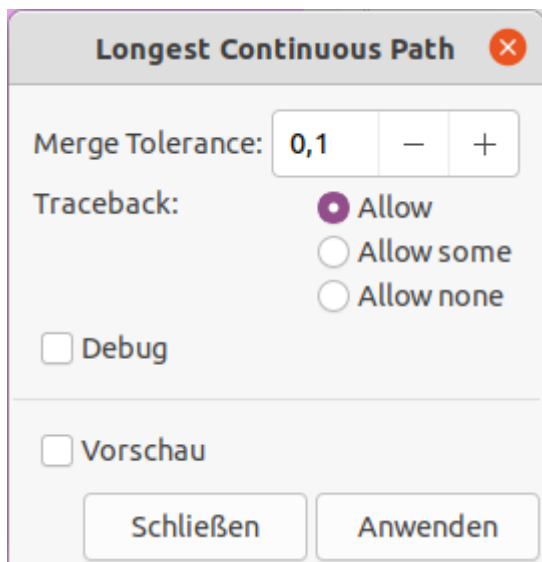


Longest Continuous Path

Source of documentation: <https://github.com/Daekkyn/inkscapeOptimizePath>

An Inkscape extension that tries to make the longest continuous paths. It converts the paths to a graph, converts the graph to an Eulerian graph and finds an Eulerian cycle. It was written for the [AxiDraw](#) pen plotter to reduce writing time of graph-like drawings like Voronoi diagrams and meshes.

https://www.youtube.com/embed/ZuaCT3Qi_-c



Usage

1. Make sure to ungroup everything → Extensions → Reorder → Deep Ungroup
2. Make sure that the paths use only absolute coordinates (see trick below) or use → Extensions → Modify Path → [To Absolute](#)
3. Make sure to not have transforms on the paths. You can use the [Apply Transformations](#) extension to remove them
4. Select all the paths you want to optimize (currently only works with polylines). Conversion to polylines can be done with [Approximate Curves by Straight Lines \(Flatten Beziars\)](#) or [Convert To Polylines](#) (as second choice). Bezier curves, which were not converted, are getting **lost** running this extension
5. Open the extension

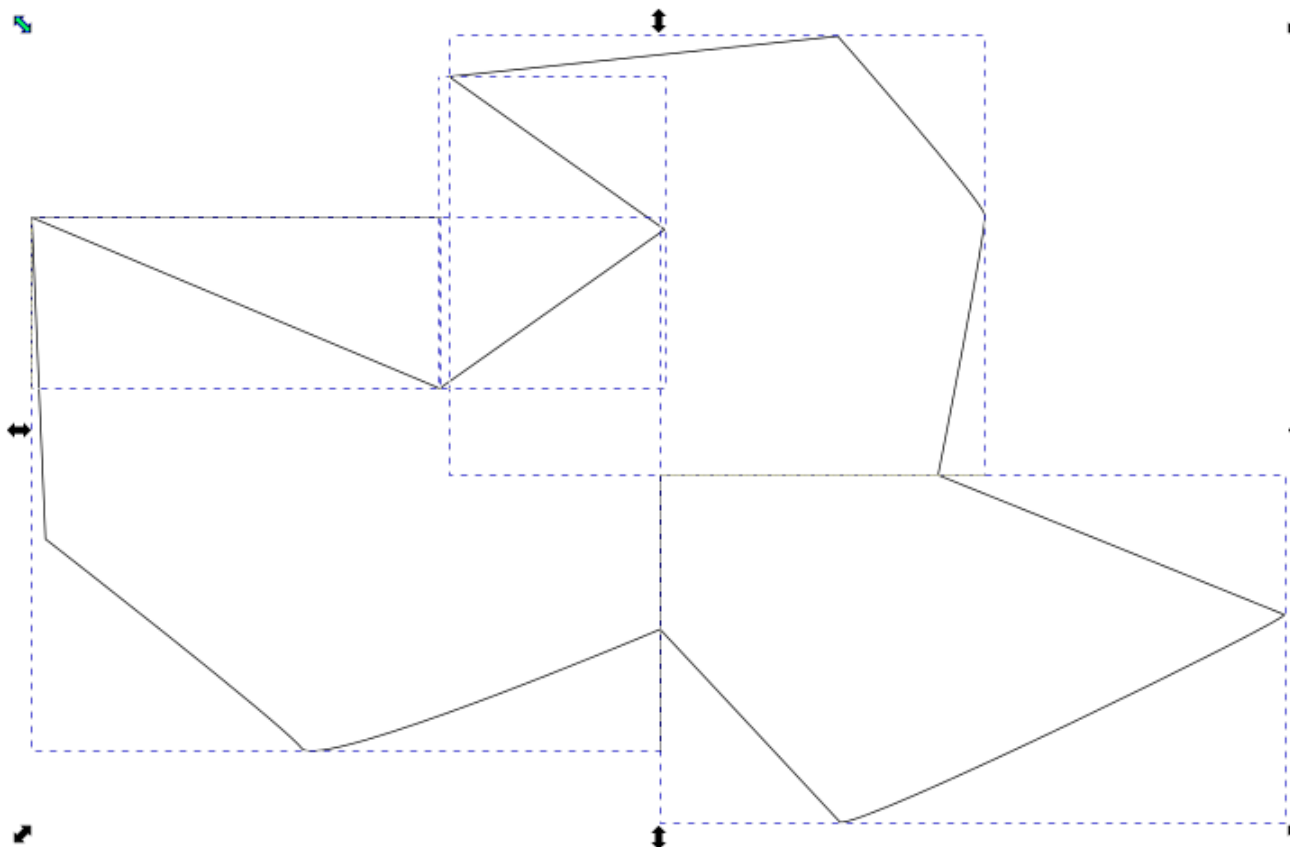
6. Set the merge tolerance (0.1 should work in most cases)
7. Choose the Overwrite rule:
 1. "Allow" means that the result will be a single path which might (probably) will have some overlapping edges.
 2. "Allow none" means that the results will be multiple disconnected paths but there will be no overlapping edges.
 3. "Allow some" is an in-between, overlapping edges are allowed, but only in short numbers. This is probably the best choice in most cases.
8. Apply

Trick to remove relative coordinates

1. Change the preferences for 'SVG Output > Path Data' to always use absolute coordinates. This will only affect newly created paths, or existing objects for which a rewrite of the path data is triggered.
2. For existing paths, use 'Edit > Select All in All Layers', and nudge the selection with the arrow keys (e.g. one step up and one back down again). This will trigger a rewrite of the path data in 'd' which will follow the changed preferences for optimized path data.

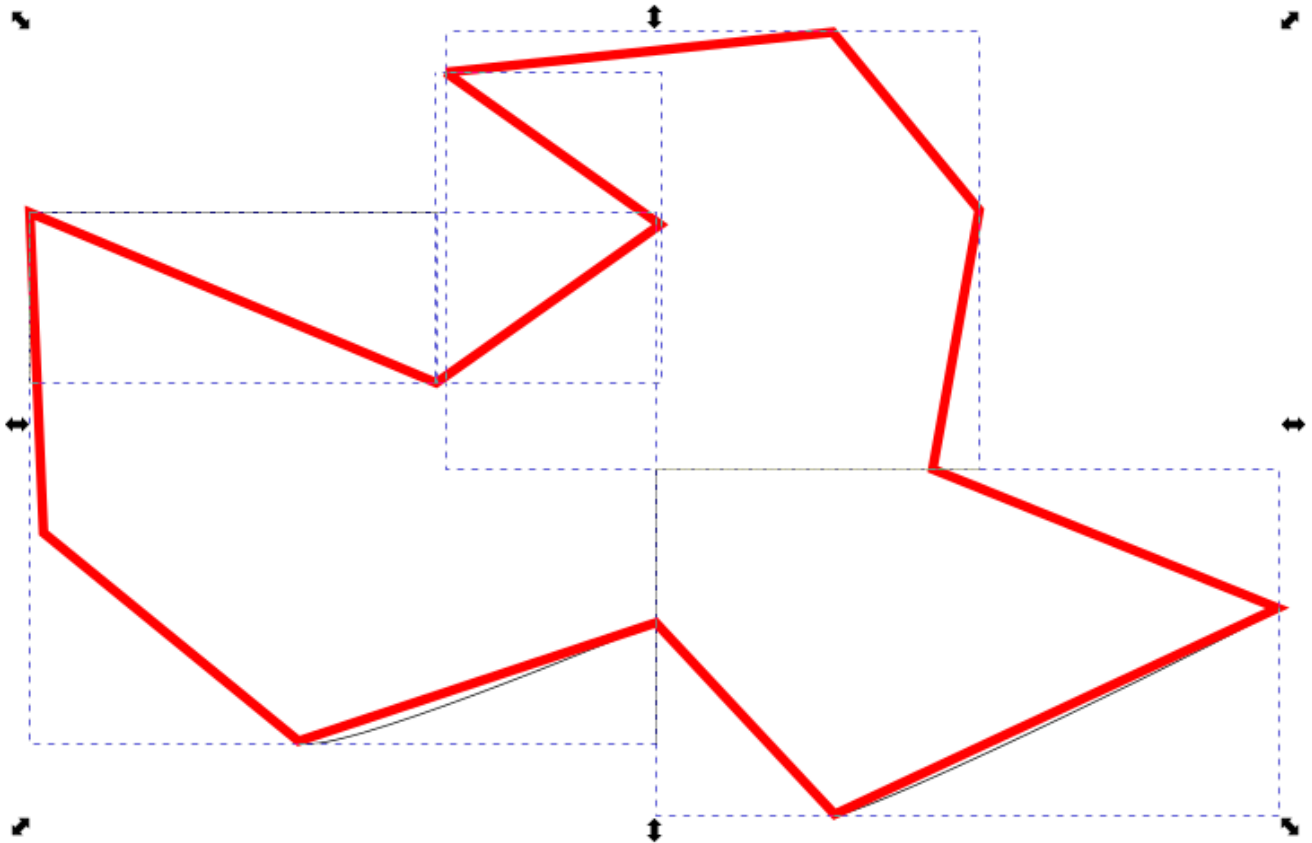
Another Example

1. **Draw some paths and put the ends together by drag and drop with snap**

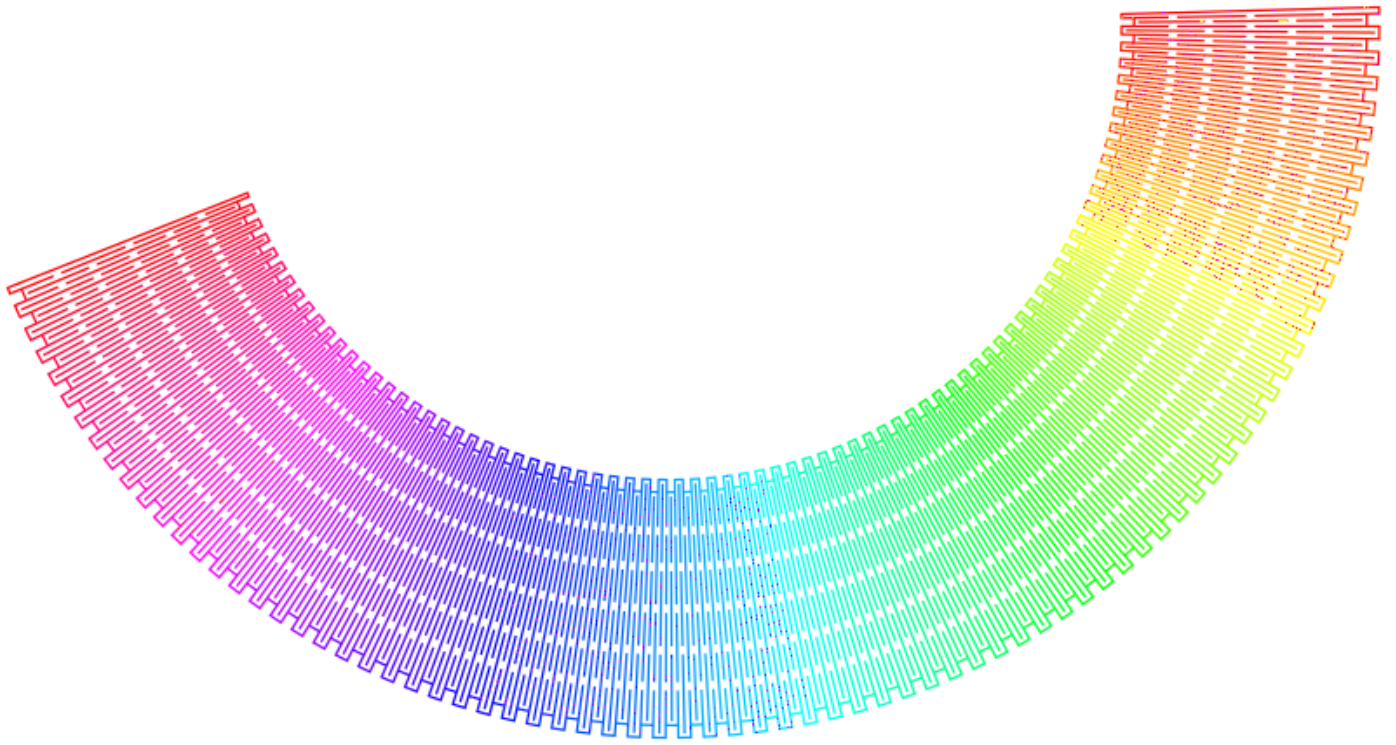


2. Get the result

The red contour is an optimized contour. It works similar like [Close Paths](#), [Chain Paths](#) or [Join Paths / Create Tabs And Dimples](#) except that it will not handle bezier curves but only polylines. So it acts like a combination of those plugins in junction with [Approximate Curves by Straight Lines \(Flatten Beziers\)](#) or [Convert To Polylines](#).



This extension can also create really cool artwork



Version #4

Erstellt: 12 Februar 2025 01:52:09 von Mario Voigt

Zuletzt aktualisiert: 30 Mai 2025 11:29:37 von Mario Voigt