Parametric Reconstruction of Bent Tube Surfaces

Ulrich Bauer

FU Berlin

joint work with Konrad Polthier

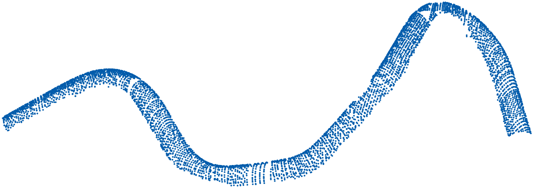
DFG Research Center MATHEON
Mathematics for key technologies

Industry Challenges in Geometric Modeling, CAD and Simulation, 14.3.2008

A problem from industry...

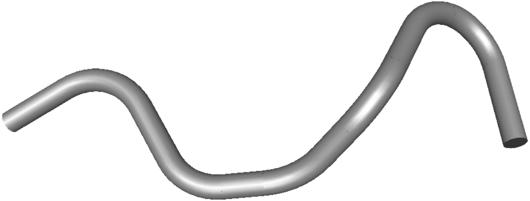
- Given: a bent metal tube

A problem from industry...



- Given: a bent metal tube
- Input: a point cloud of the tube surface (from laser scanner)

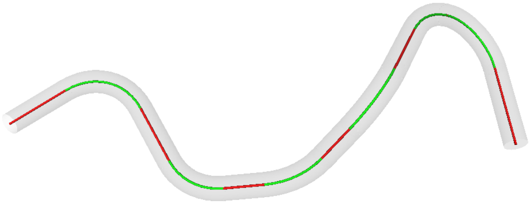
A problem from industry...



- ▷ Given: a bent metal tube
- ▷ Input: a point cloud of the tube surface (from laser scanner)
- ▷ Wanted: a parametric description of the tube surface

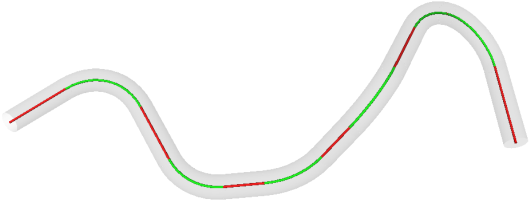
Problem description

A problem from industry. . .



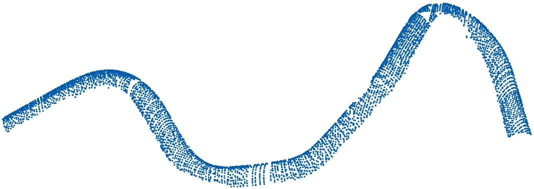
- Given: a bent metal tube
- Input: a point cloud of the tube surface (from laser scanner)
- Wanted: a parametric description of the tube surface
- Surface consists of G^1 continuous cylinder and torus segments

A problem from industry...

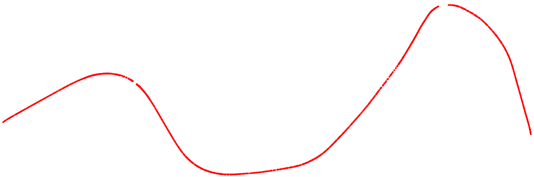


- Given: a bent metal tube
- Input: a point cloud of the tube surface (from laser scanner)
- Wanted: a parametric description of the tube surface
- Surface consists of G^1 continuous cylinder and torus segments
- *Pipe surface* (envelope of a ball moving along the *spine curve*)

1. Reconstruction of bent tubes
2. Overview of the algorithm
   - Spine curve computation
   - Curve reconstruction
   - Spine curve segmentation
   - Least squares distance optimization
3. Results
   - Performance
   - Examples

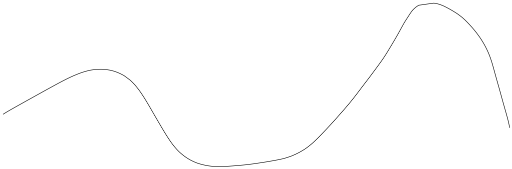


Decompose into subproblems:



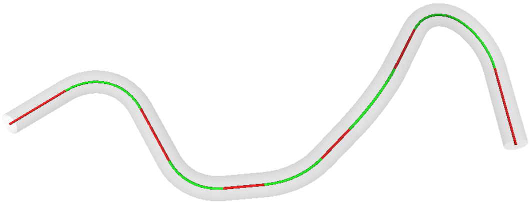
Decompose into subproblems:

- Project the surface points onto the spine curve



Decompose into subproblems:

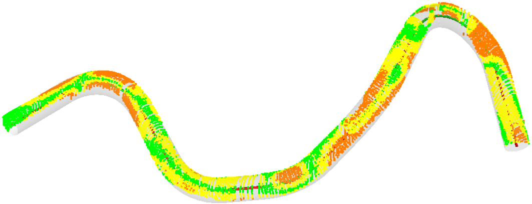
- Project the surface points onto the spine curve
- Join and simplify the spine curve points



Decompose into subproblems:

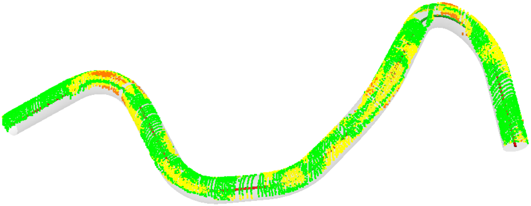
- ▷ Project the surface points onto the spine curve
- ▷ Join and simplify the spine curve points
- ▷ Approximate the curve by G^1 continuous arcs and line segments

Overview of the algorithm



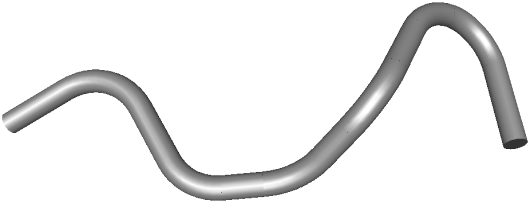
Decompose into subproblems:

- ▷ Project the surface points onto the spine curve
- ▷ Join and simplify the spine curve points
- ▷ Approximate the curve by G^1 continuous arcs and line segments
- ▷ Fit the surface to the input samples



Decompose into subproblems:

- Project the surface points onto the spine curve
- Join and simplify the spine curve points
- Approximate the curve by G^1 continuous arcs and line segments
- Fit the surface to the input samples



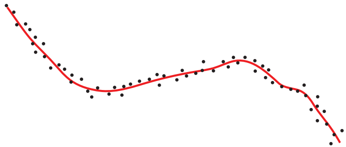
Decompose into subproblems:

- Project the surface points onto the spine curve
- Join and simplify the spine curve points
- Approximate the curve by G^1 continuous arcs and line segments
- Fit the surface to the input samples

Result: accurate parametric reconstruction of the tube surface

Outline

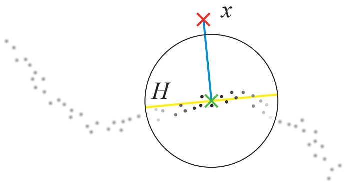
To find the spine curve, we adapt the moving least squares method.



MLS curve (D. Levin, 1998)

Define a curve from a set of points

To find the spine curve, we adapt the moving least squares method.



MLS curve (D. Levin, 1998)

Define a curve from a set of points

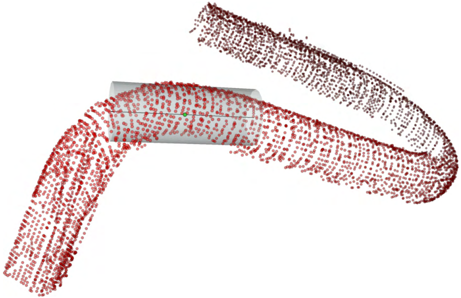
Let x be any point in the plane.

- Locally biased least squares fitting of a hyperplane H
- Center of bias moves with the projection of point x onto H
- x lies on optimal $H \Leftrightarrow x$ is a point on the MLS curve

Idea: extend MLS projection to other primitives than hyperplanes

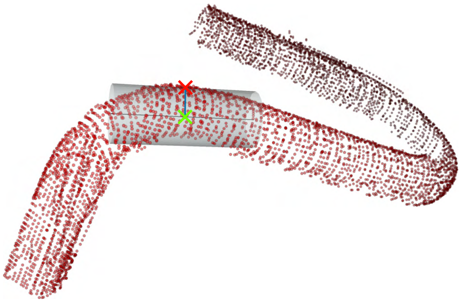
Idea: extend MLS projection to other primitives than hyperplanes

- ▷ Locally fit cylinders to surface samples



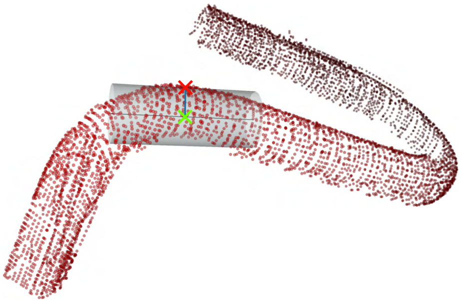
Idea: extend MLS projection to other primitives than hyperplanes

- Locally fit cylinders to surface samples
- Project samples onto axis of cylinders

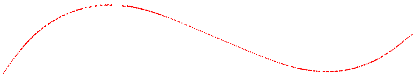


Idea: extend MLS projection to other primitives than hyperplanes

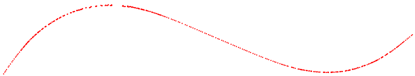
- ▷ Locally fit cylinders to surface samples
- ▷ Project samples onto axis of cylinders
- ▷ Goal: approximate spine curve of pipe surface



Comparison to previous work



Our method



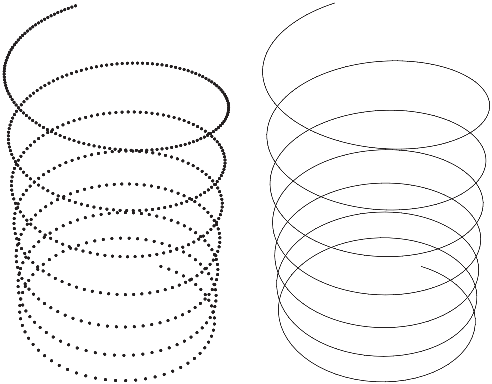
Our method

In-Kwon Lee, 2000

- Previous work: only use estimated normals and radius to shift samples
- Additional smoothing required

Outline

Curve reconstruction: well-investigated problem



We used the NN-Crust algorithm (Dey & Kumar, 1999)
Simplify polygonal curve for faster processing (Daescu, 2003)

Outline

Find a G^1 continuous curve of arc and line segments *(arc-line spline)*

- with distance $< \epsilon$ to the vertices of the input polygon
- with minimum number of segments

This problem is similar to a simpler problem...

Problem: Given a polygonal curve P.
Find another polygonal curve P' with distance $d(P, P') < \epsilon$
with minimum number of segments.

Problem: Given a polygonal curve P.
Find another polygonal curve P' with distance $d(P, P') < \epsilon$ with minimum number of segments.

Framework (H. Imai, M. Iri, 1986) used by many algorithms:

- Build a shortcut graph over vertices of polygon
- An edge e_{ij} is in the graph if the line segment $\overline{p_i p_j}$ approximates $[p_i, p_{i+1}, \ldots, p_j]$
- Find a shortest path through the shortcut graph
- $\mathcal{O}(n^3)$ time, $\mathcal{O}(n)$ space (don't construct graph explicitly)

Problem: Given a polygonal curve P.
Find another polygonal curve P' with distance $d(P, P') < \epsilon$ with minimum number of segments.

Framework (H. Imai, M. Iri, 1986) used by many algorithms:

- Build a shortcut graph over vertices of polygon
- An edge e_{ij} is in the graph if the line segment $\overline{p_i p_j}$ approximates $[p_i, p_{i+1}, \ldots, p_j]$
- Find a shortest path through the shortcut graph
- $\mathcal{O}(n^3)$ time, $\mathcal{O}(n)$ space (don't construct graph explicitly)

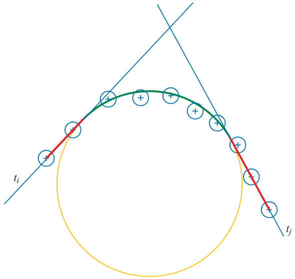
Restriction to vertices of input polygon
For unrestricted vertex positions in $\mathbb{R}^3$, no algorithm known

Arc-line spline simplification

Try something similar for arc-line splines

- Restrict solution set to something we can handle by a graph
- Find optimal solution for the restricted set using BFS

Estimated tangent lines as vertices of graph



Arc-line spline simplification

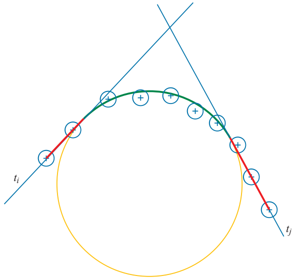
Try something similar for arc-line splines

- Restrict solution set to something we can handle by a graph
- Find optimal solution for the restricted set using BFS

Estimated tangent lines as vertices of graph

Problem: tangent lines are in general not coplanar

- To compute edge e_{ij}, adjust (tilt) tangent line t_j to make it coplanar with t_i
- Shortest path to j determines tangent line t_j for further computations



Least squares distance optimization

Fine-tune parameters of arc-line spline

- Minimize least squares distance of surface to input samples
- Nonlinear optimization problem
- Levenberg-Marquardt, approximated Jacobian

Use LRA parametrization of arc-line splines

- Length between bends
- Rotation between bends
- Angle of bend
- Radius of bend (usually fixed)

1. Reconstruction of bent tubes
2. Overview of the algorithm
   - Spine curve computation
   - Curve reconstruction
   - Spine curve segmentation
   - Least squares distance optimization
3. Results
   - Performance
   - Examples

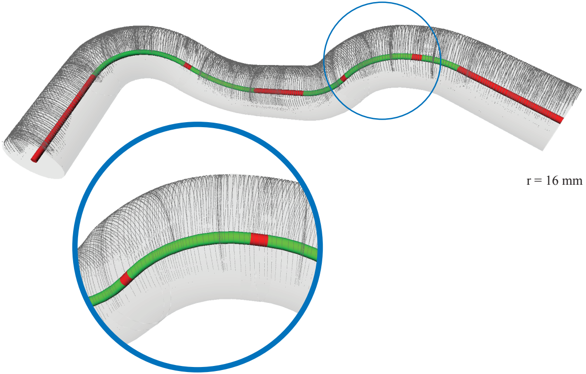
Running time

Typical example

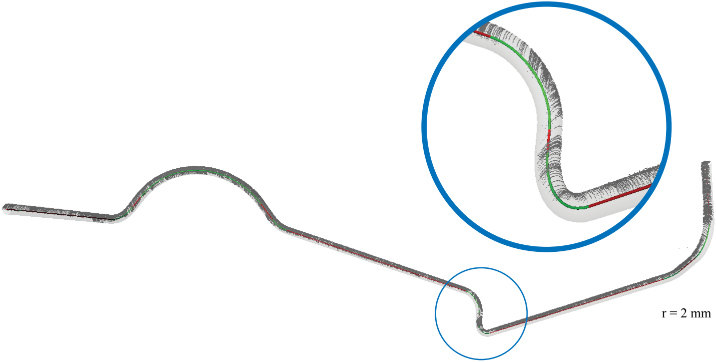
- 4 mm pipe radius
- 268 mm length
- 5 bends
- 6005 samples

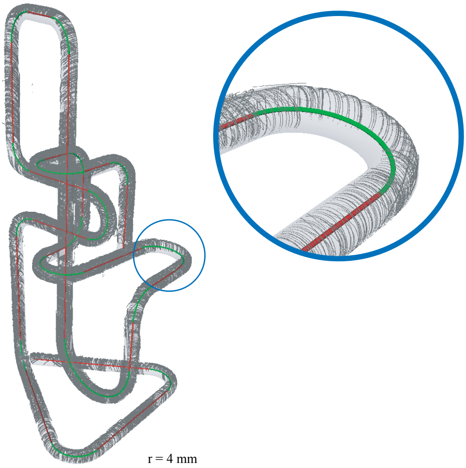
Total running time: 50 seconds

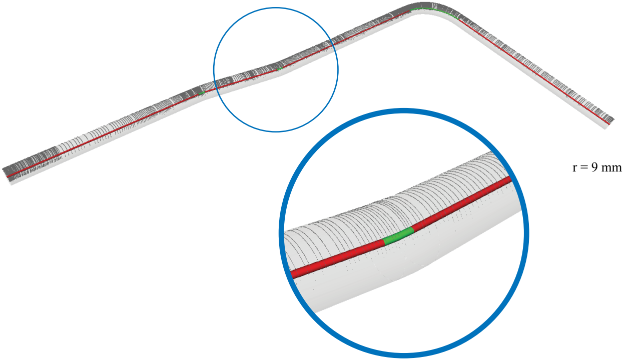
- Spine curve computation: 16 seconds
- Curve reconstruction: < 1 second
- Curve simplification: < 1 second
- Arc-line spline approximation: < 1 second
- Least-squares fitting: 34 seconds


r = 16 mm

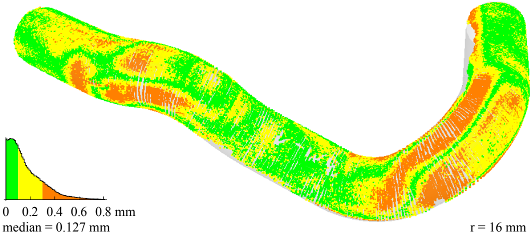
Examples




r = 4 mm



r = 9 mm


0 0.2 0.4 0.6 0.8 mm
median = 0.127 mm
r = 16 mm

Demonstration