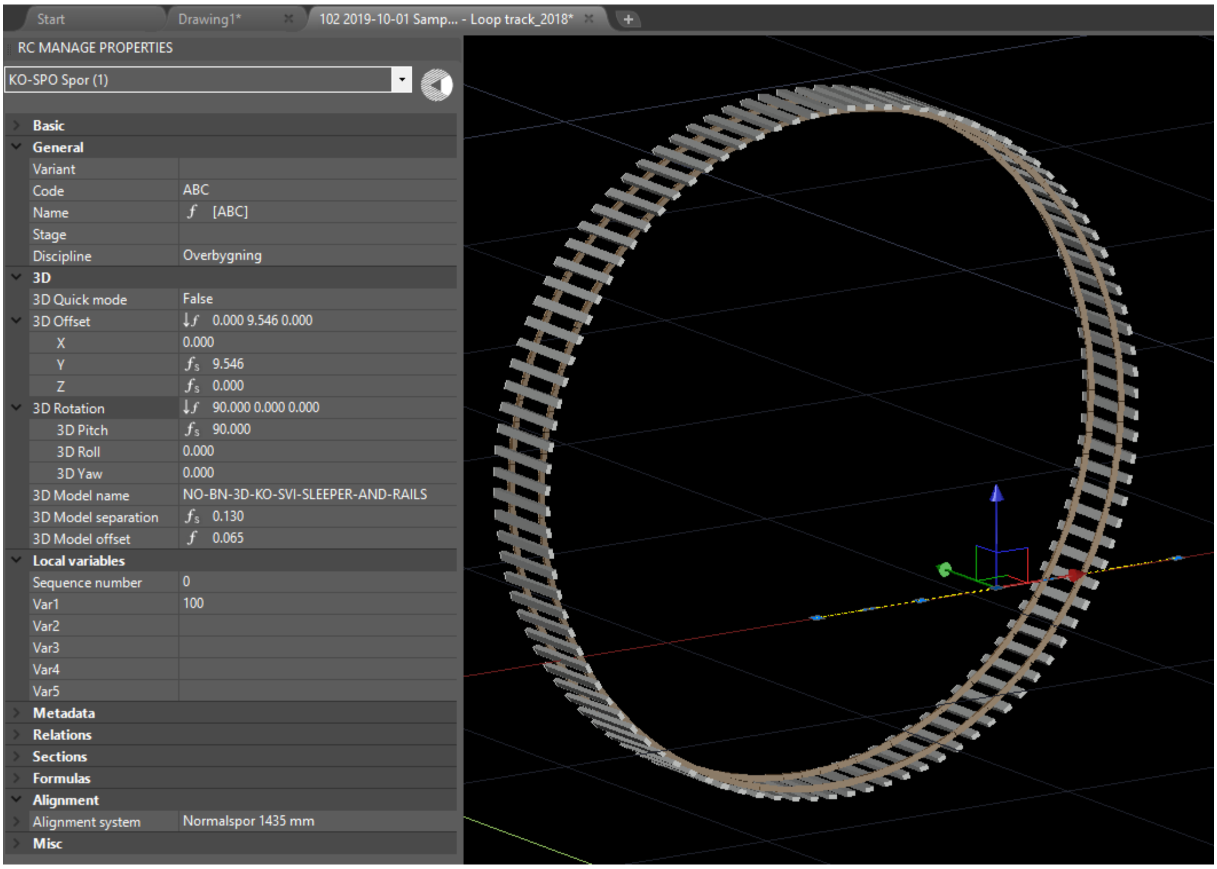
RC tutorial 102 - Sampled formulas - Loop track
==

Revised 2019-10-03

- This tutorial's goal is to teach you advanced use of sampled formulas.
  While making a railway loop might not be generally useful, it will really bring you into the intended RailCOMPLETE concepts.
- Assumed skills: Basic Lua programming, alignments and the RC alignment manager.
- Assumed Railway skills: None
- Time to spend here: Expert: 30 minutes. Intermediate: 1 hour. Beginner: 2-3 hours.

- Notice to users with non-English versions of AutoCAD – see footnote[1].
- This tutorial was prepared using software release 2018.22.1302 with Norwegian DNA version “2019.1 gamma”,”NO-BN;NO-0001;2019-09-28T20:19:00+01:00;2019.1”.
- Note: If you are using AutoCAD version 2017 or older, then open the 2013-format version of the tutorial DWG file.



RC MANAGE PROPERTIES

KO-SPO Spor (1)

Basic	
General	
Variant	
Code	ABC
Name	ƒ [ABC]
Stage	
Discipline	Overbygning
3D	
3D Quick mode	False
3D Offset	↓ƒ 0.000 9.546 0.000
X	0.000
Y	ƒs 9.546
Z	ƒs 0.000
3D Rotation	↓ƒ 90.000 0.000 0.000
3D Pitch	ƒs 90.000
3D Roll	0.000
3D Yaw	0.000
3D Model name	NO-BN-3D-KO-SVI-SLEEPER-AND-RAILS
3D Model separation	ƒs 0.130
3D Model offset	ƒ 0.065
Local variables	
Sequence number	0
Var1	100
Var2	
Var3	
Var4	
Var5	
Metadata	
Relations	
Sections	
Formulas	
Alignment	
Alignment system	Normalspor 1435 mm
Misc	

[1] Your AutoCAD session has probably been started from a Windows shortcut of the type:
“C:\Program Files\Autodesk\AutoCAD 2019\acad.exe” /product ACAD /language “fr-FR”
(“fr-FR” means “French language, France’s version) similar. Native AutoCAD commands may have different names in your language pack, other than the COPY, COPYBASE, FIND etc that you see in our tutorial texts. In order to instruct AutoCAD to accept the native English command name, precede the native (English) command name by an underscore character, ‘_’. For instance: ‘_FIND’ will start AutoCAD’s native ‘FIND’ command even if you are using AutoCAD with the French language pack, where the command in French is called ‘RECHERCHER’. If a command needs an argument ‘ON’, and the French menu says ‘Allumer’, then you can enter ‘_ON’ to instruct AutoCAD to use the option’s native name.

1. Start AutoCAD with RailCOMPLETE, then open the ‘General Tutorials’ folder and locate the DWG file named after this tutorial. You can either type RC-ShowGeneralTutorials or you can locate the button below the RC logo in the upper left corner of your AutoCAD window.

2. RailCOMPLETE introduced **sampled formulas** with release 2019.1. The notion of sampled formula is a formula which is evaluated (sampled) when a digital cursor slides (in steps) along an alignment’s horizontal profile (the curve it describes in the XY plane). Only alignments can have sampled formulas. The alignment properties that hold sampled formulas instead of ordinary formulas are the following:

    - Offset3D.X, Offset3D.Y, Offset3D.Z
    - Rotation3D.X, Rotation3D.Y, Rotation3D.Z
    - Model3DName
    - Model3DSeparation

3. RailCOMPLETE provides a special data structure called _position for use with sampled formulas. It has two attributes:

    - _position.Pos *a real number (which we call the ‘cursor’)*
    - _position.Ref *a unique identifier referencing the alignment*

4. The sampling process starts with evaluating all formulas once (sampled and ordinary)

5. _position.Pos is set to the value of Model3DOffset, as a starting value for the ‘cursor’

6. The sampled formulas are then evaluated, using the current value of _position.Pos

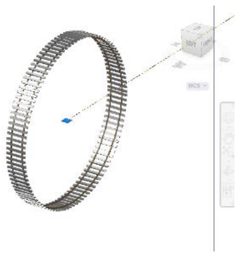
7. The 3D preview or export process uses the current values of all sampled and ordinary formulas.

8. _position.Pos is incremented with the value along the 2D alignment which corresponds to a 3D displacement with the current stepsize value of ‘Model3DSeparation’ along the alignment’s path in 3D space.

9. The three previous steps are repeated until _position.Pos reaches the end of the alignment’s geometry, which in this tutorial is referred to as ‘length2D’

10. length2D is accessible in Lua as *RcAlignment.HorizontalProfile.Length*
    length3D is accessible in Lua as *RcAlignment.Length3D*

11. As an extreme example, we have provided a **loop track example**:



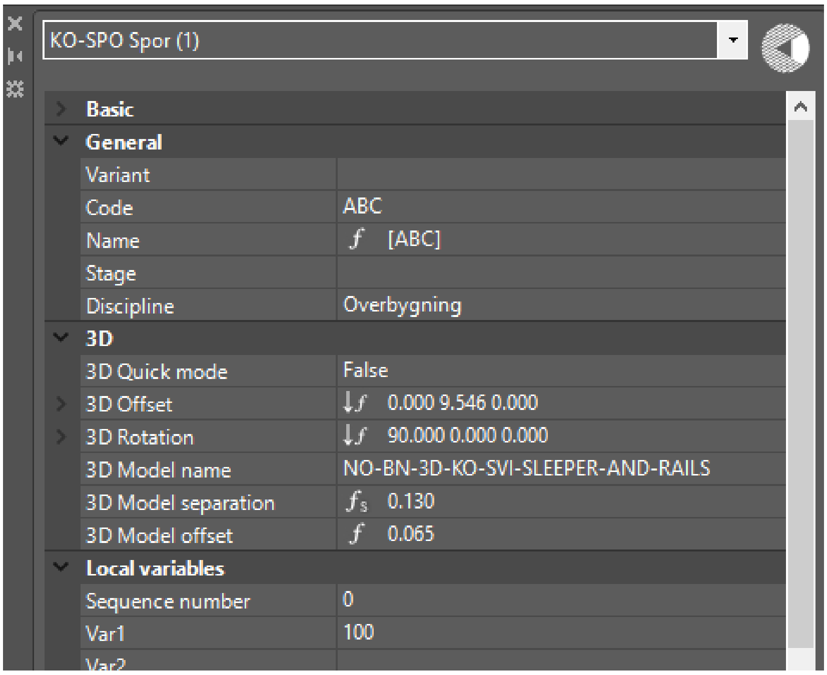
12. An alignment cannot do loops (there is just one elevation Z(s) per position ‘s’ along the geometry), but we can use the alignment as a kind of “slider” and place sleepers in 3D space as the sampled formula cursor moves along the alignment.

13. Open the Loop tutorial file.

14. Activate RC-Show3DPreview and select the short alignment stub. The loop appears.

15. Turn on the Properties manager and inspect the 3D tab's properties.

16. This is what the Properties manager should look like (click on the 3D and Local variables tabs to open them):



KO-SPO Spor (1)

Basic

General

Variant	
Code	ABC
Name	*ƒ* [ABC]
Stage	
Discipline	Overbygning

3D

3D Quick mode	False
3D Offset	↓*ƒ* 0.000 9.546 0.000
3D Rotation	↓*ƒ* 90.000 0.000 0.000
3D Model name	NO-BN-3D-KO-SVI-SLEEPER-AND-RAILS
3D Model separation	f_s 0.130
3D Model offset	*ƒ* 0.065

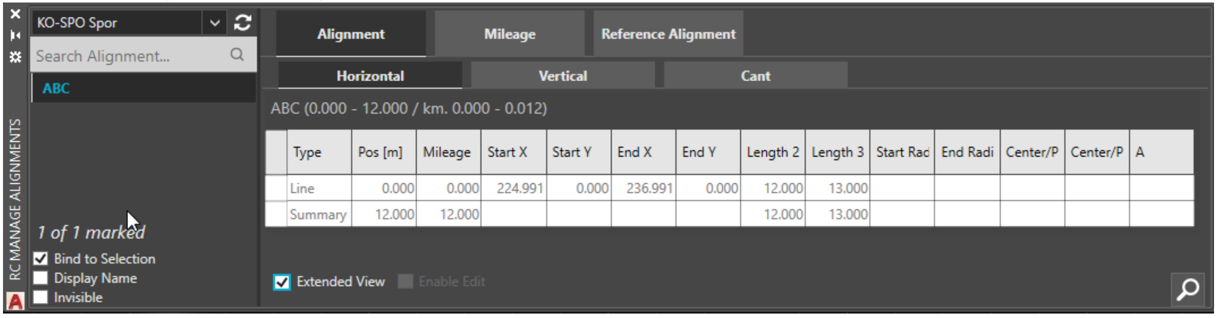
Local variables

Sequence number	0
Var1	100
Var2	

17. We have prepared a short, straight track 'ABC' for you, with 2D length 12 meters. It starts at altitude Z=5 and climbs to altitude Z=10 meters above sea level (a steep incline for a railway...). The 3D length is sqrt(12*12 + 5+5) = 13 meters. It has no cant data. It uses the 3D object model NO-BN-3D-KO-SVI-SLEEPER-AND-RAILS to express itself in 3D, a simple object model featuring two 60 cm long stubs of rail and a sleeper.

18. This is what the Alignment Manager should look like:

Horizontal alignment datagrid:



KO-SPO Spor

Search Alignment...

ABC

1 of 1 marked

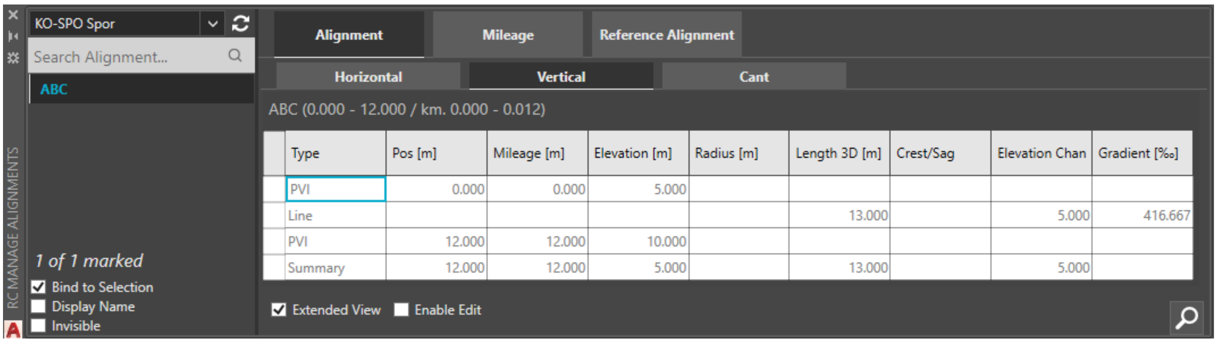
☑ Bind to Selection
□ Display Name
□ Invisible

ABC (0.000 - 12.000 / km. 0.000 - 0.012)

Type	Pos [m]	Mileage	Start X	Start Y	End X	End Y	Length 2	Length 3	Start Rad	End Radi	Center/P	Center/P	A
Line	0.000	0.000	224.991	0.000	236.991	0.000	12.000	13.000					
Summary	12.000	12.000					12.000	13.000					

☑ Extended View □ Enable Edit

Vertical alignment datagrid:



1 of 1 marked

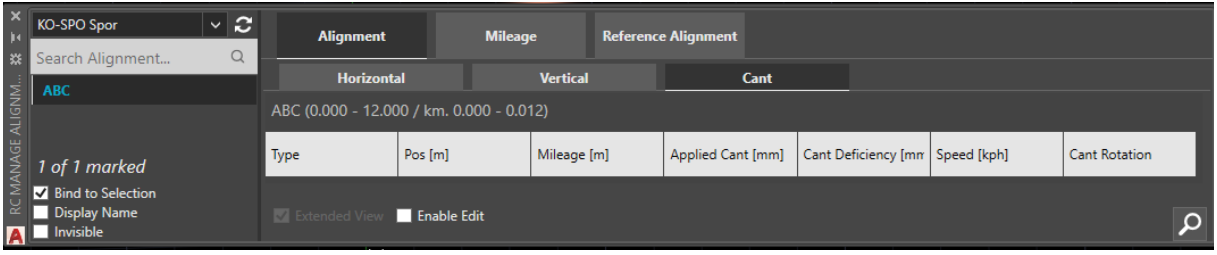
☑ Bind to Selection
□ Display Name
□ Invisible

ABC (0.000 - 12.000 / km. 0.000 - 0.012)

Type	Pos [m]	Mileage [m]	Elevation [m]	Radius [m]	Length 3D [m]	Crest/Sag	Elevation Chan	Gradient [‰]
PVI	0.000	0.000	5.000					
Line					13.000		5.000	416.667
PVI	12.000	12.000	10.000					
Summary	12.000	12.000	5.000		13.000		5.000	

☑ Extended View □ Enable Edit

Cant datagrid (no data):



1 of 1 marked

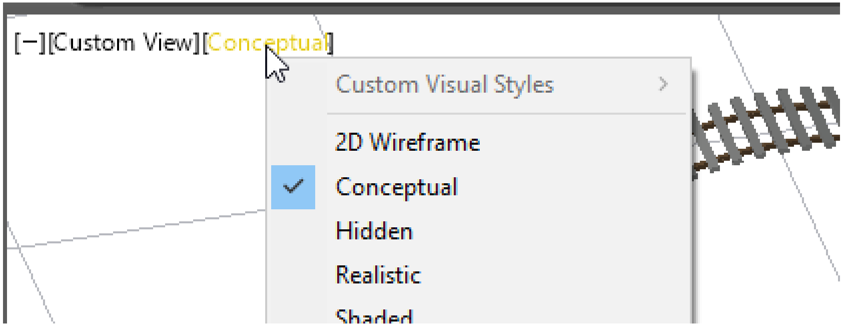
☑ Bind to Selection
□ Display Name
□ Invisible

ABC (0.000 - 12.000 / km. 0.000 - 0.012)

Type	Pos [m]	Mileage [m]	Applied Cant [mm]	Cant Deficiency [mm	Speed [kph]	Cant Rotation

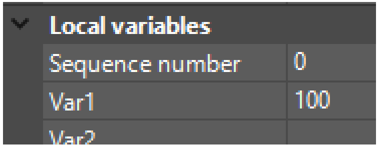
☑ Extended View □ Enable Edit

19. Change to Conceptual or Realistic mode if you want to see solids instead of wireframe objects.



20. Open RC-ManageProperties and ensure that "3D Quick mode" in the 3D tab is set to ‘False’.

21. **Loop size.** Open the 'Local variables' tab. We have chosen to let Var1 represent the number of sleepers in the loop track, initially set to Var1 = 100.



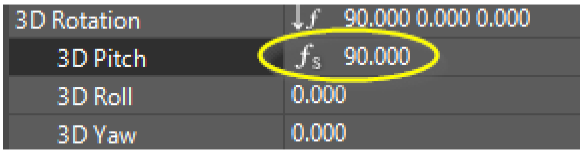
Local variables

Sequence number	0
Var1	100
Var2	

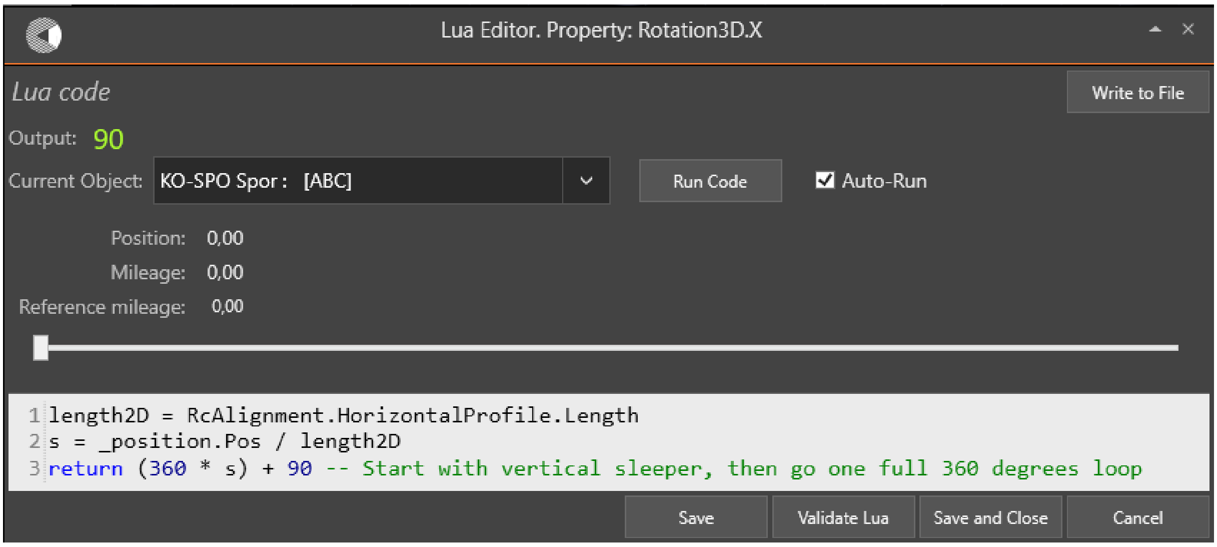
22. With some trigonometry, you will find that a polygon of N sides, each w long, has an inscribed circle with radius r = w / (2*tan(pi/N)). The loop track is inscribed in a polygon consisting of N=Var1=100 rail stubs, each w=0.6 meters long, arranged in a perfect 100-sided polygon.

23. In the '3D' tab, open the sub-tab named 'Rotation 3D'. With F3, inspect the three formulas for Pitch, Roll and Yaw. We have entered (360*s + 90) as pitch ('gradient'), where s runs from 0 to 1 as the formula is sampled along ABC. The other two formulas are void (they contain a simple value of 0). Adding 90 degrees makes the first sleeper vertical.

3D Pitch (Rotation3D.X)



3D Rotation	90.000 0.000 0.000
3D Pitch	90.000
3D Roll	0.000
3D Yaw	0.000



Output: 90

Current Object: KO-SPO Spor : [ABC]

☑ Auto-Run

Position: 0,00

Mileage: 0,00

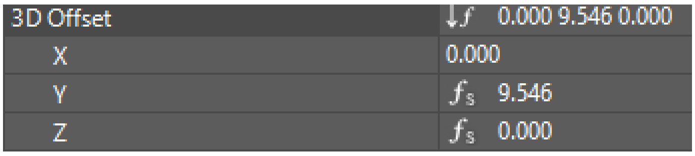
Reference mileage: 0,00

```
length2D = RcAlignment.HorizontalProfile.Length
s = _position.Pos / length2D
return (360 * s) + 90 -- Start with vertical sleeper, then go one full 360 degrees loop
```

24. Open the sub-tab named '3D Offset'. You will find offset formulas for X, Y and Z.

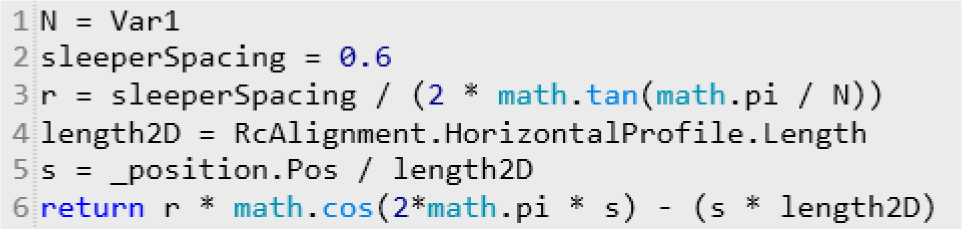
    X is sideways motion
    Y is motion along ABC in the XY-plane
    Z is vertical motion.

    We set X(s) to zero, and we set {Y(s), Z(s)} to {r*cos(2*pi*s), r*sin(2*pi*s)}.



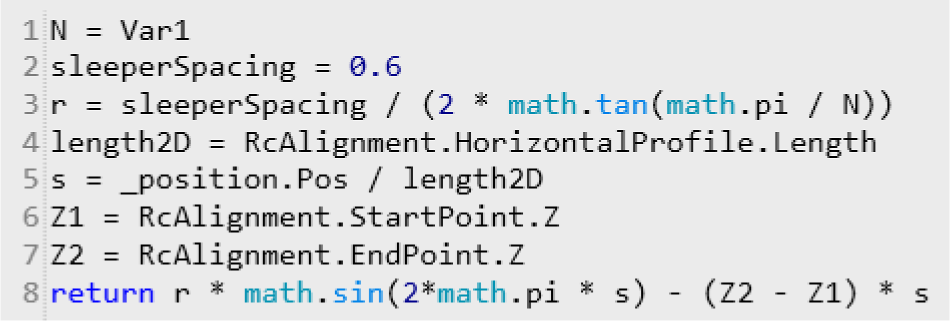
3D Offset	0.000 9.546 0.000
X	0.000
Y	9.546
Z	0.000

Longitudinal motion (Offset3D.Y):



```
N = Var1
sleeperSpacing = 0.6
r = sleeperSpacing / (2 * math.tan(math.pi / N))
length2D = RcAlignment.HorizontalProfile.Length
s = _position.Pos / length2D
return r * math.cos(2*math.pi * s) - (s * length2D)
```

Vertical motion (Offset3D.Z):



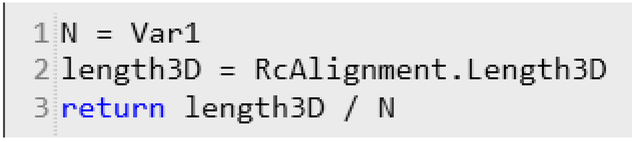
```
N = Var1
sleeperSpacing = 0.6
r = sleeperSpacing / (2 * math.tan(math.pi / N))
length2D = RcAlignment.HorizontalProfile.Length
s = _position.Pos / length2D
Z1 = RcAlignment.StartPoint.Z
Z2 = RcAlignment.EndPoint.Z
return r * math.sin(2*math.pi * s) - (Z2 - Z1) * s
```

25. As you might have noticed already, there are correction terms to the above-mentioned formulas for Y and Z. Since {X,Y,Z} represents a 3D point moving along ABC's 3D curved line in space as s in our tutorial goes from 0 to 1, we will have to subtract the sleepers' 3D insertion position at every sampled point if we shall get a loop. Therefore we subtract length2D*s from Y and (Z2-Z1)*s from Z at every sampled point along ABC. Try to remove the correction terms and see what happens.

26. You could also access the underlying Z coordinate directly: **getAlignmentInfo**(_position).Point.Z

27. Locate the 3D property '3D Model separation' and open it with F3. In order to obtain exactly N sleepers around the loop, we must sample ABC exactly N times. The {X,Y,Z} 3D-cursor slides along ABC's curve in 3D space as s runs from 0 to 1. The step size is therefore the 3D length of ABC divided by the number of steps we need.

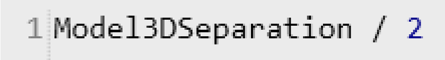
3D Model separation (Model3DSeparation)



```
N = Var1
length3D = RcAlignment.Length3D
return length3D / N
```

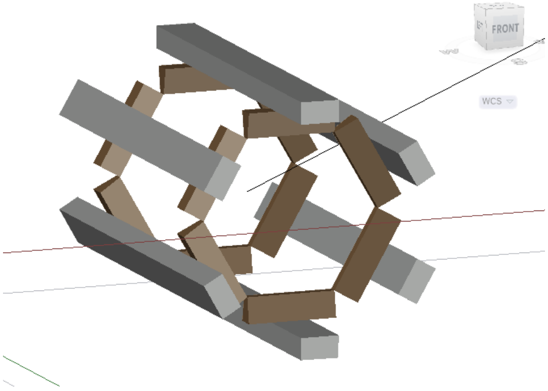
28. Add an initial Model3Doffset to s, the starting location for the sampling process, just to avoid that you get a sleeper at s=0 and one at s=1 (a total of N+1). Choosing an initial offset of Model3DSeparation/2 is always a safe bet.

3D Model offset (Model3Doffset)

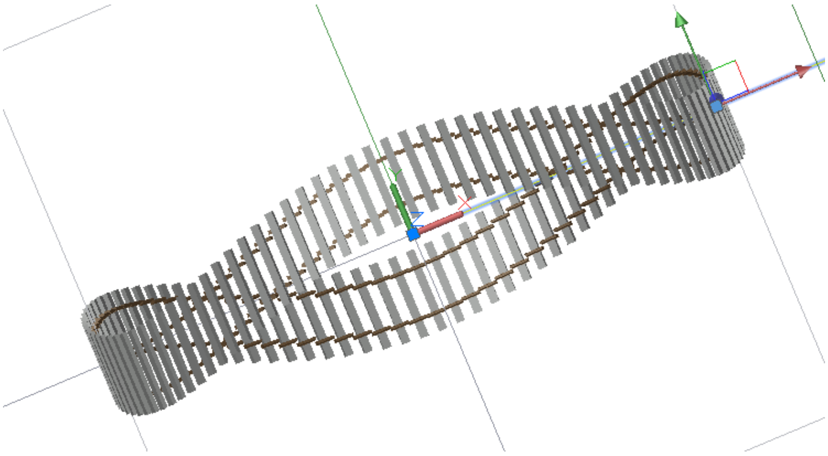


```
Model3DSeparation / 2
```

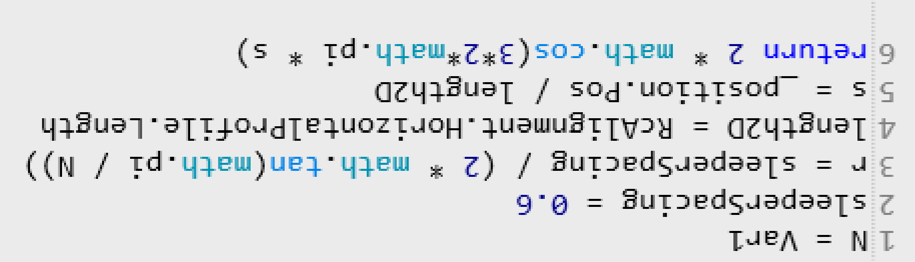
29. Try changing the number of sleepers from 100 to some other number. N=6 is shown below.



30. **Challenge**: Try adding a sideways sine curve to Offset3D.X. Try to figure out yourself how to make it wiggle sideways 3 times per loop, with a maximum sideways offset of 2 meters.



Suggestion:



```
N = Var1
sleeperSpacing = 0.6
r = sleeperSpacing / (2 * math.tan(math.pi / N))
length2D = RcAlignment.HorizontalProfile.Length
s = _position.Pos / length2D
return 2 * math.cos(3*2*math.pi * s)
```

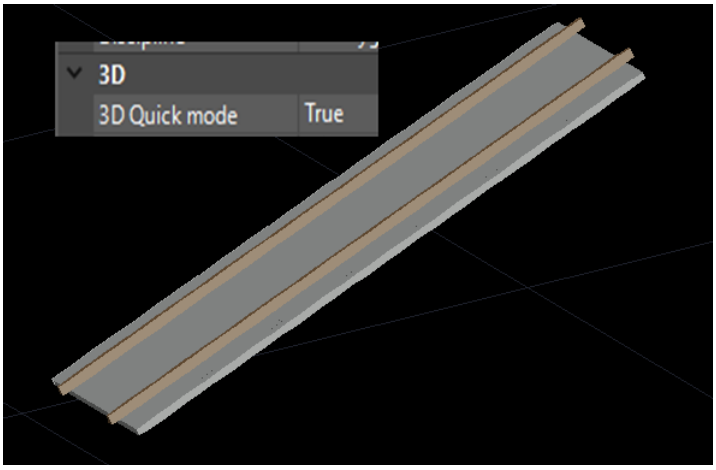
31. Instead of adding a sideways motion to Offset3D.X, you can merely adjust ABC's geometry sideways.

32. Try stretching ABC in length or modifying it any way. Try adding polyline grips to the ABC polyline in modelspace and move them sideways. Try converting line segments to arcs. To modify a grip, hover over the polyline's grip and select 'Add vertex', 'Remove vertex', 'Convert to line', 'Convert to Arc'.

    NOTE: When you stretch the geometry just by moving the polyline grips in modelspace, you might extend it beyond the extents of the vertical profile data that are stored in the alignment. The results may be unexpected, depending on how well your Lua code deals with missing alignment data. Open the RC-ManageAlignments editor and check the contents of horizontal and vertical data.

33. Try using the RC-ManageAlignments editor to modify your loop. The vertical profile data must be at least as long as ABC’s geometry in order for the example formulas to work well.

34. Toggle 3D Quick mode between True and False. In 3D Quick mode, RailCOMPLETE uses standard built-in formulas for Offset3D and Rotation3D, suitable for dealing with railway tracks.

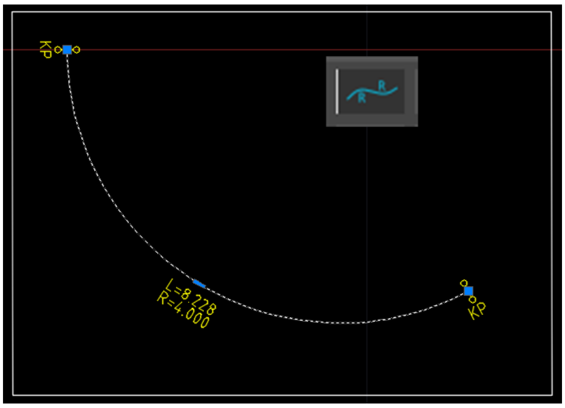


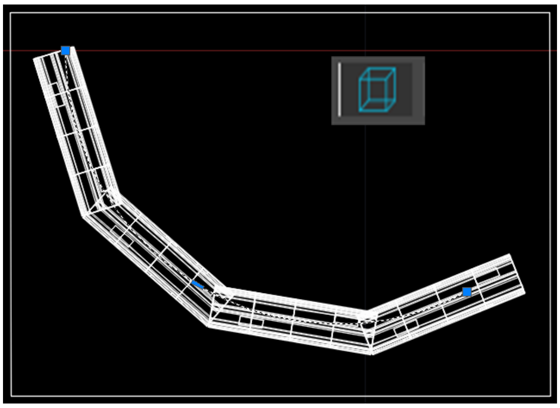
3D

3D Quick mode True

35. A more reasonable use of sampled formulas is to vary the Model3DName attribute as the cursor slides along the alignment. For instance, assume that you have a railway track with an associated 3D object model for a sleeper and two stubs of rail. You can make a formula for Model3DName that checks if the sampled function cursor is inside a switch and then suppress expressing sleepers. Or you might instead ask the switch for its construction length and add this to the Model3DSeparation stepsize to make one long jump. The switch object itself can then be expressed in its full detail as an ordinary 3D object. Also, when the sampled formula cursor moves through the switch rear part, the sleeper type can be varied from short upright 1:60 sleepers to normal length sleepers with 1:20 inclined tracks.

36. The NO-BN cable duct alignment object uses sampled formulas to insert concrete elements more often when the alignment radius is low, to account for the necessary cutting of neighbor elements to make them fit nicely:





37. We really hope that you have developed a sense of understanding for the Sampled Formula concept by now, and that you are ready to apply your knowledge to your own realistic projects.

38. Have fun!

Please check our website www.railcomplete.com for updates.

Corrections and suggestions are welcome to support@railcomplete.no.

Thank you for using RC Tutorials!