

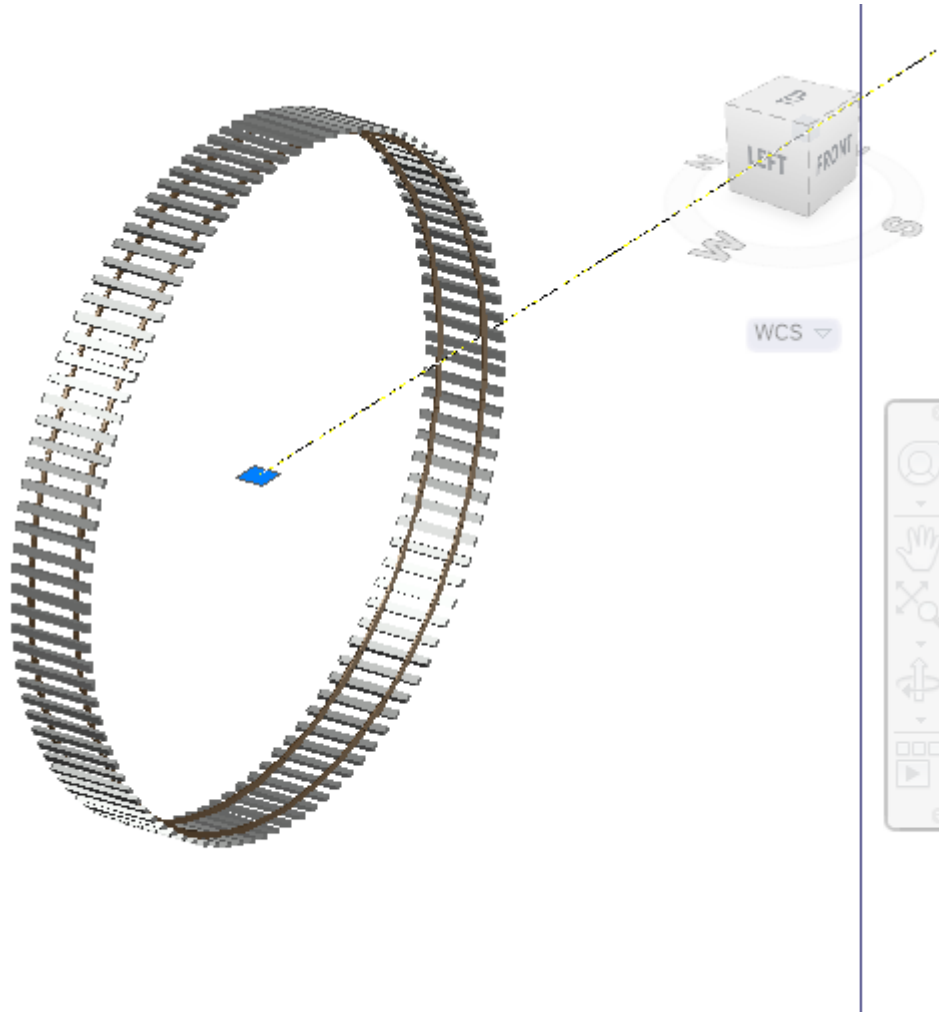
RC tutorial 102 - Sampled formulas - Loop track

=====

Revised 2019-09-20.


- This tutorial was prepared with RC version 2018.22.1226 with DNA version NO-BN;NO-0001;2019-09-18T22:38:00+01:00;2019.1.
 - Assumed skills: Basic Lua programming, alignments and the RC alignment manager.
 - Time to spend here: Intermediate: 30 minutes.
 - Further reading: None.
1. This tutorial's goal is to teach you advanced use of sampled formulas. Making a railway loop might not be useful, but it really brings you into the intended RailCOMPLETE concepts.
 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 two 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**:



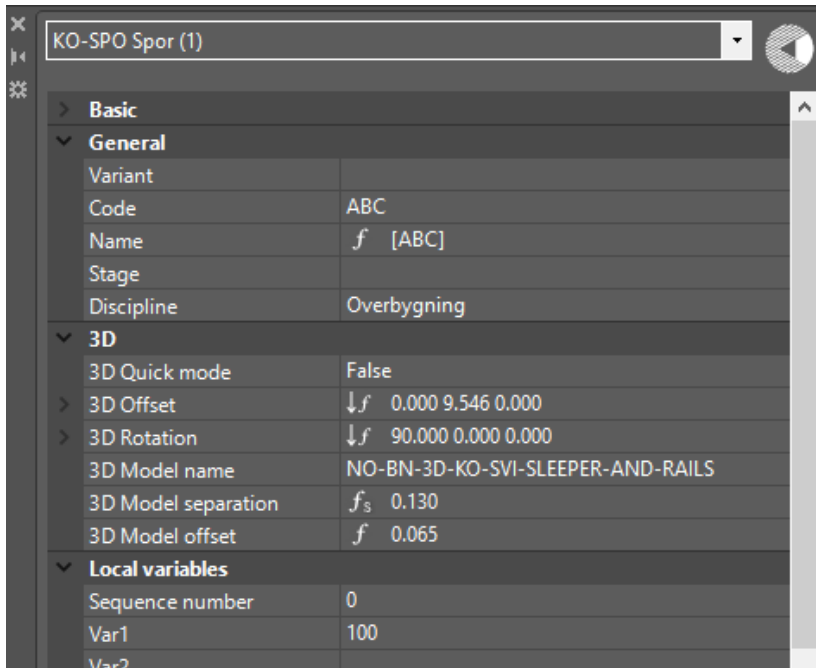
12. An alignment can of course not do loops, but we can use the alignment as a kind of “slider” and place sleepers in 3D space as the sampled formula cursor moves along the alignment.

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: f [ABC]

Stage

Discipline: Overbygning

3D

3D Quick mode: False

3D Offset: ↓f 0.000 9.546 0.000

3D Rotation: ↓f 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: f 0.065

Local variables

Sequence number: 0

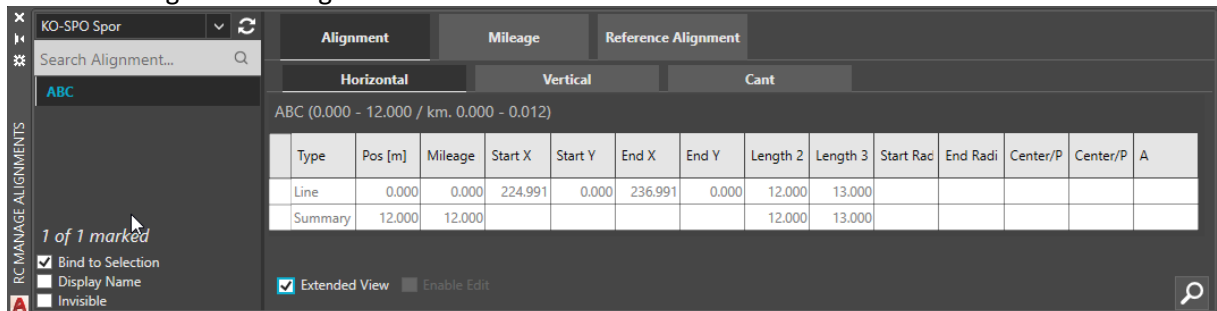
Var1: 100

Var2:

17. We have prepared a short, straight track 'ABC' for you, with 2D length 12 meters. It starts at altitude 5 and climbs to altitude 10 meters above sea level (a steep incline for a railway...). The 3D length is $\sqrt{12^2 + 5^2} = 13$ meters. It has no cant data. It uses NO-BN-3D-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

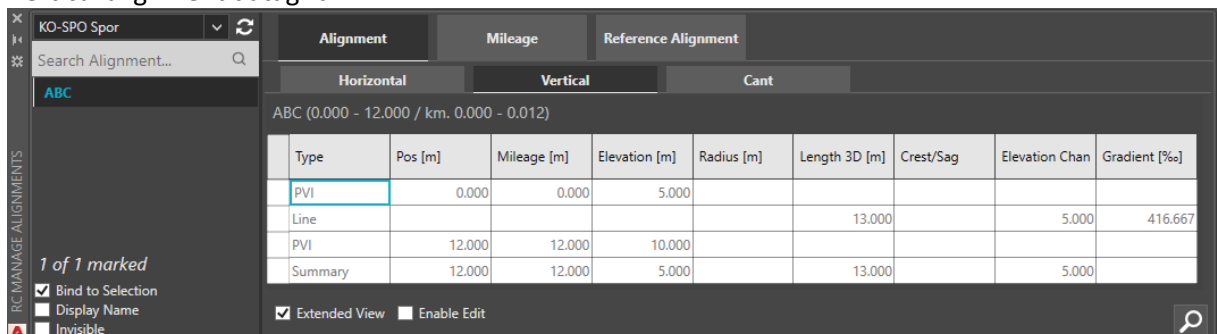
☐ Display Name

☐ Invisible

☒ Extended View ☐ Enable Edit

Type	Pos [m]	Mileage	Start X	Start Y	End X	End Y	Length 2	Length 3	Start Rac	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					

Vertical alignment datagrid:



KO-SPO Spor

Search Alignment...

ABC

1 of 1 marked

☒ Bind to Selection

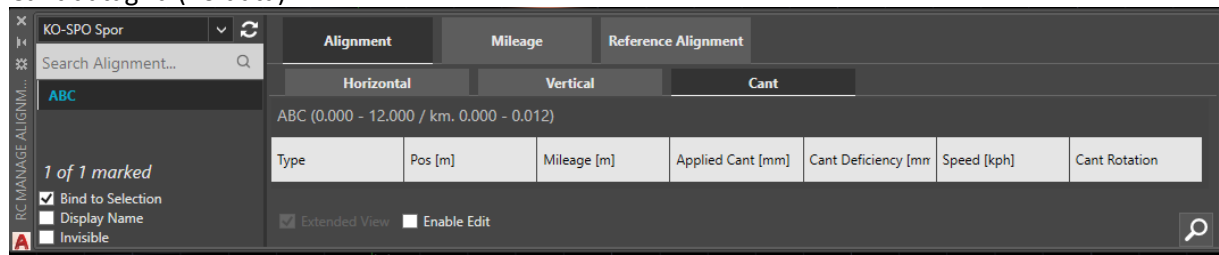
☐ Display Name

☐ Invisible

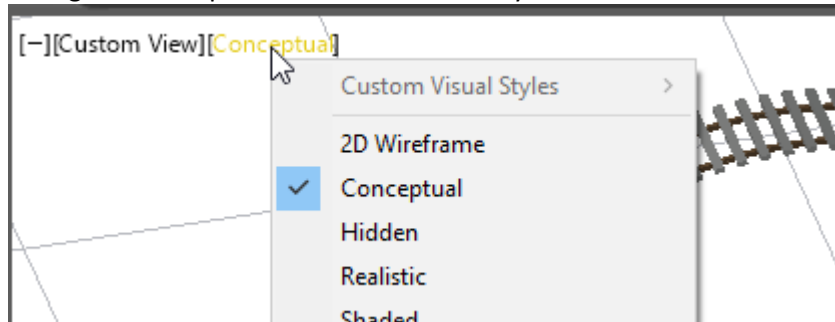
☒ Extended View ☐ Enable Edit

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	

Cant datagrid (no data):



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.

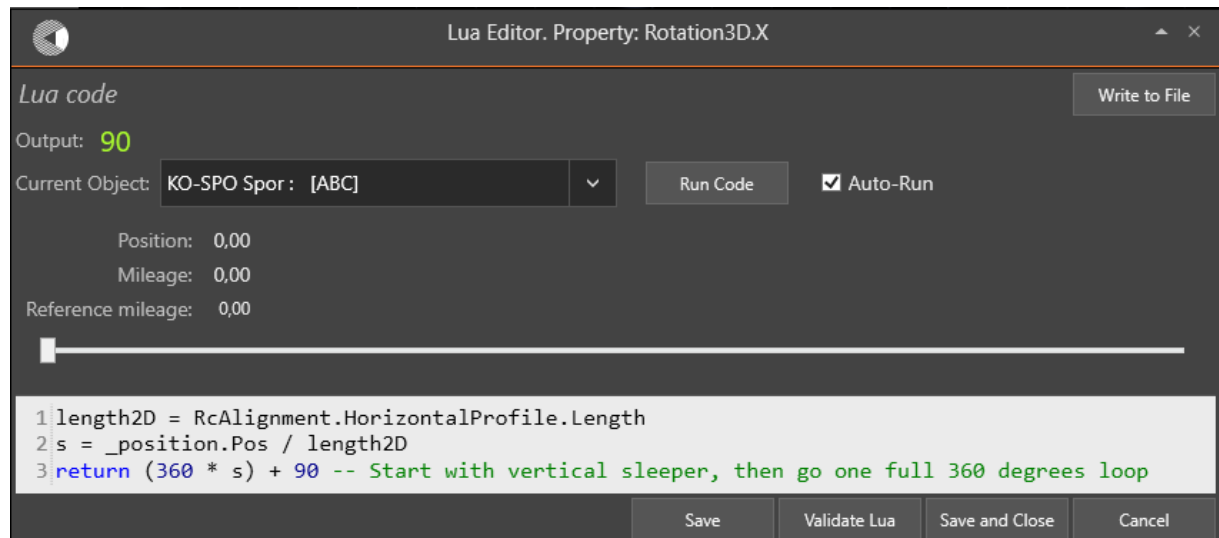
Var1	100
------	-----

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	f_s	90.000	0.000	0.000
3D Pitch	f_s	90.000		
3D Roll		0.000		
3D Yaw		0.000		



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, and Z is vertical motion. We set X to zero, and (Y, Z) to $(r \cdot \cos(2 \cdot \pi \cdot s), r \cdot \sin(2 \cdot \pi \cdot s))$.

3D Offset	$\downarrow f_s$ 0.000 9.546 0.000
X	0.000
Y	f_s 9.546
Z	f_s 0.000

Longitudinal motion (Offset3D.Y):

```

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

```

Vertical motion (Offset3D.Z):

```

1 N = Var1
2 sleeperSpacing = 0.6
3 r = sleeperSpacing / (2 * math.tan(math.pi / N))
4 length2D = RcAlignment.HorizontalProfile.Length
5 s = _position.Pos / length2D
6 Z1 = RcAlignment.StartPoint.Z
7 Z2 = RcAlignment.EndPoint.Z
8 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 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 $\text{length2D} \cdot s$ from Y and $(Z2 - Z1) \cdot s$ from Z at every sampled point along ABC. Try to remove the correction terms and see what happens.

26. 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.

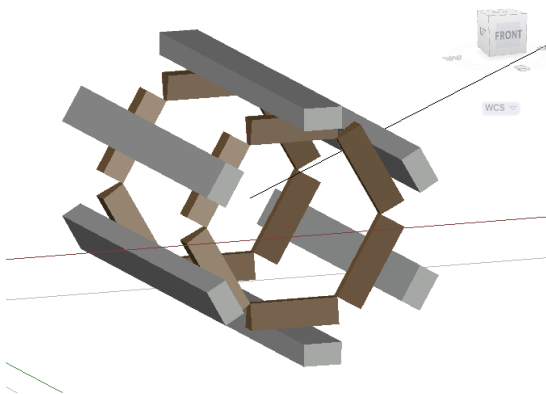
```
1 N = Var1
2 length3D = RcAlignment.Length3D
3 return length3D / N
```

27. 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.

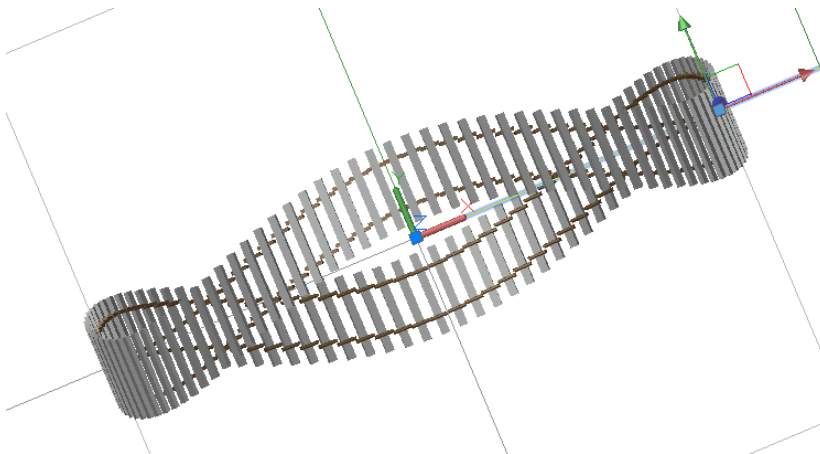
3D Model offset (Model3DOffset)

```
1 Model3DSeparation / 2
```

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



29. **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:

```

1 N = Var1
2 sleeperSpacing = 0.6
3 r = sleeperSpacing / (2 * math.tan(math.pi / N))
4 length2D = Rcalignment.HorizontalProfile.Length
5 s = _position.Pos / length2D
6 return 2 * math.cos(3*2*math.pi * s)

```

30. Instead of adding a sideways motion to Offset3D.X, you can merely adjust ABC's geometry sideways.
31. 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'.
32. 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.
33. 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.
34. A 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.
35. 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.
36. 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!