500166

Parametric Modeling and Automating the Placement of Universal TBM Rings

Atul Tegar
COWI

Learning Objectives

- Parametric modelling of Universal ring
- Automate the placement of universal ring along alignment
- Exploring CivilConnection package in Dynamo for TBM Tunnel
- Create Python script in Dynamo

Description

TBM Tunnels are made up of prefabricated concrete segments which form a ring to be repeated in series, the rings laid in succession follow the course of the alignment. Autodesk Civil 3D is a great tool for infrastructure works like roads, pipelines also for modeling linear structures (bridge, tunnel, retaining wall), and Autodesk Revit is good for parametrical modeling. This class covers the use of Autodesk Revit together with Dynamo and CivilConnection package to leverage the power of Civil 3D inside Revit to automate the process of placement of universal TBM ring along alignment while minimizing the deviation in the placement of TBM ring along the alignment. In this class, I will demonstrate the parametrical modeling of universal ring and placement and updating parameters (e.g. rotation of ring) of the ring along alignment using Dynamo.

Speaker(s)

Atul Tegar

Atul Tegar is a BIM Specialist at COWI, Denmark. He is a graduated Civil Structural Engineer and over the years has gained an in-depth knowledge of BIM modeling and management tools. He has an experience of more than 11 years in the field of construction, project management, BIM coordination and modeling of infrastructure and building works. He is an Autodesk AutoCAD Civil 3D 2015 – Certified Professional. He also worked for AU 2016 class – TR21202 Model Linear Structures: Aligning AutoCAD Civil 3D and Revit with Dynamo for Viaducts and Tunnels as a team member for modeling and setting up standards in Civil 3D. He is also continuously contributing towards the development of CivilConnection package for Dynamo Revit published by Autodesk.

About COWI

Our Solutions

- **INFRASTRUCTURE:** We are among the leading consultants within complex infrastructure design with particular strongholds within bridges, tunnels and marine structures. We are ranked #5 within Marine and Port facilities*.

- **BUILDINGS:** Our expertise covers the hospitals, commercial, industry, residential, cultural and educational buildings and transportational hubs. We are ranked #8 within buildings*.

- **ENVIRONMENT:** We are ranked as the international market leader within solid waste management*. We work with contaminated sites, nature, sustainability, climate change,urbanization, environmental impact assessment, strategic environmental consultancy and environment, health & safety.

- **WATER:** We work with all aspects of water sector development, climate adaptation and coastal protection.

- **ENERGY:** We work with all aspects of energy planning, production, and distribution within bioenergy and thermal power, wind, solar and renewable energy and oil & gas.

- **INDUSTRY:** We design industrial plants with extensive know-how. The result is an optimized production where efficiency meets sustainability.

- **PLANNING:** We offer planning with evidence-based decision support. Thus, we help establish the ground for sustainable infrastructure development and overall sustainable economic development.

* According to Engineering News Record ranking

COWI Iconic Projects

- Copenhagen Metro, Denmark

- Oslo Airport Expansion, Norway

- London Array, United Kingdom

- New NY (Tappan Zee) Bridge, New York, USA

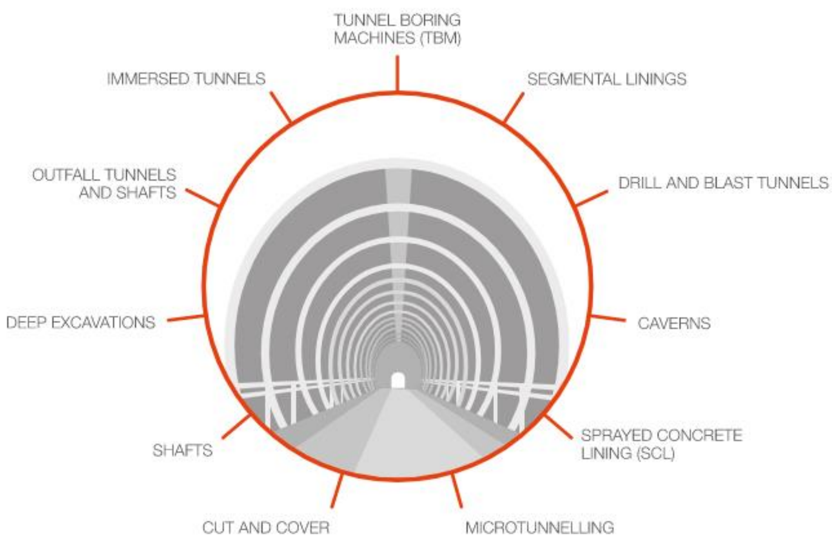
- Osman Gazi Bridge, Turkey

- Stonecutters Bridge, Hong Kong

COWI Tunnels

COWI Tunnels is a world leading specialist in tunnel & underground engineering and geotechnical design

- Immersed tunnels
- Bored tunnels and shafts
- SCL and rock tunnelling
- Cut-and-cover tunnels
- Underground stations, parks and storage
- Operations and maintenance
- Risk management and analysis
- Service-life design
- MEP



- COWI Tunnels has more than 450 skilled employees, making it one of the largest tunnel groups
- COWI Tunnels has a global pool of specialists covering multiple technical fields and are able to deliver large and complex multidisciplinary projects

COWI

Tunnel Market Segments


METRO

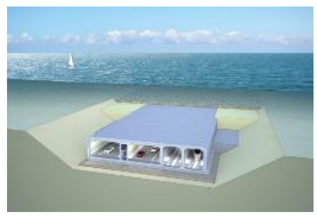

ROAD & RAIL CORRIDORS


UTILITY NETWORKS


CAVERNISATION


GROUND ENGINEERING


MEP


IMT


HYDROPOWER

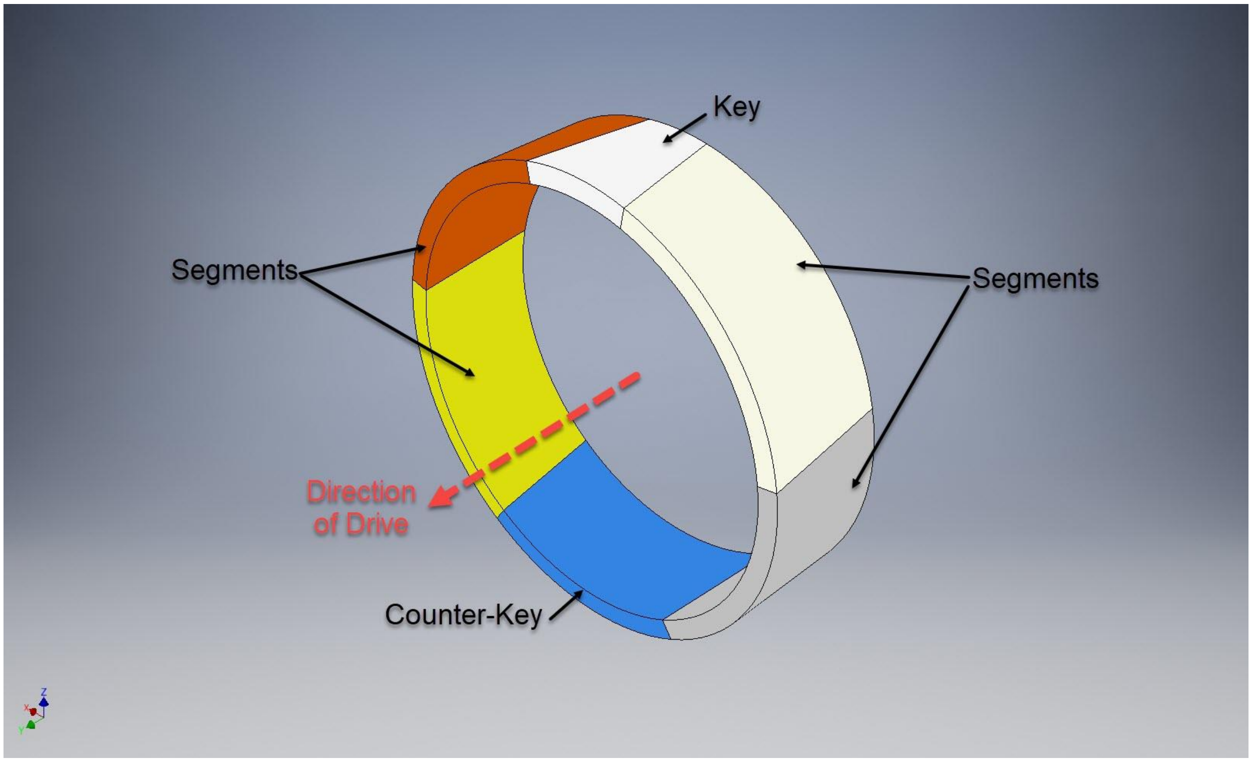

DEEP BASEMENTS &
RETAINING STRUCTURES

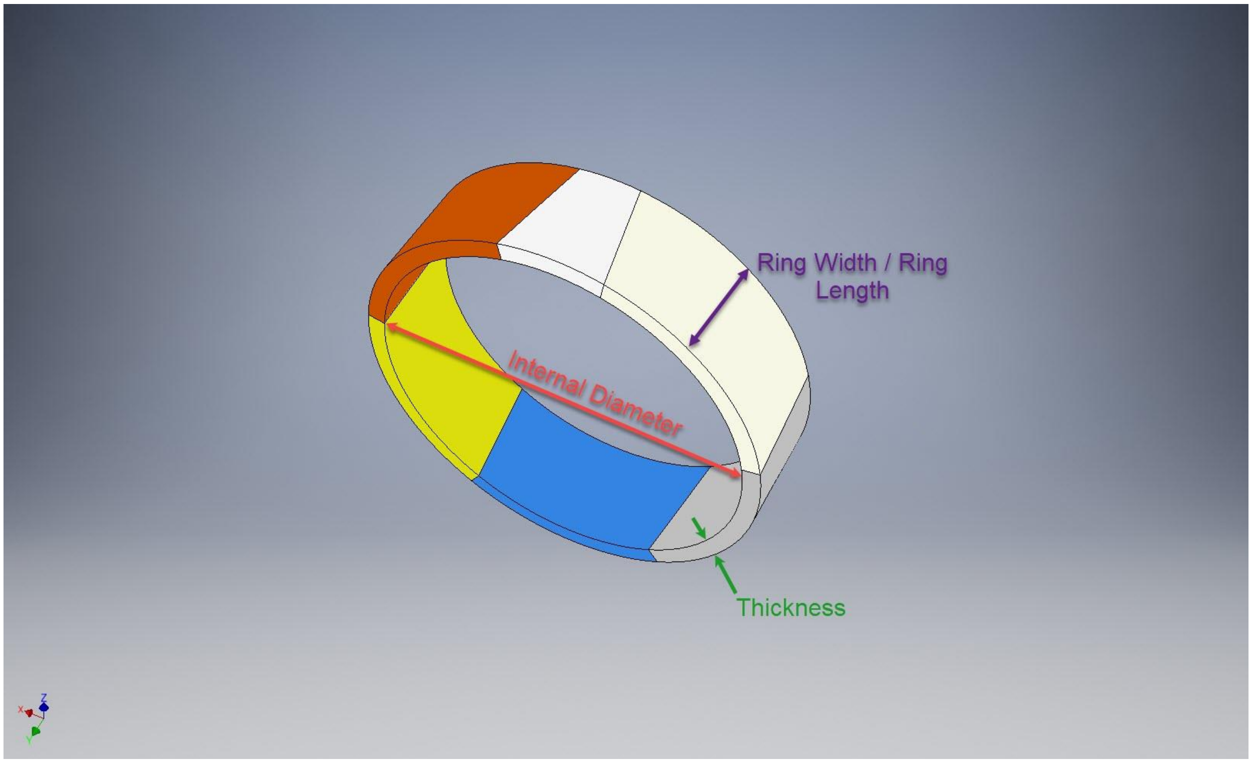

CONCRETE TECHNOLOGY

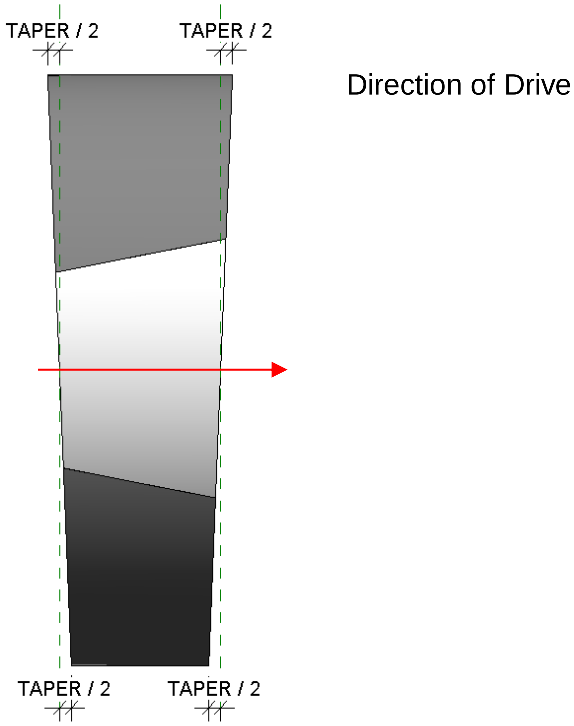
Introduction

TBM tunnels are made up of prefabricated concrete segments, which form a ring to be repeated in series. The rings laid in succession through mechanical connectors follow the course of the alignment. The taper of the ring is ensured by taking into account the minimum radius of curvature of the alignment.

Elements of the Ring







Universal Ring with taper on both sides

Segment Geometry
- Rectangular
- Trapezoidal
- Rhomboidal or parallelogrammical

Key Geometry
Trapezoidal

Segmental Ring Systems

Three types of ring systems are commonly used:

a. Parallel rings with corrective rings
   This ring system comprising rings with parallel end faces.
b. Right/Left rings
   In this ring system, one face is slanted or tapered to follow the course of alignment. One ring is tapered on the right side
c. Universal ring
   This ring system comprising of ring having taper at both of the end faces

Ring Configuration
A ring consists number of segments. Shorter the segments make it easier to handle and transport but creating more number of joints in larger diameter of the ring.

While longer segments produce less joints which forms a stiffer segmental ring but segments will not be easy to handle and transport.

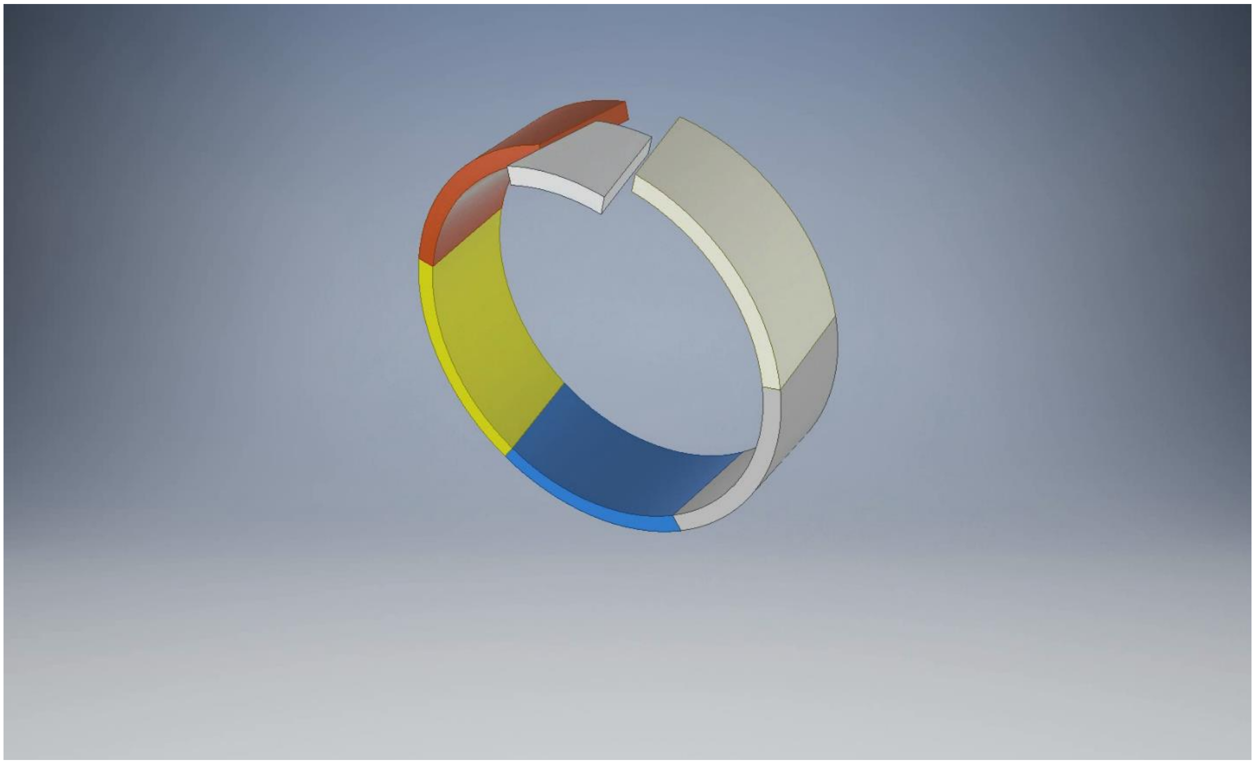
For smaller diameter of tunnel, ring with fewer segments to be used. e.g. 5+1 ring configuration means 5 regular segment and 1 key segments can be used for smaller diameter tunnel.

For medium diameter of tunnel 5+2 or 6+1 ring configuration can be used.

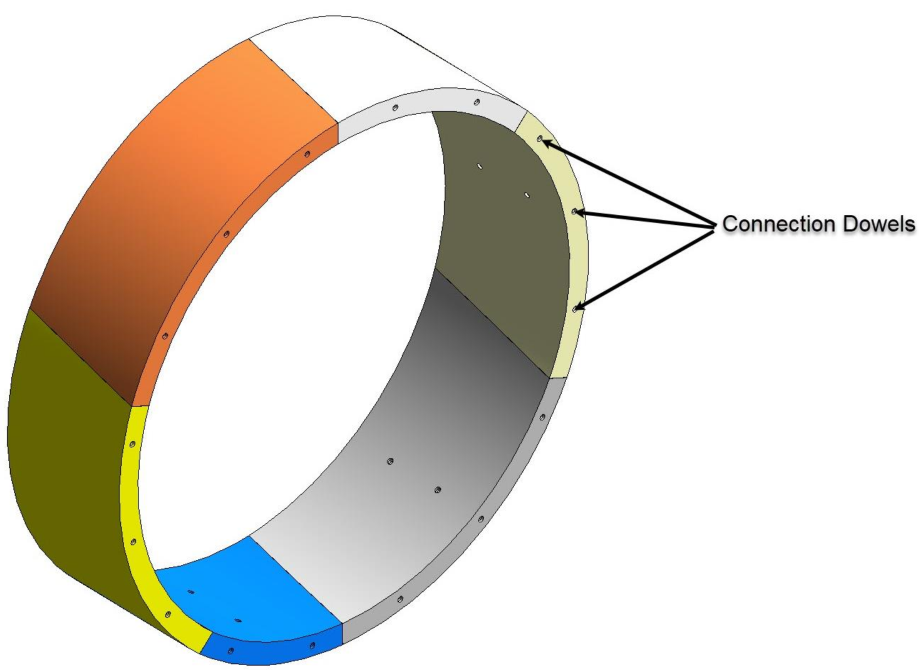
And for larger diameter of tunnel ring with 6 regular segments with 2 key segments or 7 regular segments with 1 key segment can be used.

Segmental Ring Installation
First the counter-key segment is installed followed by regular segments, and at the last key segment is inserted as shown below.

Reference : ACI 533.5R-20: Guide for Precast Concrete Tunnel Segments



Segmental Ring Installation



TBM Ring with Connection Dowels

Challenges in modeling and placement of ring along TBM Centreline

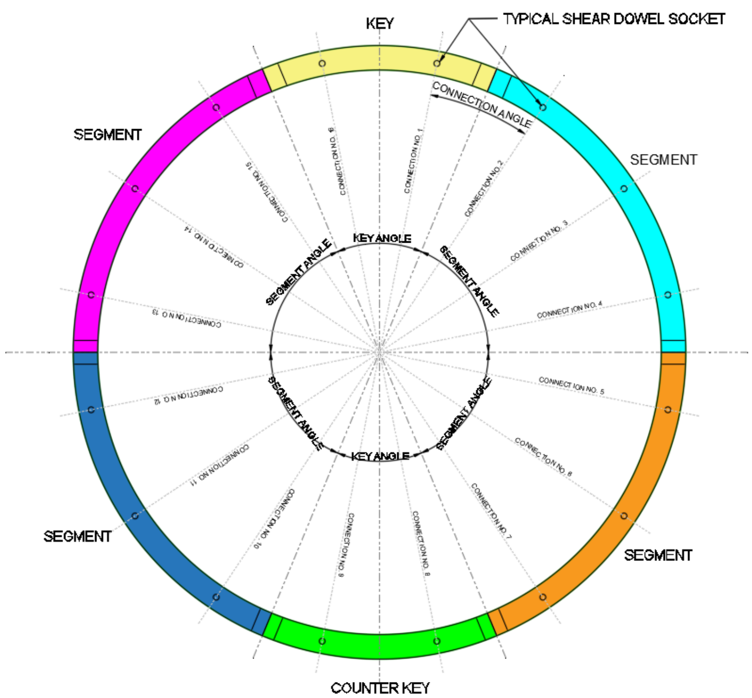
Rotation of subsequent rings to defined positions only

The placement of the next ring along alignment is govern by the course of alignment. The next ring is rotated to only specified angle which depends on the number of connections of each ring. e.g. if a ring have 16 number of connection dowels, this means the next ring can be placed by rotating it to these 16 different rotation angle. Angle of rotation is calculated by the below formula:

Connection Angle = 360^{o} / Number of connections

Therefore the position of each ring is calculated by its angle of rotation. For 0^{o} rotation of ring can be called as Position 1 and Position 2 when angle of rotation is (0^{o} + Connection Angle) and so on.

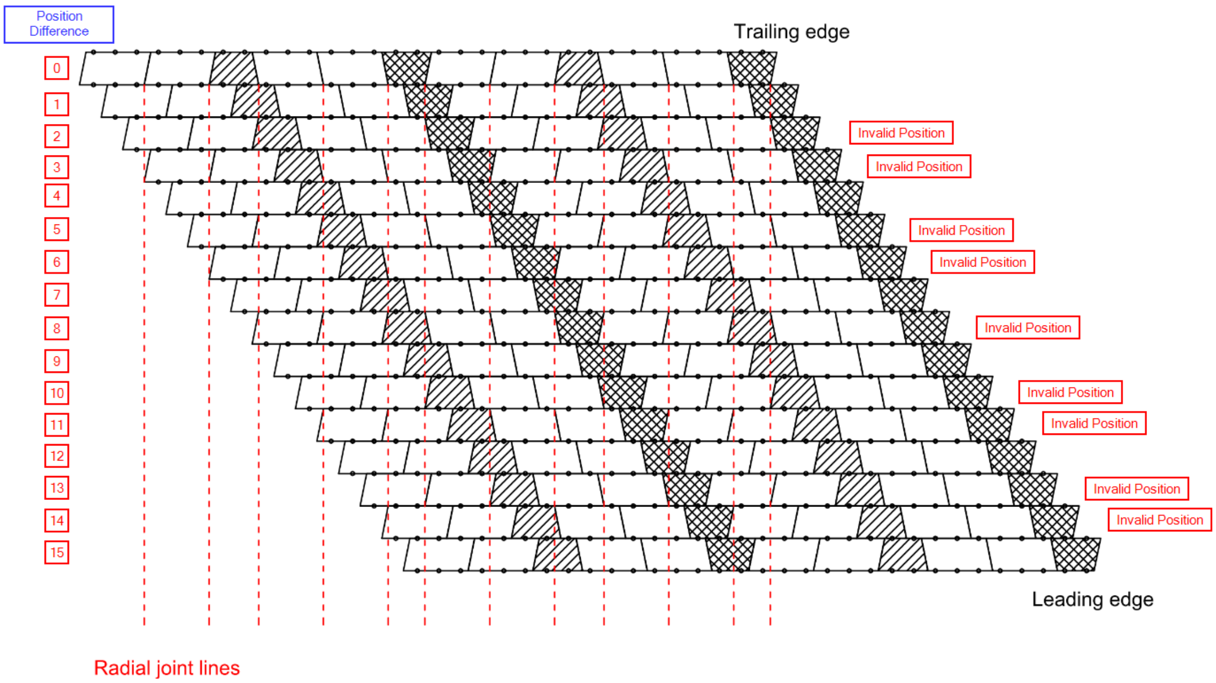
So the next ring can have maximum n different position.



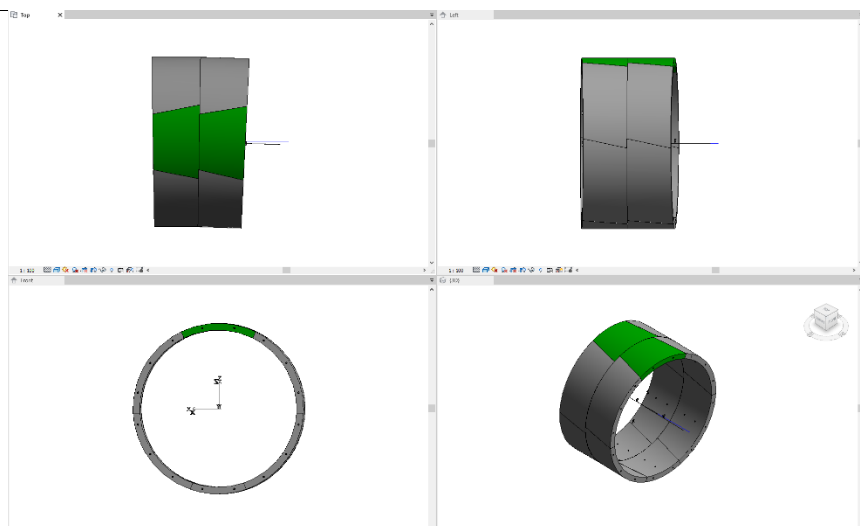
Positions without cruciform joints

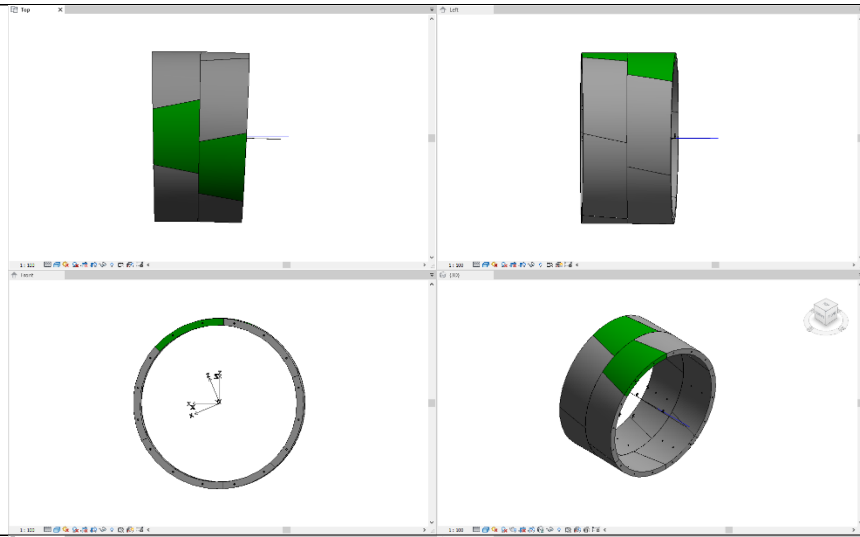
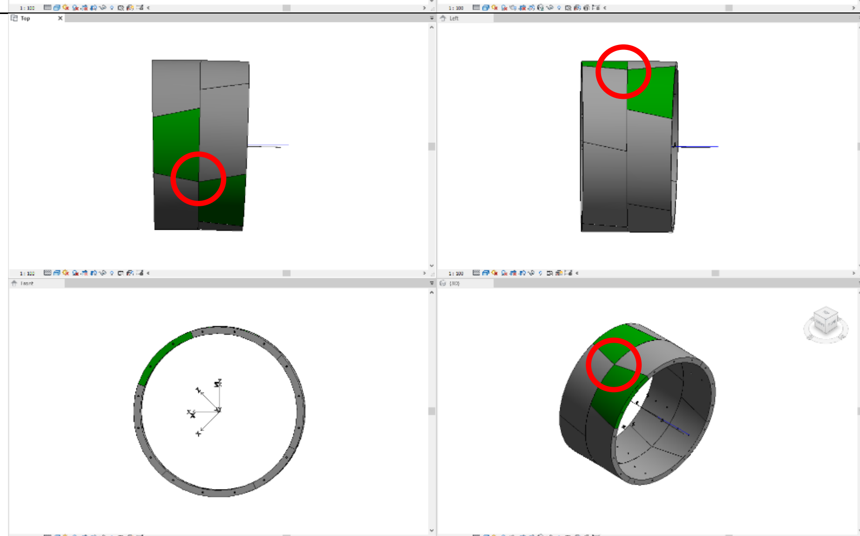
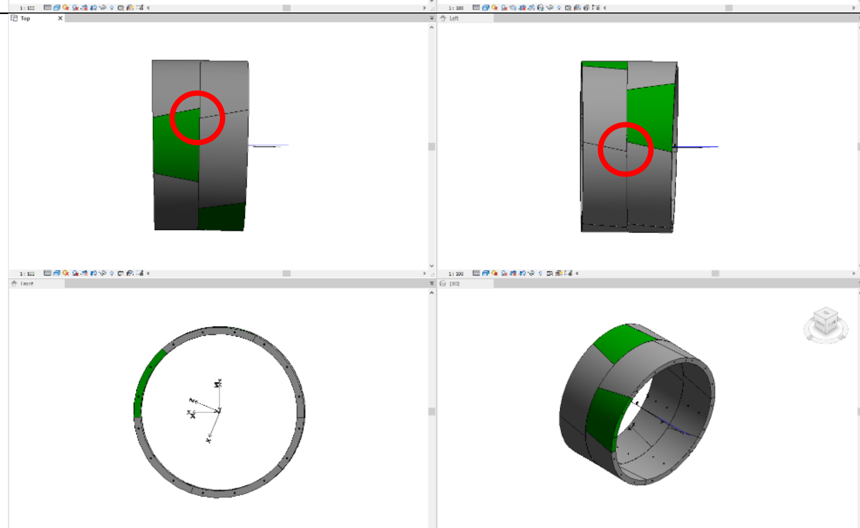
Some position of the ring will form cruciform joint with other ring. Below sketch shows the possible positions with or without cruciform joints with other ring.

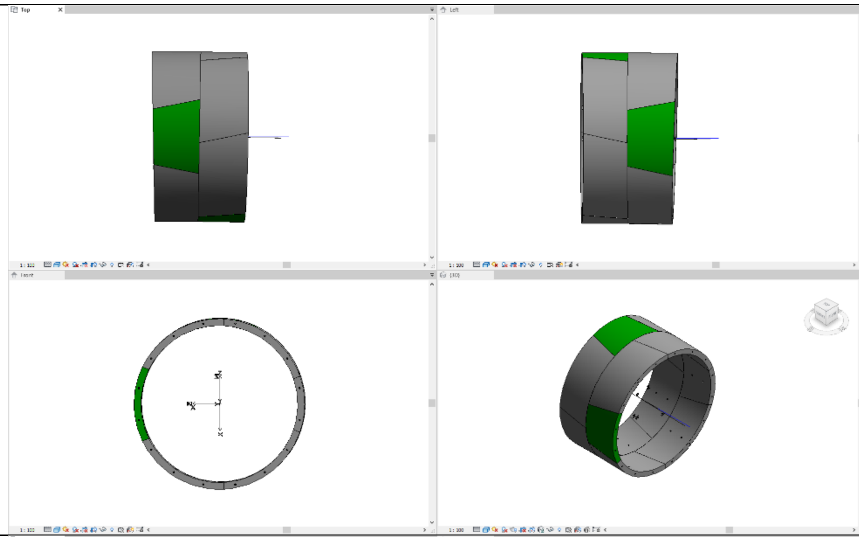
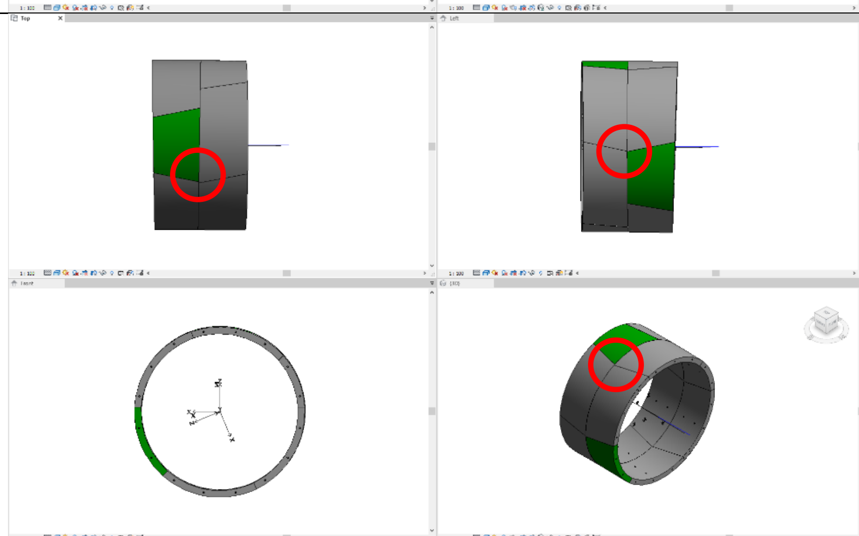
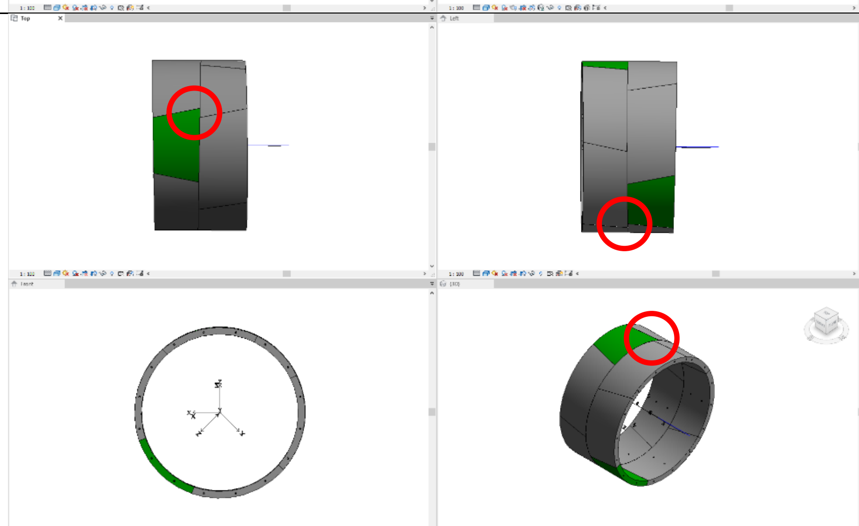
4+2 segmentation w/ two-third-size key & counter key

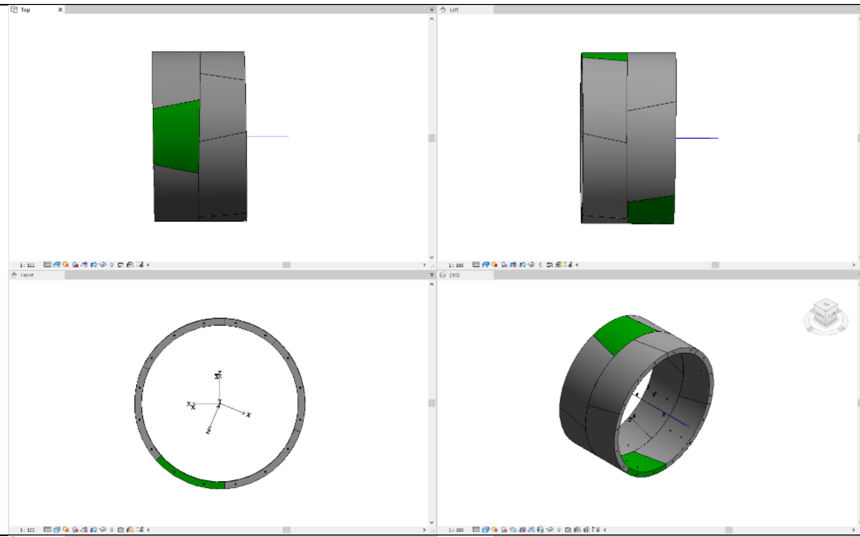
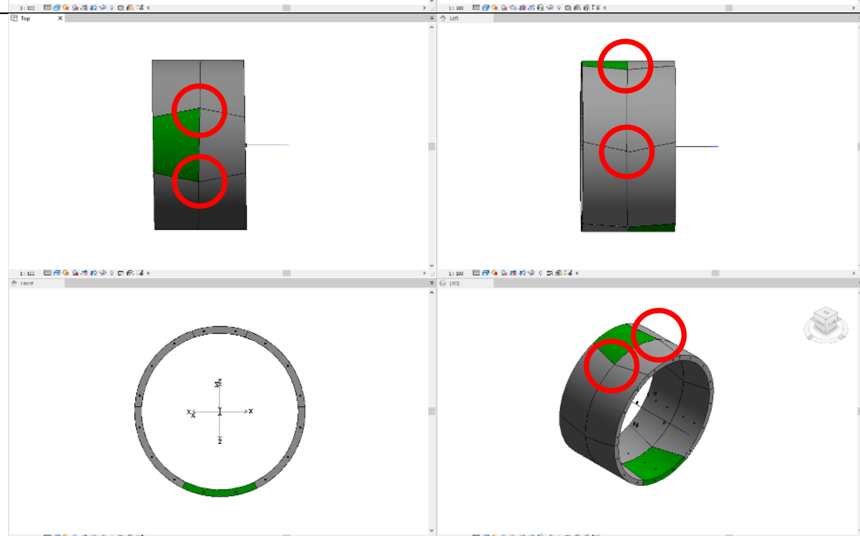
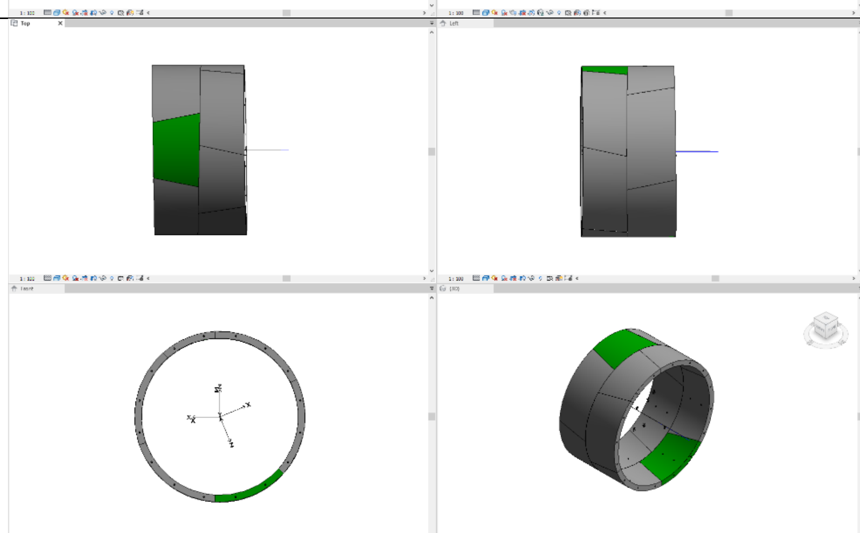


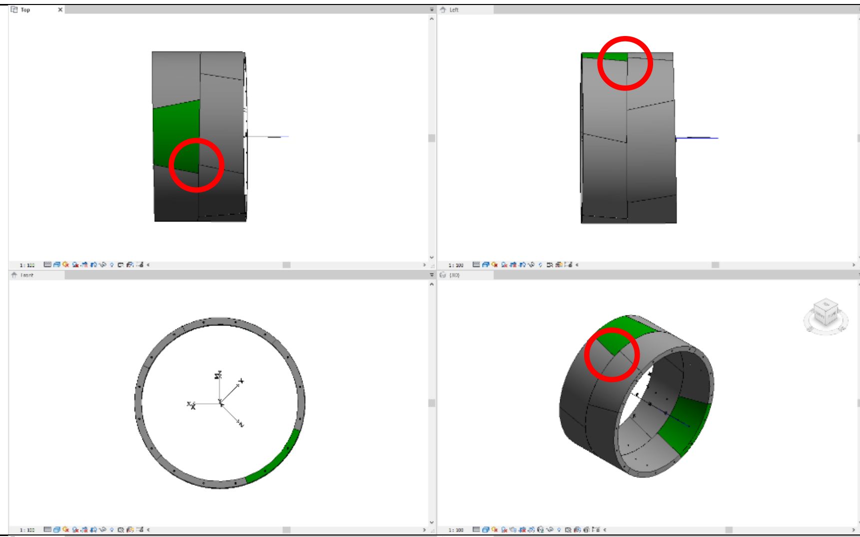
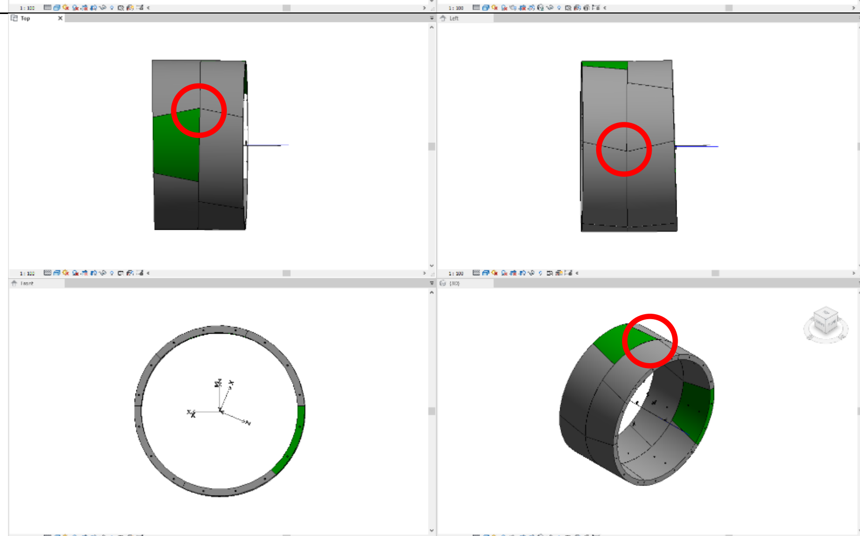
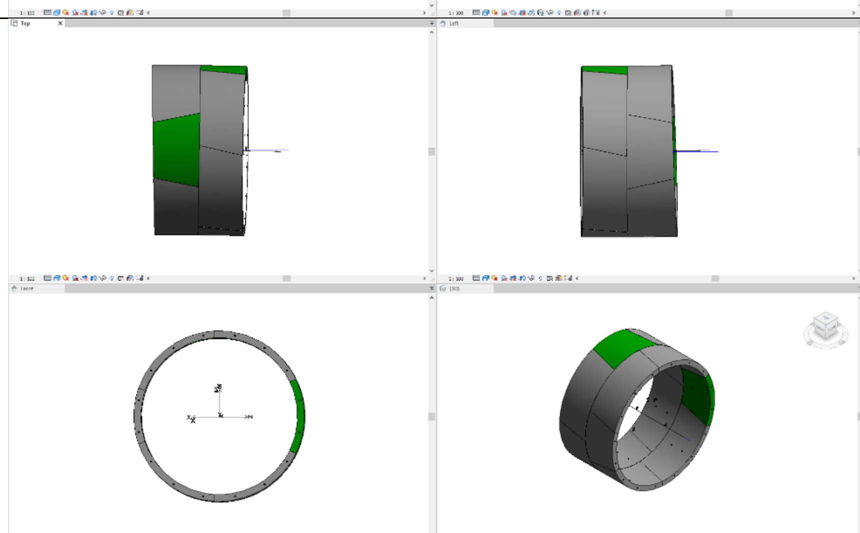
Different positions that will create cruciform joint with other ring are shown in the table below:

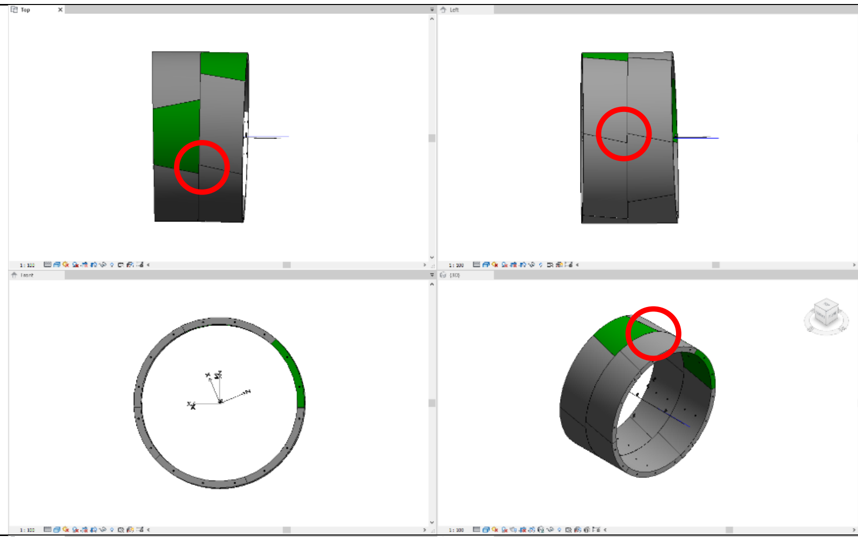
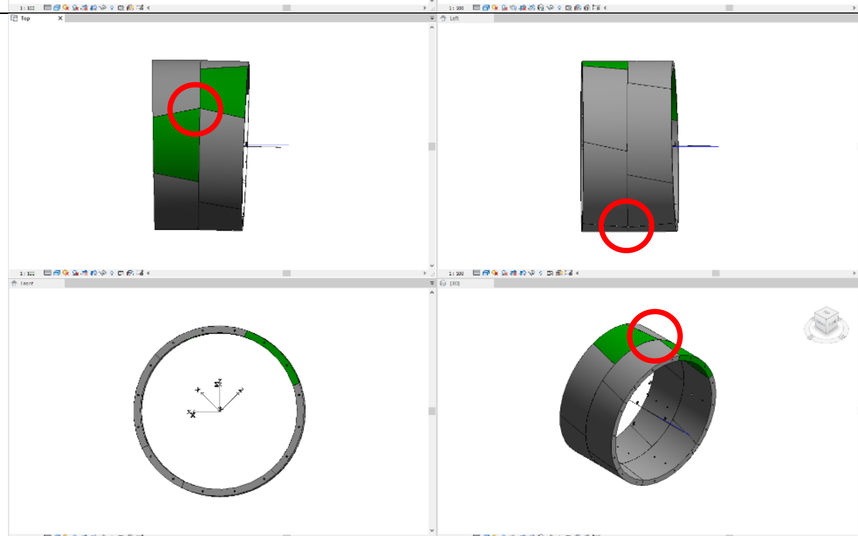
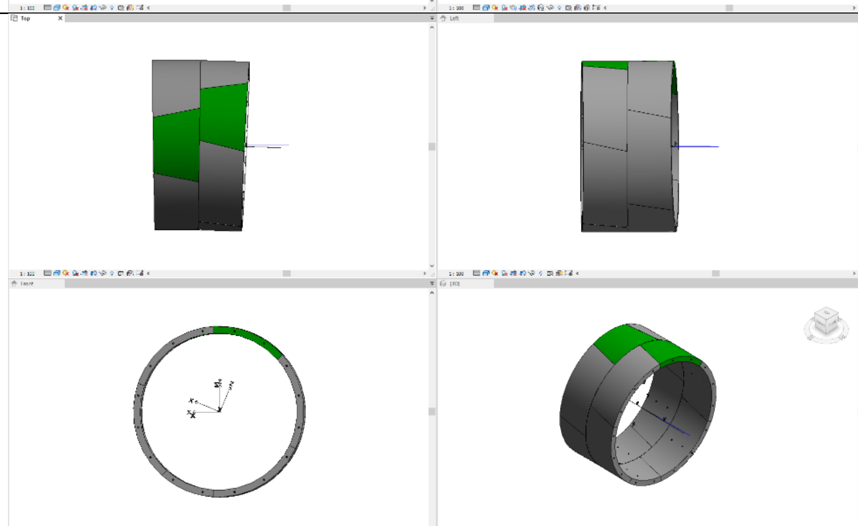
Position Difference		Valid Position
0		

1		✓
2		✗
3		✗

4		
5		
6		

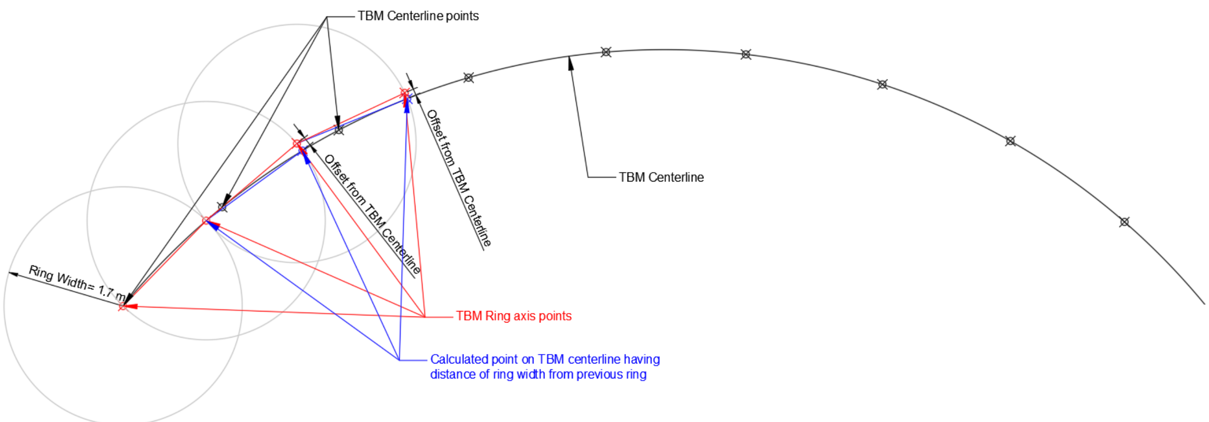
7		
8		
9		

10		
11		
12		

13		
14		
15		

Minimize deviation to TBM alignment

Rings are placed in succession by following a path along alignment also needs to be avoid cruciform joints. Below sketch shows the concept for calculation of position of next rings based on minimum deviation to TBM alignment. The valid position (by avoiding positions that can form cruciform joints) which is having minimum deviation from TBM alignment is selected.

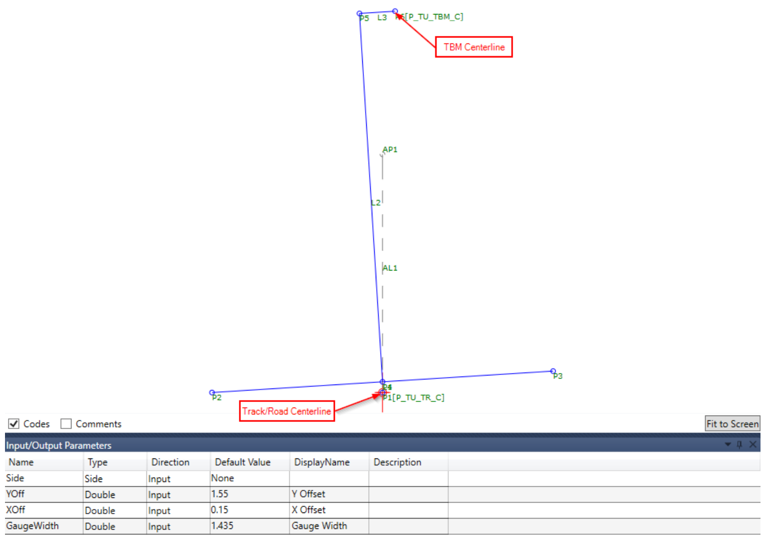


Solutions

TBM Centerline from Track Alignment

To get the centerline of TBM tunnel, a subassembly is created in Autodesk Subassembly Composer

Subassembly

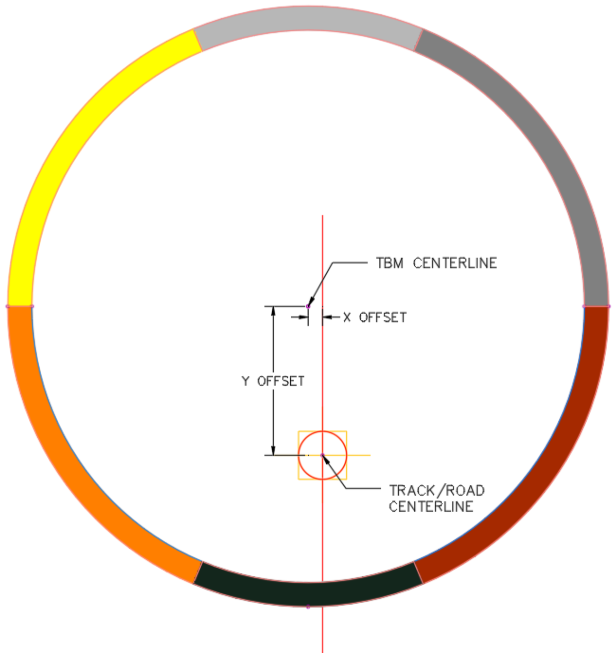


Name	Type	Direction	Default Value	DisplayName	Description
Side	Side	Input	None		
YOff	Double	Input	1.55	Y Offset	
XOff	Double	Input	0.15	X Offset	
GaugeWidth	Double	Input	1.435	Gauge Width	

The TBM Centerlinepoint is calculated based on the X and Y offset from the track centerline or Top of Rail, it is also taking care of cant or superelevation of the alignment along curves.

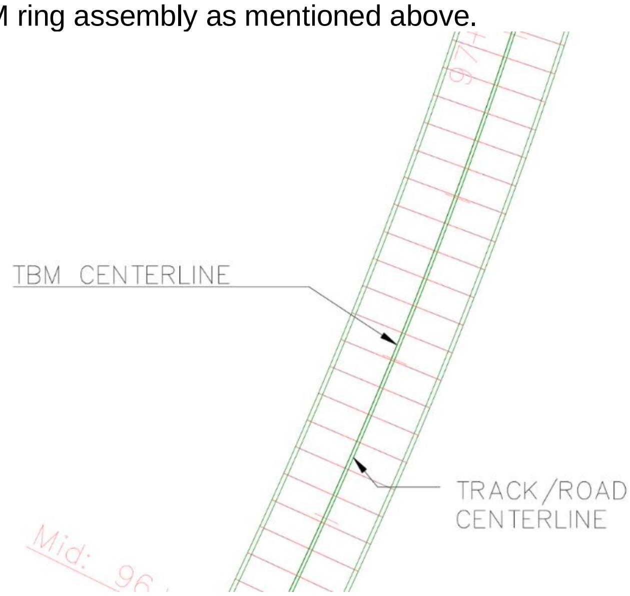
Assembly

The assembly consists of TBM Centerline subassembly with or without subassembly for TBM ring. This assembly will be used to create corridor in Civil 3D



TBM Corridor in Civil 3D

TBM corridor is created by using track or road alignment together with Top of Rail of design profile of road and TBM ring assembly as mentioned above.

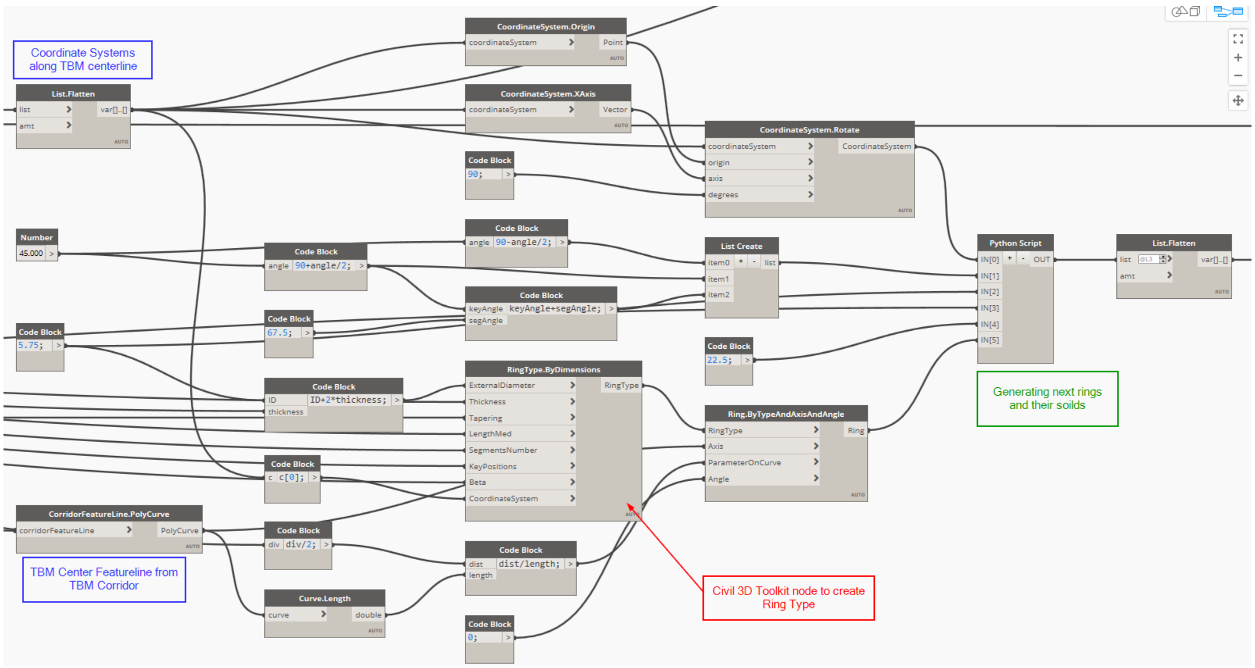


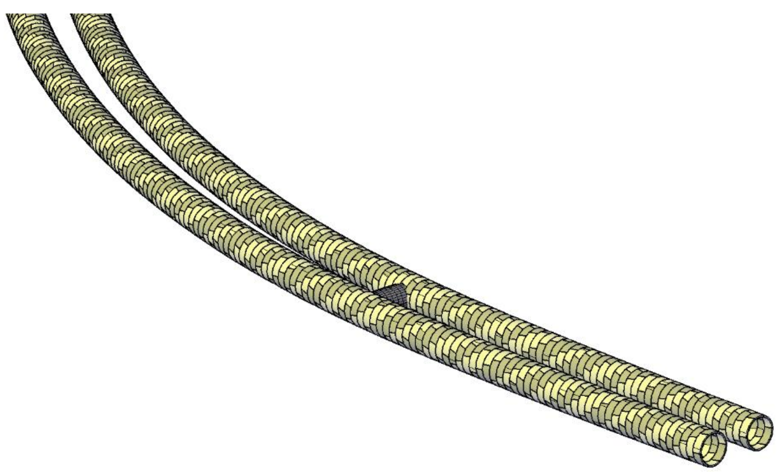
TBM Tunnel Corridor in Civil 3D

TBM centerline will be generated as Corridor featureline as shown above which is to be used to place the TBM rings later inside Revit.

Civil 3D toolkit -Dynamo for Civil 3D

Civil 3D toolkit is a package available inside Dynamo for Civil 3D. For the modeling of TBM Tunnel inside Civil 3D this Civil 3D toolkit package is used.
The rings are generated as solids which are later inserted as a block inside Civil 3D.





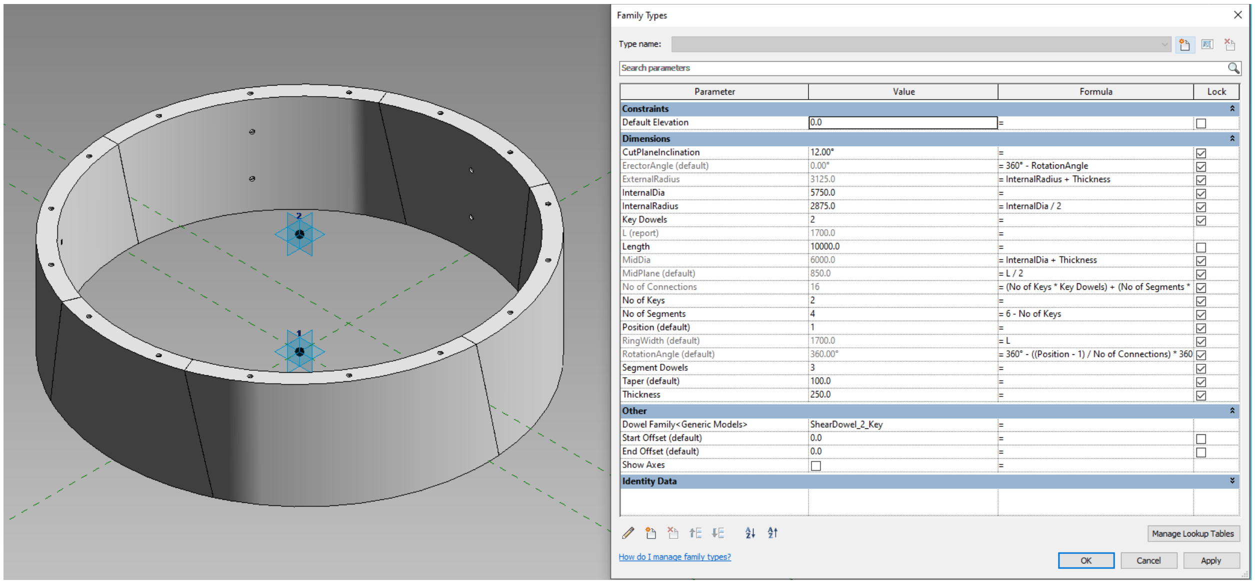
TBM model in Civil 3D using Dynamo with Civil3DToolkit

Limitations:

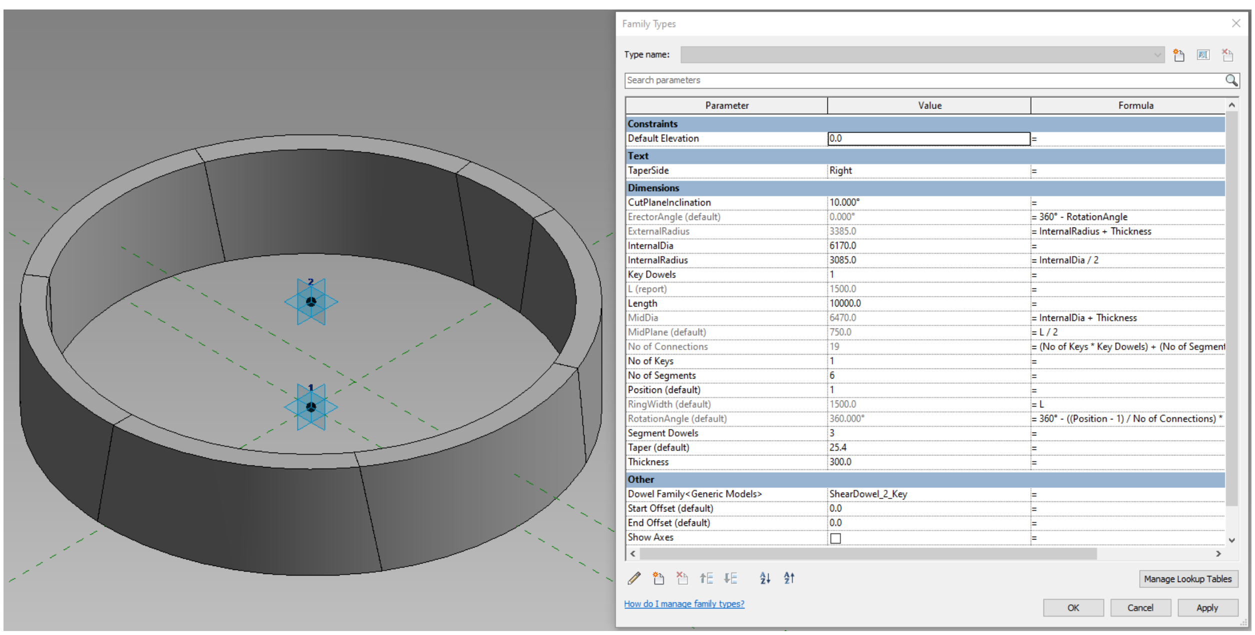
- Not parametric
- Generate as a 3D solid element, difficult to update the same element
- Joints and other connection details to be modelled separately

Universal Ring Revit Family

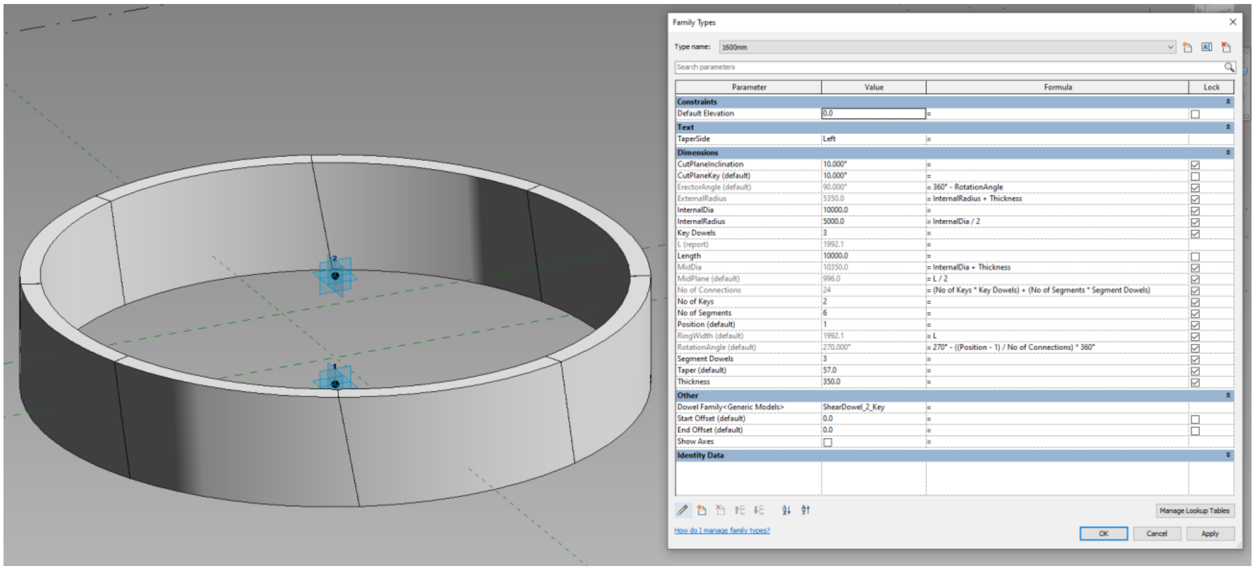
The universal ring family is created as a 2-point adaptive family in Revit with all the necessary parameters as shown below:



6 segments Ring Family

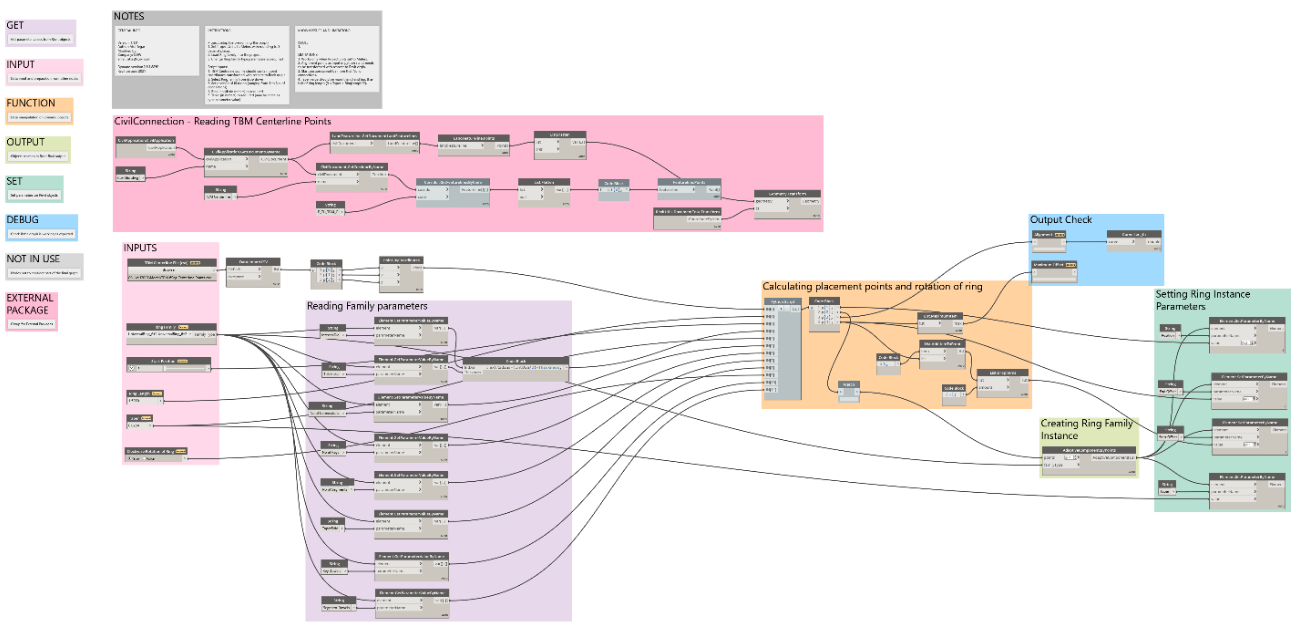


7 segments Ring Family



8 segments Ring Family

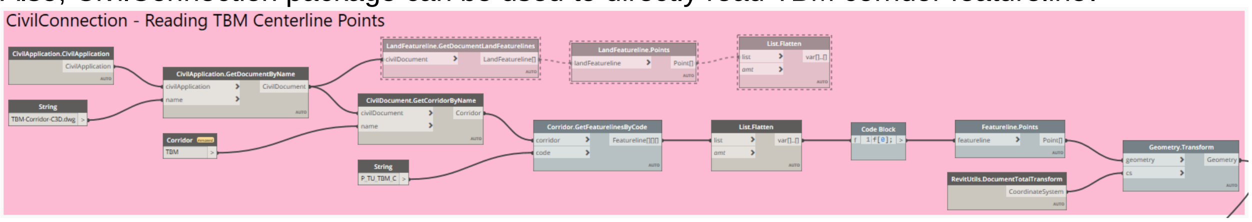
Dynamo script for placement of ring

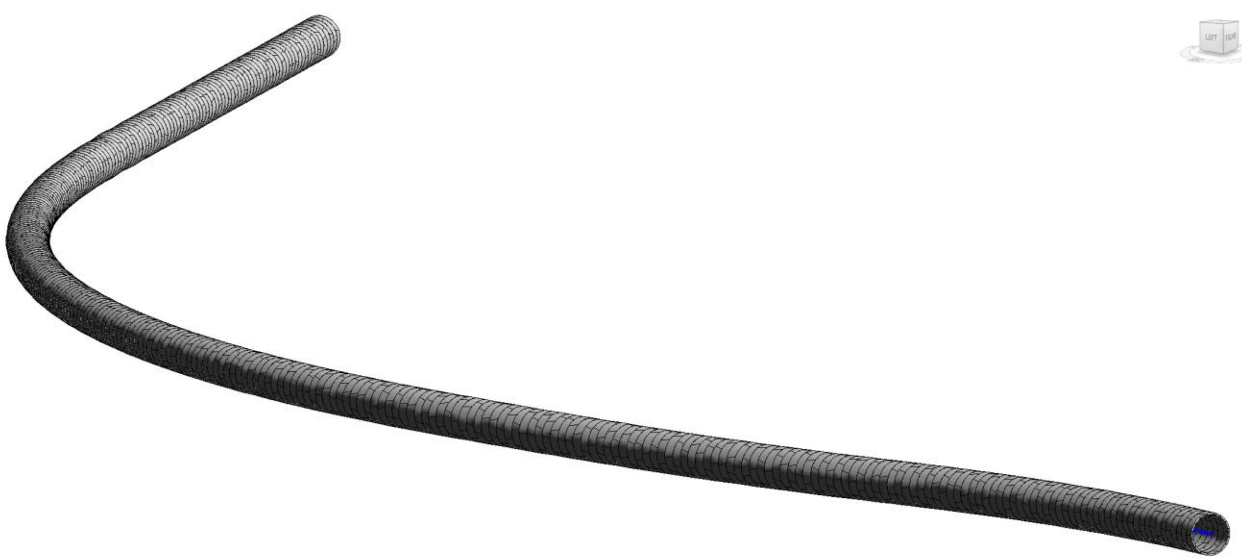


Inputs for Dynamo Script

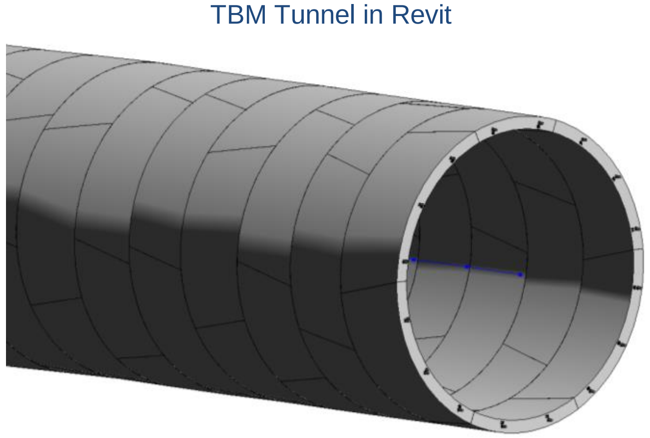
1. TBM Centerline points csv file (File Path)
2. Ring Family (Family Types)
3. Position of first ring (Integer Slider)
4. Ring Length in meters (Number)
5. Taper in meters (Number)
6. Clockwise Rotation of Ring (Boolean)

Also, CivilConnection package can be used to directly read TBM corridor featureline.





TBM Tunnel in Revit



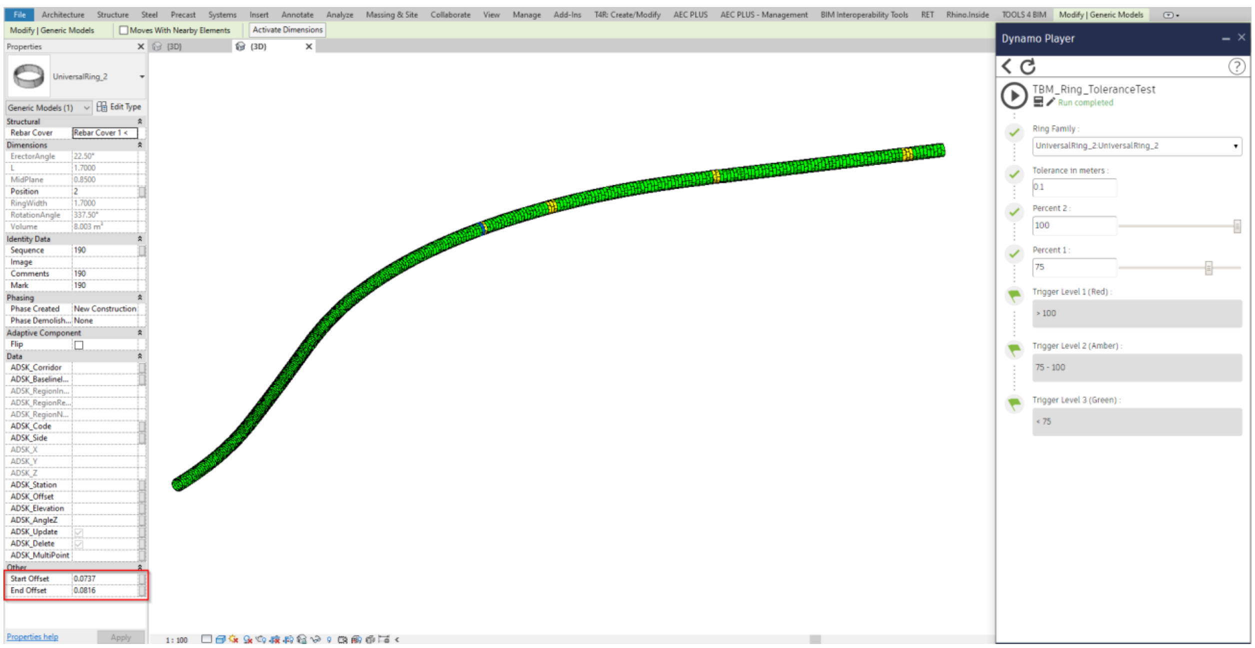
TBM Tunnel in Revit

Tolerance Check – Dynamo Script

A Dynamo script is used to override colors of each ring segments based on its tolerance level. for example the ring which is having StartOffset or EndOffset (offset or devation from TBM centerline) more than the tolerance limit, then that ring color will be override as Red color as shown below. The rings which are in the range of 75-100% tolerance limit will be colored as Amber, while rings below 75% tolerance limit will be colored as Green

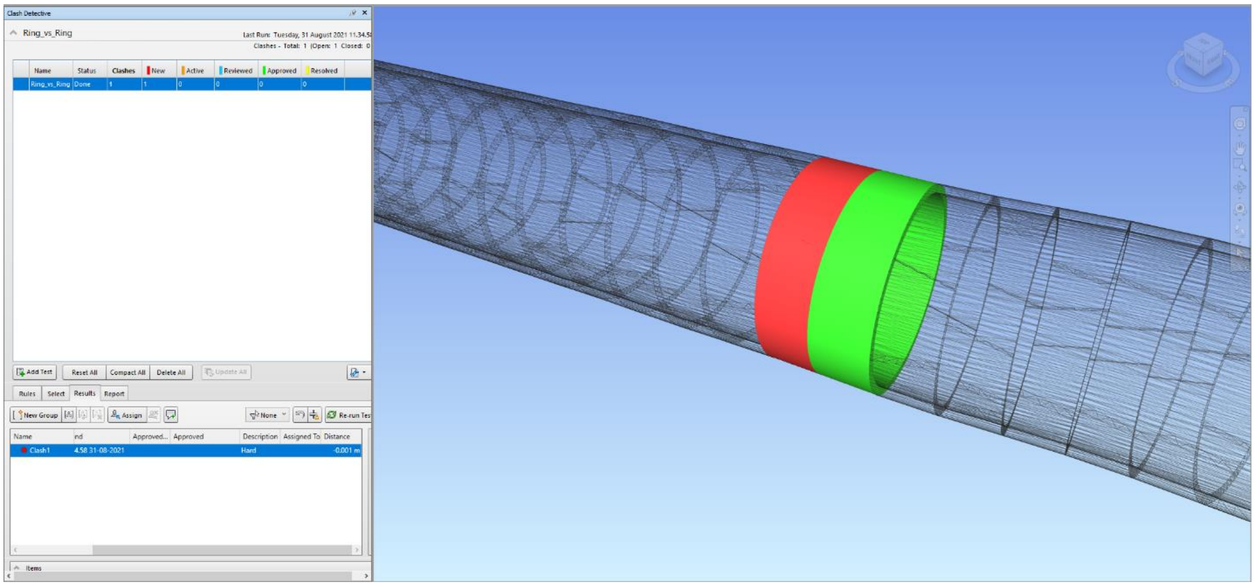
The inputs to this Dynamo Script are as follows:

1. Ring Family (Family Types)
2. Tolerance in meters (Number)
3. Percent 2 (Number slider)
4. Percent 1 (Number slider)

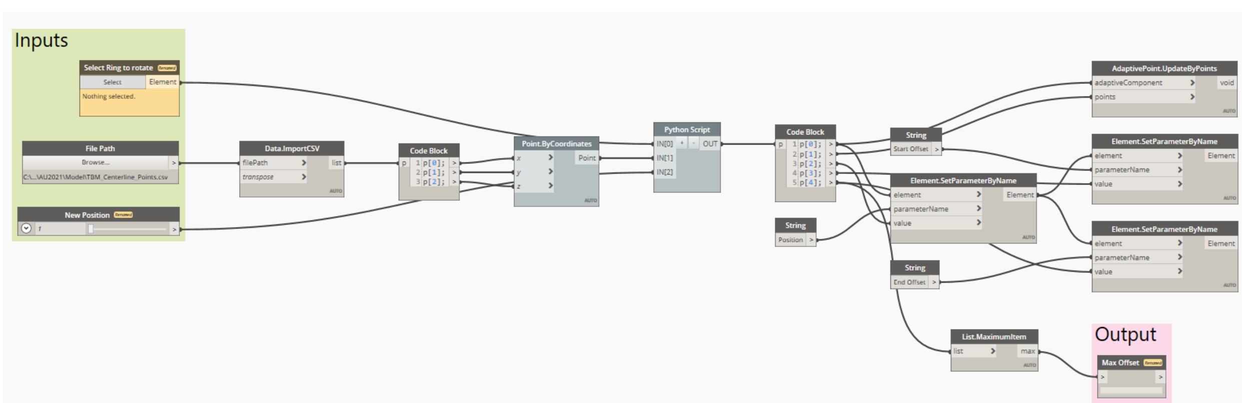


Clash check in Navisworks

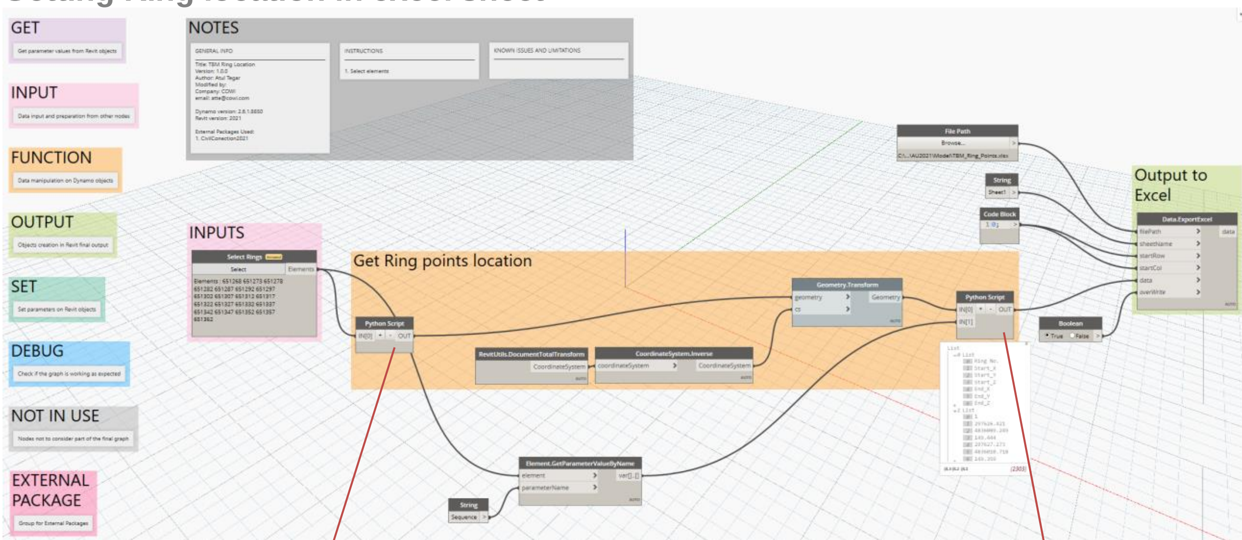
A clash coordinate model is prepared to check the clashes between rings. The clash test is a hard clash with tolerance of 0.001m



Rotation of individual ring with subsequent rings



Getting Ring location in excel sheet



Python Script

```python
# Enable Python support and load DesignScript library
import clr
import sys

# Import RevitAPI
clr.AddReference('RevitAPI')
from Autodesk.Revit.DB import *
# Import RevitAPIUI
clr.AddReference('RevitAPIUI')
# Import DocumentManager and TransactionManager
clr.AddReference('RevitServices')
import RevitServices
from RevitServices.Persistence import DocumentManager
from RevitServices.Transactions import TransactionManager

# Import Dynamo Python References
clr.AddReference('ProtoGeometry')
from Autodesk.DesignScript.Geometry import *
# Import RevitNodes
clr.AddReference('RevitNodes')

import Autodesk
from Autodesk.Revit.DB import ElementId

clr.AddReference("System")
from System.Collections.Generic import List

from Autodesk.Revit.DB import ElementId as ElementId


# Assign Labels to Revit Document and Application
doc = DocumentManager.Instance.CurrentDBDocument

# The inputs to this node will be stored as a list in the IN variables.
elements = IN[0]

outList = []

for ele in elements:
    pointList = []
    element = UnwrapElement(ele)
    placePointIds = AdaptiveComponentInstanceUtils.GetInstancePlacementPointElementRefIds(element)

    for i in range(len(placePointIds)):
        try:
            refPoint = doc.GetElement(placePointIds[i])

            refPointXYZ = refPoint.Position
            refPointX = round(Autodesk.Revit.DB.UnitUtils.ConvertFromInternalUnits
            (refPointXYZ.X,DisplayUnitType.DUT_METERS),3)
            refPointY = round(Autodesk.Revit.DB.UnitUtils.ConvertFromInternalUnits
            (refPointXYZ.Y,DisplayUnitType.DUT_METERS),3)
            refPointZ = round(Autodesk.Revit.DB.UnitUtils.ConvertFromInternalUnits
            (refPointXYZ.Z,DisplayUnitType.DUT_METERS),3)
            refRVTPoint = Point.ByCoordinates(refPointX,refPointY,refPointZ)
            pointList.append(refRVTPoint)
        except:
            pointList.append("Element not found")

    outList.append(pointList)
```

Run | Save Changes | Revert

Python Script

```python
# Load the Python Standard and DesignScript Libraries
import sys
import clr
clr.AddReference('ProtoGeometry')
from Autodesk.DesignScript.Geometry import *

# The inputs to this node will be stored as a list in the IN variables.
pointList = IN[0]
marksList = IN[1]


# List definintions
output = []
output.extend([["Ring No.","Start_X","Start_Y","Start_Z","End_X","End_Y","End_Z"]])

for i in range(len(pointList)):
    outPoints = []
    ringMark = marksList[i]
    outPoints.append(ringMark)
    for point in pointList[i]:
        pointX = point.X
        pointY = point.Y
        pointZ = point.Z
        outPoints.extend([pointX, pointY, pointZ])
    output.append(outPoints)

# Place your code below this line

# Assign your output to the OUT variable.
OUT = output
```

Run | Save Changes | Revert

	A	B	C	D	E	F	G
1	Ring No.	Start_X	Start_Y	Start_Z	End_X	End_Y	End_Z
2	1	297626.421	4836009.249	149.444	297627.273	4836010.718	149.359
3	2	297627.273	4836010.718	149.359	297628.126	4836012.185	149.264
4	3	297628.126	4836012.185	149.264	297628.949	4836013.67	149.183
5	4	297628.949	4836013.67	149.183	297629.78	4836015.152	149.108
6	5	297629.78	4836015.152	149.108	297630.603	4836016.636	149.014
7	6	297630.603	4836016.636	149.014	297631.394	4836018.139	148.945
8	7	297631.394	4836018.139	148.945	297632.192	4836019.639	148.882
9	8	297632.192	4836019.639	148.882	297632.998	4836021.132	148.775
10	9	297632.998	4836021.132	148.775	297633.785	4836022.635	148.668
11	10	297633.785	4836022.635	148.668	297634.539	4836024.156	148.586
12	11	297634.539	4836024.156	148.586	297635.291	4836025.679	148.514
13	12	297635.291	4836025.679	148.514	297636.057	4836027.195	148.428
14	13	297636.057	4836027.195	148.428	297636.803	4836028.719	148.341
15	14	297636.803	4836028.719	148.341	297637.531	4836030.253	148.255
16	15	297637.531	4836030.253	148.255	297638.25	4836031.792	148.17
17	16	297638.25	4836031.792	148.17	297638.978	4836033.326	148.088
18	17	297638.978	4836033.326	148.088	297639.693	4836034.865	147.991
19	18	297639.693	4836034.865	147.991	297640.385	4836036.417	147.932
20	19	297640.385	4836036.417	147.932	297641.068	4836037.972	147.874
21	20	297641.068	4836037.972	147.874	297641.74	4836039.53	147.765
22	21	297641.74	4836039.53	147.765	297642.422	4836041.083	147.653
23	22	297642.422	4836041.083	147.653	297643.09	4836042.644	147.577
24	23	297643.09	4836042.644	147.577	297643.726	4836044.218	147.486
25	24	297643.726	4836044.218	147.486	297644.37	4836045.789	147.389
26	25	297644.37	4836045.789	147.389	297645.007	4836047.363	147.312
27	26	297645.007	4836047.363	147.312	297645.642	4836048.938	147.245
28	27	297645.642	4836048.938	147.245	297646.283	4836050.511	147.169
29	28	297646.283	4836050.511	147.169	297646.882	4836052.099	147.063
30	29	297646.882	4836052.099	147.063	297647.447	4836053.699	146.972
31	30	297647.447	4836053.699	146.972	297648.027	4836055.296	146.896
32	31	297648.027	4836055.296	146.896	297648.604	4836056.892	146.81
33	32	297648.604	4836056.892	146.81	297649.18	4836058.49	146.734
34	33	297649.18	4836058.49	146.734	297649.761	4836060.085	146.649
35	34	297649.761	4836060.085	146.649	297650.3	4836061.695	146.555
36	35	297650.3	4836061.695	146.555	297650.842	4836063.304	146.472
37	36	297650.842	4836063.304	146.472	297651.385	4836064.912	146.378

Ring No. with Start Point & End Point in Excel

Conclusion

- Parametric Ring family
- Ring location and position is calculated based on the minimum deviation from the TBM alignment or Centreline.
- Rotation is limited to defined positions based on number of connections, and by avoiding cruciform joint between adjacent ring
- Easy to update whole tunnel
- Joint other connection details can be implemented in Ring family afterwards based on desired LOD.
- Other MEP equipment can be hosted on individual rings.
- Easy to extract parameters, quantities, location of the rings
- Individual rings can be divided in separate segments (parts)