 120.

SIGHTLINEPROF will draw a series of lines from an eye height above the design profile to the target above the design profile at a length ('X' distance)





The analysis will be able to be visually inspected for conflict locations.

7. Maximum Profile Sight Distance

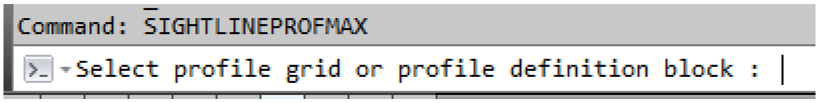
Commands Used:

- RAB
- SIGHTLINEPROFMAX

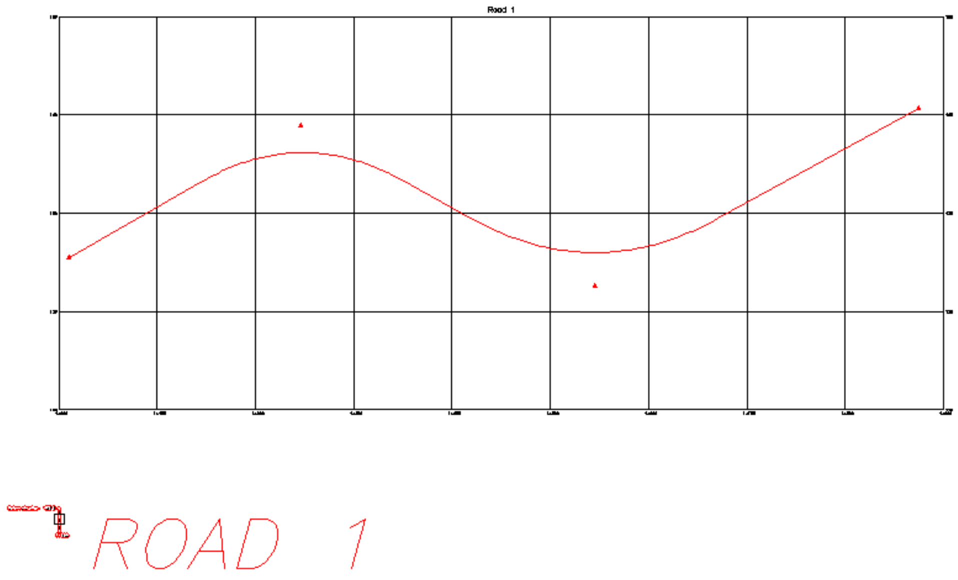
This tool will calculate the maximum sight distance along a profile iteratively between to chainages.

First ensure you have a profile loaded into memory using RAB (if unfamiliar with RAB please refer to Lesson 1).

Enter ‘SIGHTLINEPROFMAX’ at the command prompt. You will be prompted to select the profile grid definition block

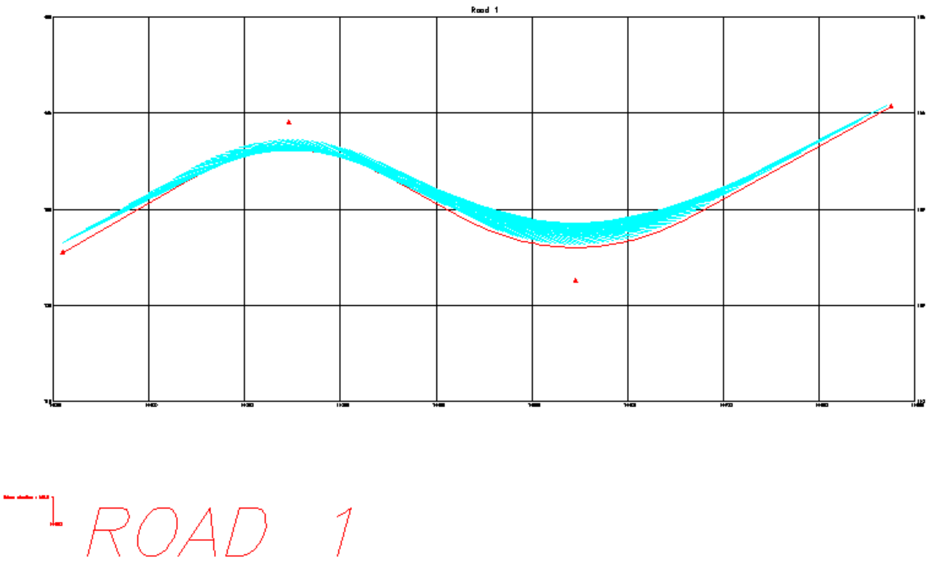


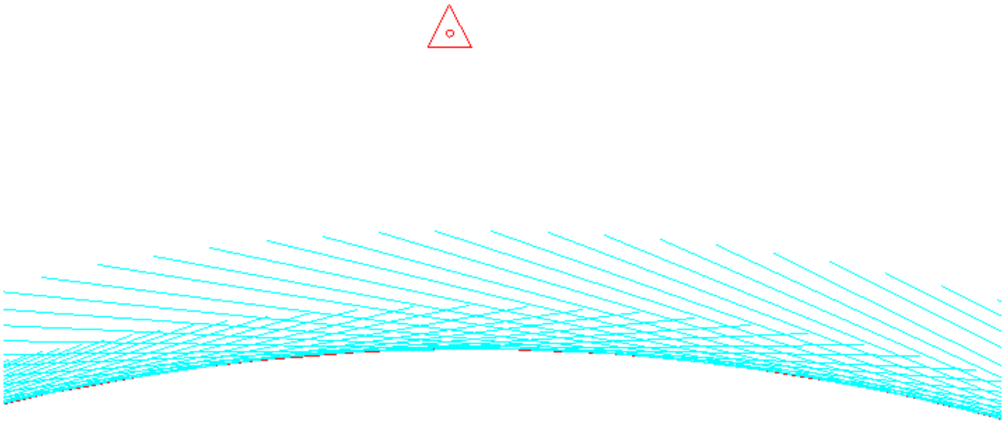
```
Command: SIGHTLINEPROFMAX
Select profile grid or profile definition block :
```



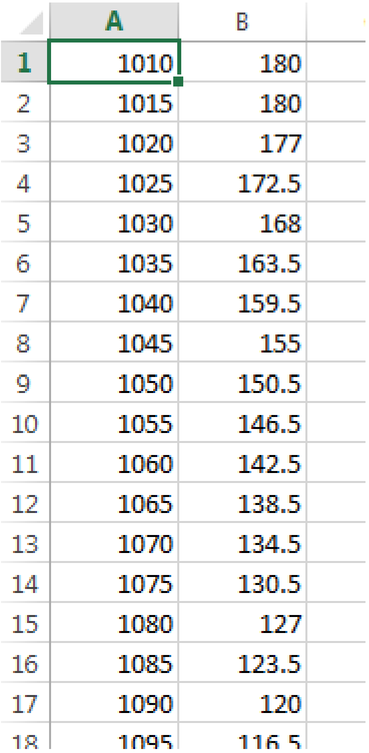
Enter the desired direction, Forward (up chainage) or Reverse (down chainage). Start and end chainage (here I will use the start and end of the profile). Eye height and target (I'll use 1.05 and 0.38). And the step size (I'll use 5). You will next be prompted for the maximum sight distance – this gives the program a stopping point (it won't look any longer than this distance, I'll use 180).

SIGHTDISTPROFMAX will now traverse the station range and look for maximum sight distance. The calculated distances will be rounded to 'STEP/10' units. You will see the sightlines drawn on the profile and they will just 'skim' over the design profile





Additional you will find a summary .CSV file located in your ‘Documents’ folder

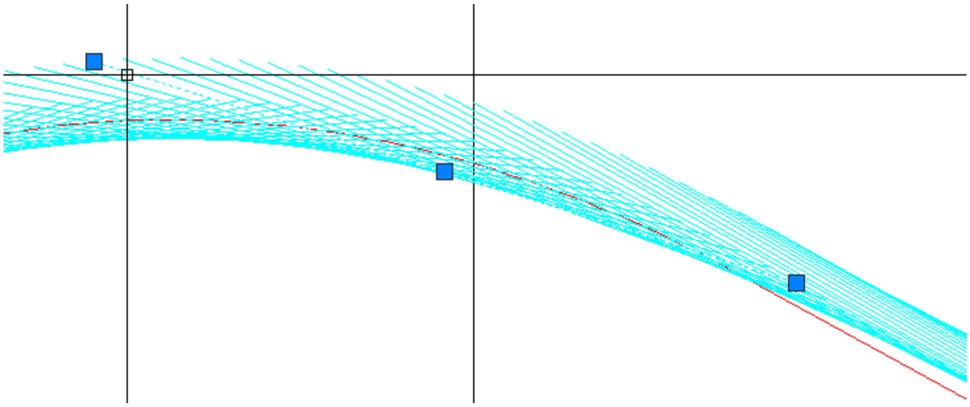


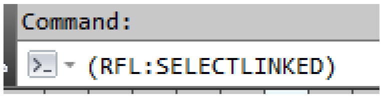
	A	B
1	1010	180
2	1015	180
3	1020	177
4	1025	172.5
5	1030	168
6	1035	163.5
7	1040	159.5
8	1045	155
9	1050	150.5
10	1055	146.5
11	1060	142.5
12	1065	138.5
13	1070	134.5
14	1075	130.5
15	1080	127
16	1085	123.5
17	1090	120
18	1095	116.5

8. Bonus Tool – Select Linked

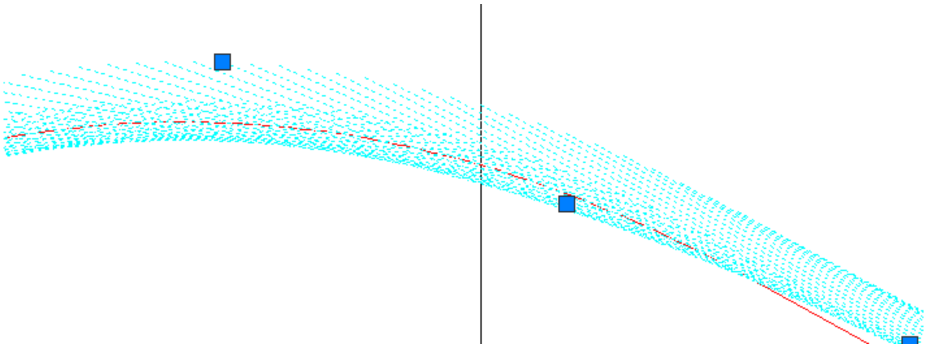
The vast majority of my tools that create multiple entities now ‘link’ them together. Basically any entity created knows the one that was created previously to it and the one that was created after it.

If you highlight an entity and then enter '(RFL:SELECTLINKED)' the tool will highlight all the entities created in the command that created it



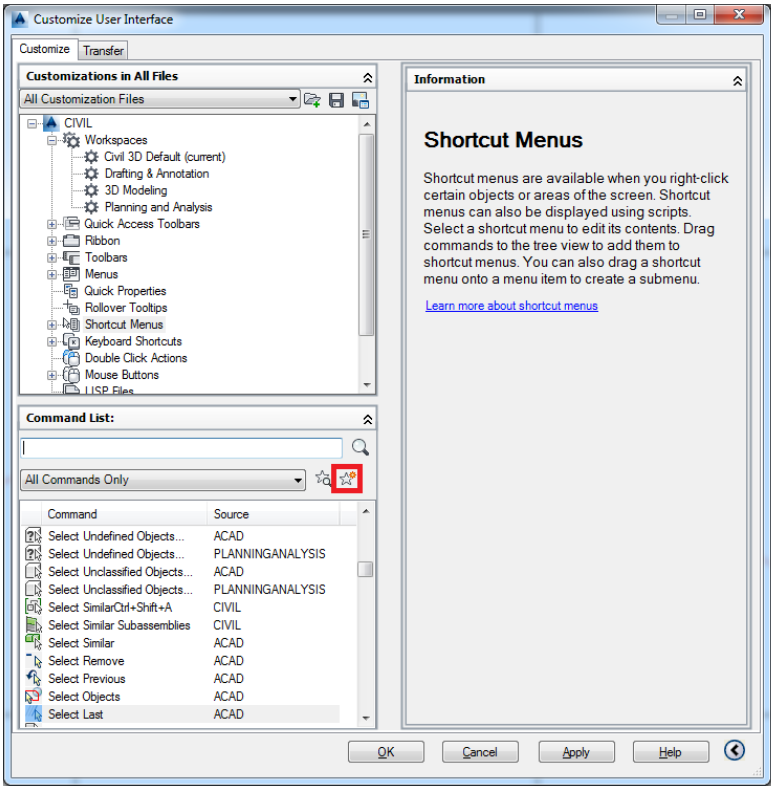


```
Command:
(RFL:SELECTLINKED)
```



You can now delete, move, etc.

To make this tool even easier to use you can add it to your 'Right-Click' menu. Start by entering 'CUI' at the command prompt and select 'Create a new command'



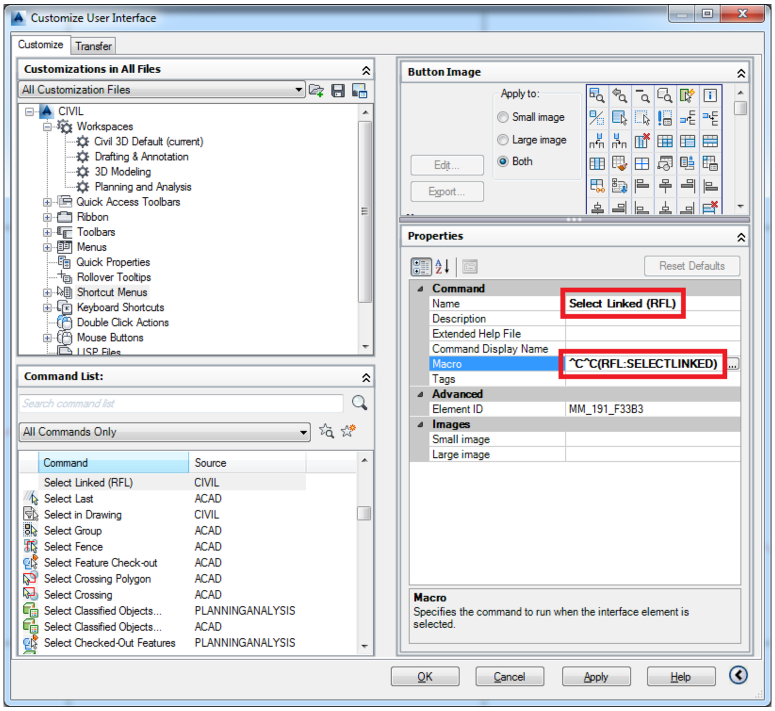
Shortcut Menus

Shortcut menus are available when you right-click certain objects or areas of the screen. Shortcut menus can also be displayed using scripts. Select a shortcut menu to edit its contents. Drag commands to the tree view to add them to shortcut menus. You can also drag a shortcut menu onto a menu item to create a submenu.

Learn more about shortcut menus

Command	Source
Select Undefined Objects...	ACAD
Select Undefined Objects...	PLANNINGANALYSIS
Select Unclassified Objects...	ACAD
Select Unclassified Objects...	PLANNINGANALYSIS
Select SimilarCtrl+Shift+A	CIVIL
Select Similar Subassemblies	CIVIL
Select Similar	ACAD
Select Remove	ACAD
Select Previous	ACAD
Select Objects	ACAD
Select Last	ACAD

Give your command a name (I'm using 'Select Linked (RFL)') and enter the command '^C^C(RFL:SELECTLINKED) '



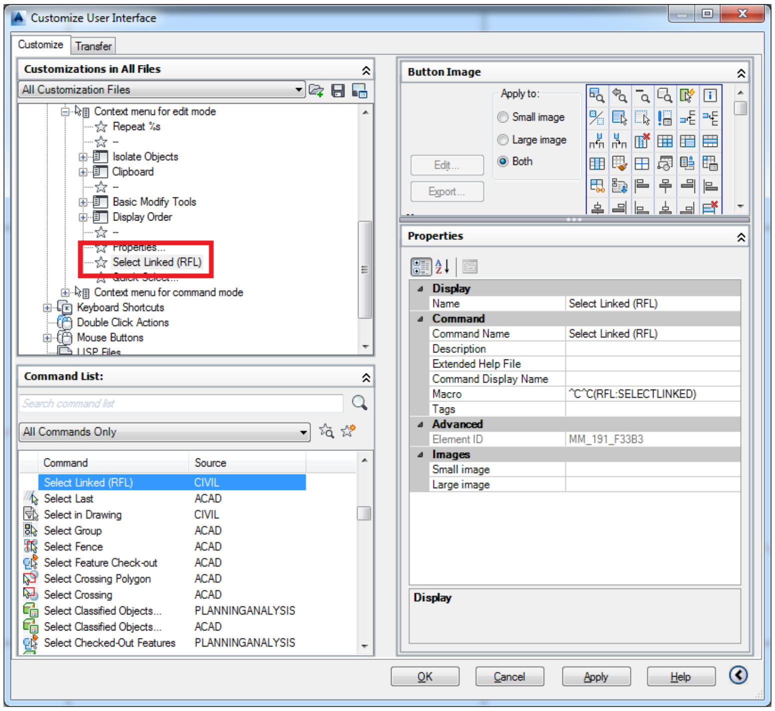
Command	Source
Select Linked (RFL)	CIVIL
Select Last	ACAD
Select in Drawing	CIVIL
Select Group	ACAD
Select Fence	ACAD
Select Feature Check-out	ACAD
Select Crossing Polygon	ACAD
Select Crossing	ACAD
Select Classified Objects...	PLANNINGANALYSIS
Select Classified Objects...	ACAD
Select Checked-Out Features	PLANNINGANALYSIS

Command	
Name	Select Linked (RFL)
Description	
Extended Help File	
Command Display Name	
Macro	^C^C(RFL:SELECTLINKED)
Tags	
Advanced	
Element ID	MM_191_F33B3
Images	
Small image	
Large image	

Macro
Specifies the command to run when the interface element is selected.

Remember to include the space after the closing bracket.

Next expand the ‘Shortcut Menus’, locate and expand ‘Context menu for edit mode’ and then drag the new command to your desired location in the list



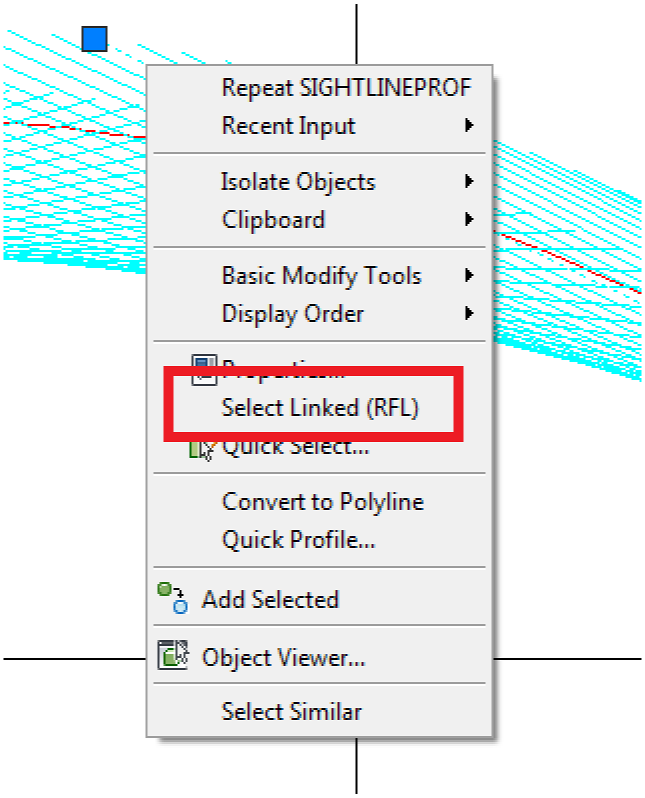
Command	Source
Select Linked (RFL)	CIVIL
Select Last	ACAD
Select in Drawing	CIVIL
Select Group	ACAD
Select Fence	ACAD
Select Feature Check-out	ACAD
Select Crossing Polygon	ACAD
Select Crossing	ACAD
Select Classified Objects...	PLANNINGANALYSIS
Select Classified Objects...	ACAD
Select Checked-Out Features	PLANNINGANALYSIS

Apply to:
○ Small image
○ Large image
● Both

Display	
Name	Select Linked (RFL)
Command	
Command Name	Select Linked (RFL)
Description	
Extended Help File	
Command Display Name	
Macro	^C^C(RFL:SELECTLINKED)
Tags	
Advanced	
Element ID	MM_191_F33B3
Images	
Small image	
Large image	

Display

Press ‘OK’ and you will have the command available when you have an entity highlighted


Repeat SIGHTLINEPROF
Recent Input
Isolate Objects
Clipboard
Basic Modify Tools
Display Order
Properties...
Select Linked (RFL)
Quick Select...
Convert to Polyline
Quick Profile...
Add Selected
Object Viewer...
Select Similar