Project 5

Arnaud Lina, Vincent Zalzal, Dominique Orban, Chris Prior, Apala Majumdar, Marina Chugunova, Beatriz Cortés, Etienne Ayotte-Sauvé

Outline

- Introduction

- Intersecting rectangles

- Equidistant points on a curve

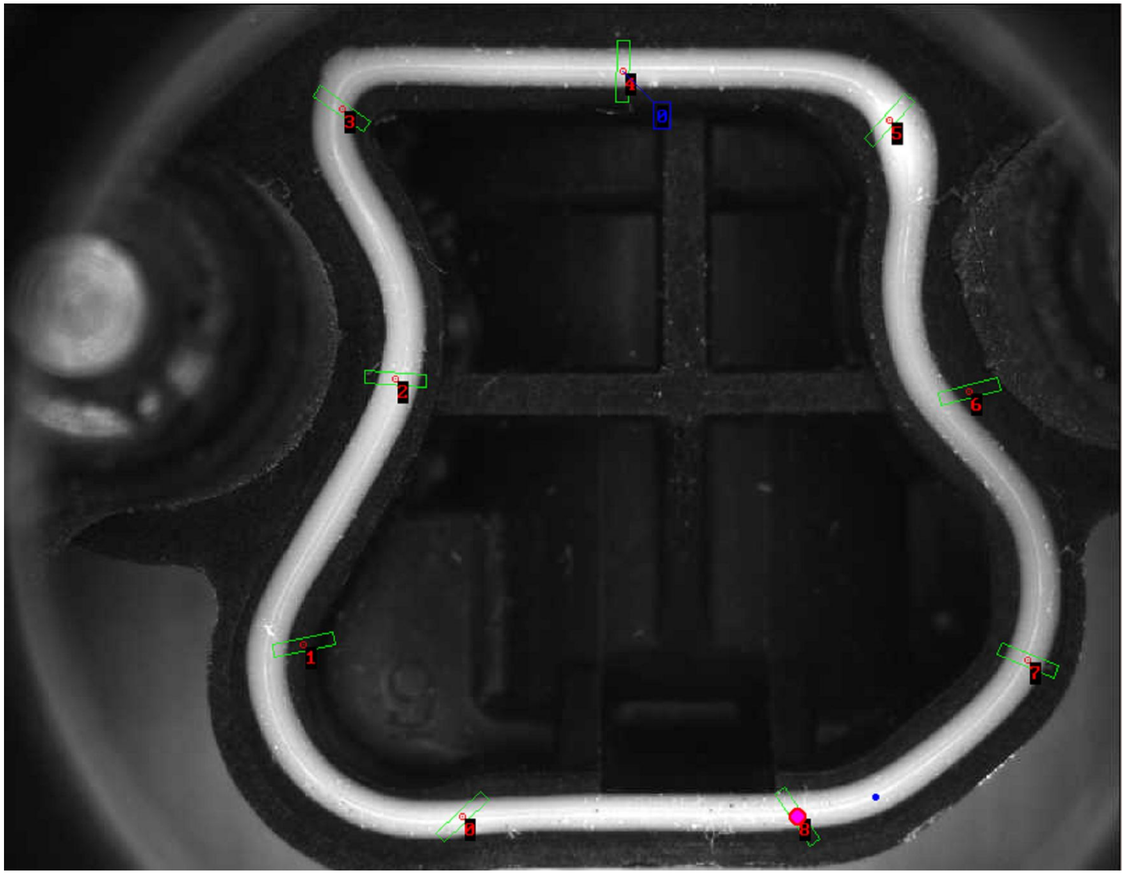
  - Mathematical programming approach
    - global parameterization: NLP
    - piecewise parameterization:
      - NLP with black-box constraints
      - MINLP
      - MPEC

  - Geometric approach
    - exhaustive algorithm by bisection
    - existence of solution (?)

- Conclusion and future work

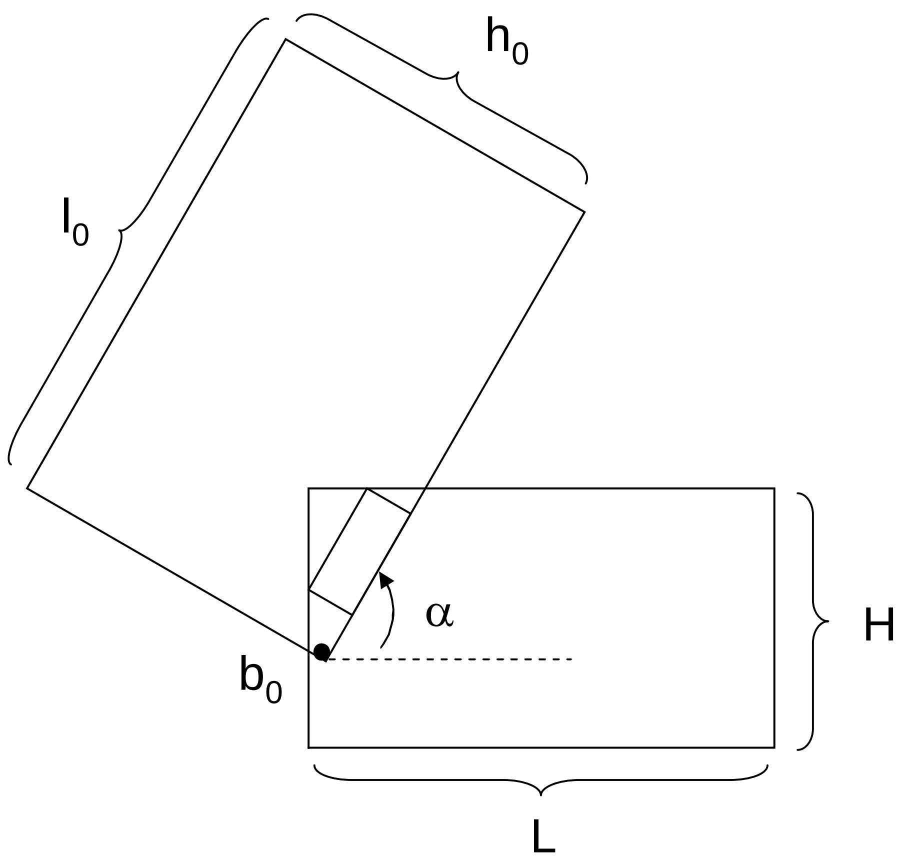
Introduction

- Matrox is a software/hardware design company for graphics, imaging and computer vision applications.



Intersecting rectangles

Given two rectangles in the plane, find a maximal area (*resp. length, height*) rectangle contained in their intersection with angle α.



Geometric approach

- Helmut Alt, David Hsu, and Jack Snoeyink (1995) solved the maximal area problem for the case where the intersection of the two rectangles consists in a convex polygon. They devised a nested binary search algorithm which converges in O(log(n)) time.

- Karen Daniels, Vitor Milenkovic, Dan Roth (1997) generalized previous results to non-convex polygonal intersections.

Mathematical programming approach

$$\max_{(l,h,b_1,b_2)\in\mathbb{R}^4} f\left(l,h,b_1,b_2\right)$$

subject to

$$\begin{cases}
l,h,b_1,b_2 \geq 0 \\
L - b_1 \geq 0 \\
H - b_2 \geq 0 \\
b_1 + l\cos(\alpha) \geq 0 \\
b_2 + l\sin(\alpha) \geq 0 \\
L - b_1 - l\cos(\alpha) \geq 0 \\
L - b_1 + h\sin(\alpha) \geq 0 \\
H - b_2 - l\sin(\alpha) \geq 0 \\
H - b_2 - h\cos(\alpha) \geq 0 \\
b_1 - h\sin(\alpha) \geq 0 \\
b_2 + h\cos(\alpha) \geq 0 \\
b_1 + l\cos(\alpha) - h\sin(\alpha) \geq 0 \\
b_2 + h\cos(\alpha) + l\sin(\alpha) \geq 0 \\
L - b_1 - l\cos(\alpha) + h\sin(\alpha) \geq 0 \\
H - b_2 - h\cos(\alpha) - l\sin(\alpha) \geq 0 \\
(b_1 - b_{0,1})\cos(\alpha) + (b_2 - b_{0,2})\sin(\alpha) \geq 0 \\
(b_2 - b_{0,2})\cos(\alpha) - (b_1 - b_{0,1})\sin(\alpha) \geq 0 \\
l_0 - (b_1 - b_{0,1})\cos(\alpha) - (b_2 - b_{0,2})\sin(\alpha) \geq 0 \\
h_0 - (b_2 - b_{0,2})\cos(\alpha) + (b_1 - b_{0,1})\sin(\alpha) \geq 0 \\
l + (b_1 - b_{0,1})\cos(\alpha) + (b_2 - b_{0,2})\sin(\alpha) \geq 0 \\
-l + l_0 - (b_1 - b_{0,1})\cos(\alpha) - (b_2 - b_{0,2})\sin(\alpha) \geq 0 \\
h + (b_2 - b_{0,2})\cos(\alpha) - (b_1 - b_{0,1})\sin(\alpha) \geq 0 \\
-h + h_0 - (b_2 - b_{0,2})\cos(\alpha) + (b_1 - b_{0,1})\sin(\alpha) \geq 0.
\end{cases}$$

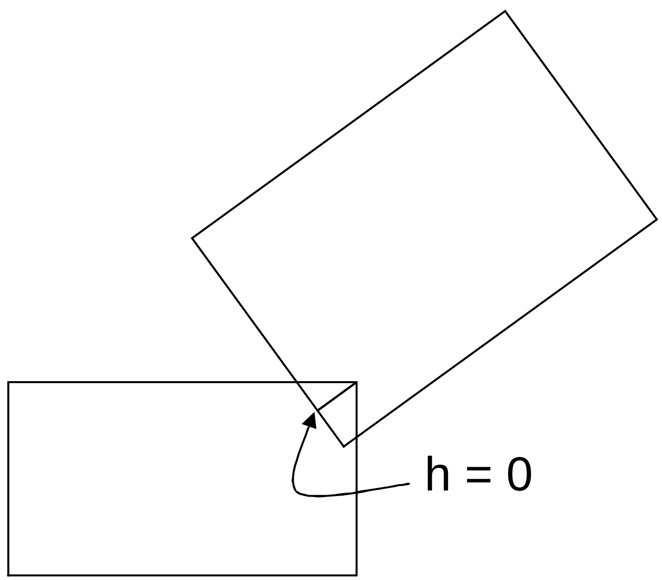
Remarks

- All constraints are linear inequalities.

- Implemented in Mathematica.

- Successfully solved 20 test cases.

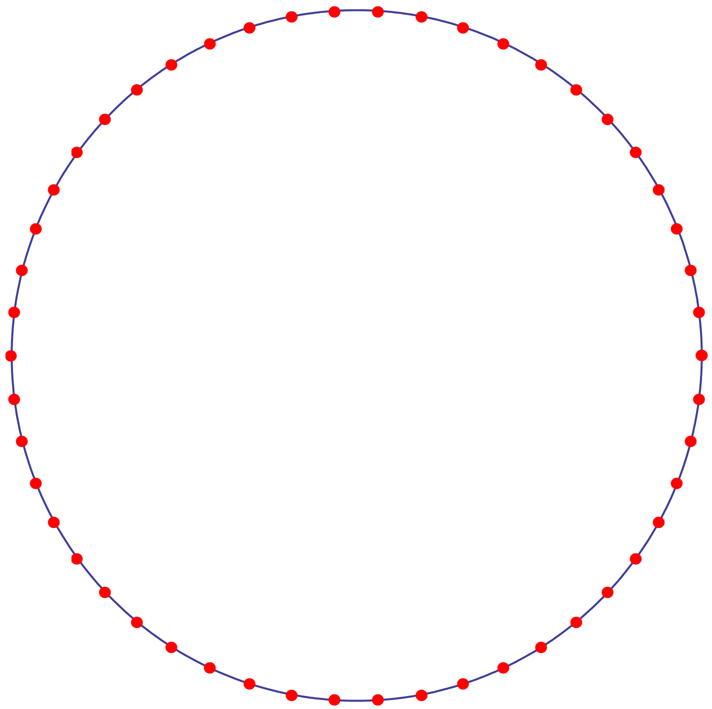
- If one wishes to maximize the height (*h*) or length (*l*) of the rectangle, then a (strictly) positive lower bound should be given to *l* or *h* (respectively) to avoid degenerate solutions, i.e. line segments.



Equidistant points on a curve

Consider a curve $C \subseteq \mathbb{R}^2$, i.e. $C = p[a,b]$ where $p:[a,b] \to \mathbb{R}^2$ is continuous.

Given a positive integer N > 1, find a set of N equidistant* points on C which contains the endpoints p(a) and p(b).



*: with respect to the Euclidean metric.

Mathematical programming approach

Assuming the parameterization p is known, one could then solve the following NLP optimization problem:

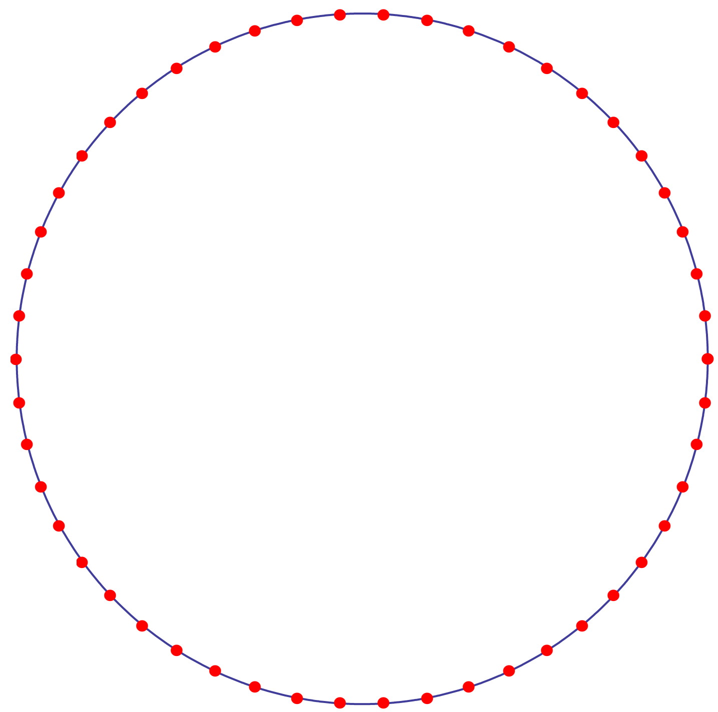
$$\max_{(t_1,\ldots,t_N)\in[a,b]^N} \sum_{i=1}^{N-1} \left\| p(t_{i+1}) - p(t_i) \right\|^2$$

subject to

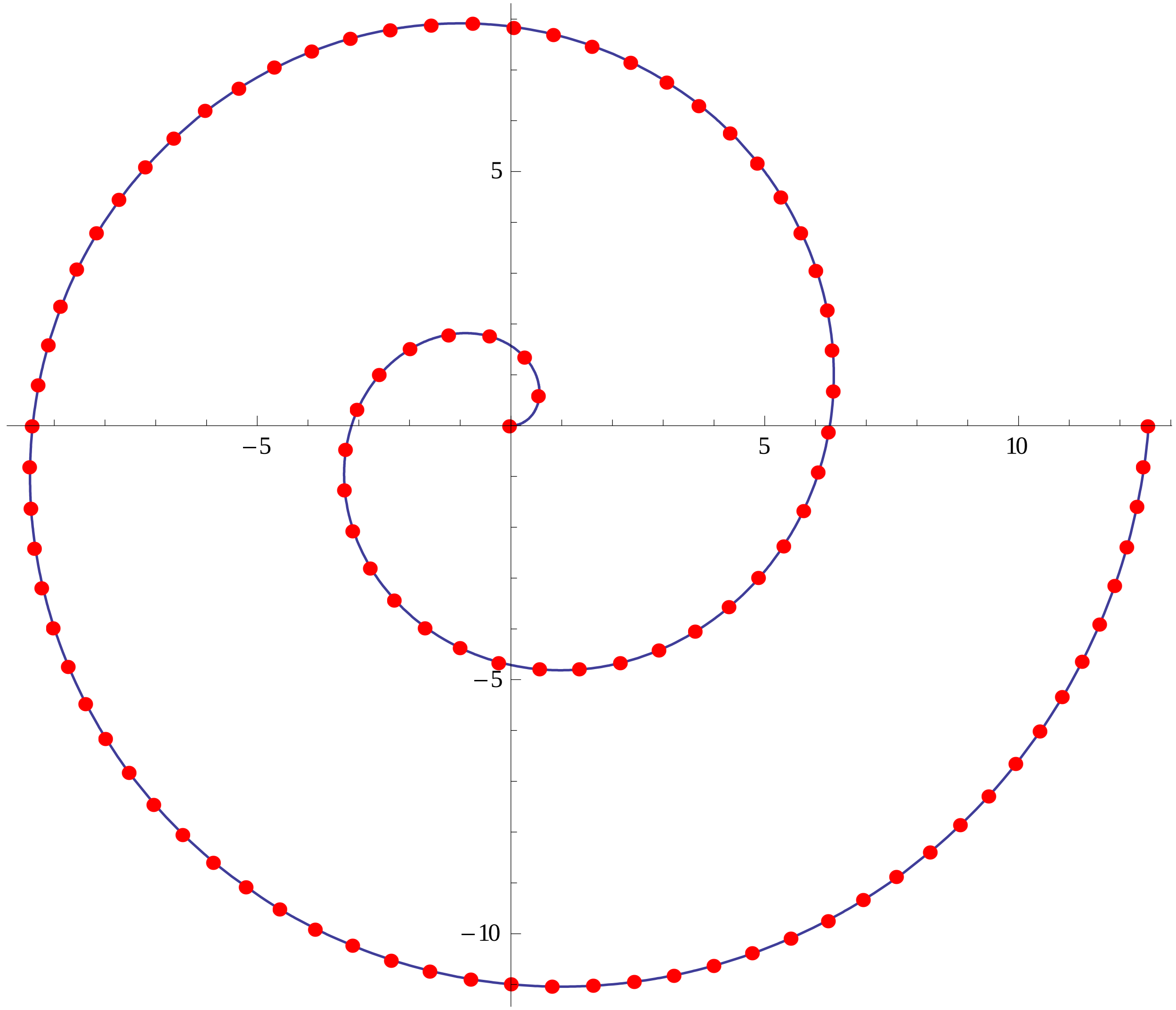
$$\begin{cases} t_1 = a \\ t_N = b \\ t_i \le t_{i+1},\ i = 1,\ldots,N-1 \\ \left\| p(t_{i+1}) - p(t_i) \right\|^2 = \left\| p(t_2) - p(t_1) \right\|^2,\ i = 2,\ldots,N-1. \end{cases}$$

Case 1: smooth global parameterization

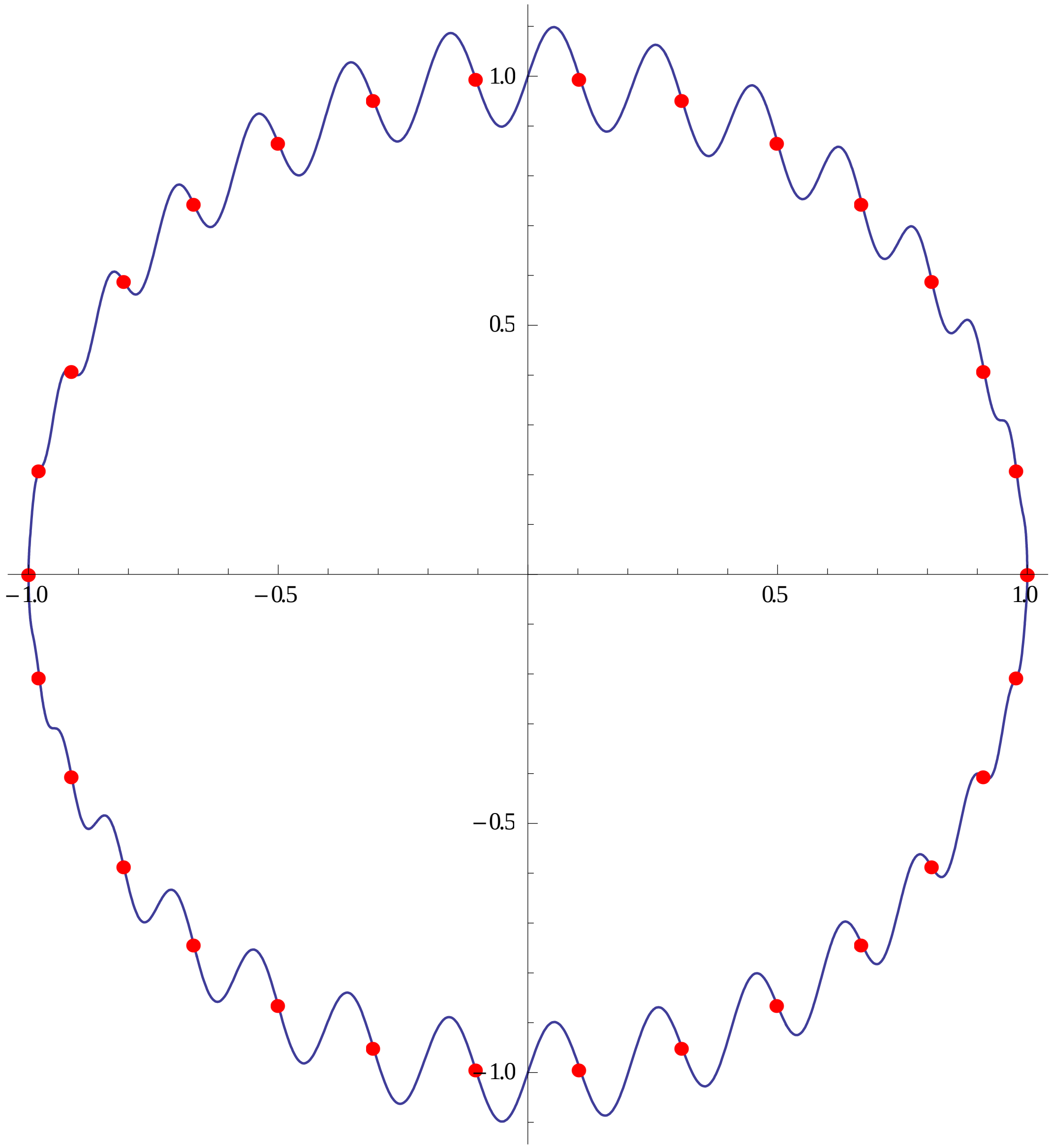
For this case, existing commercial and open-source NLP solvers (CONOPT, IPOPT, KNITRO, SNOPT, etc.) were tested on three examples.



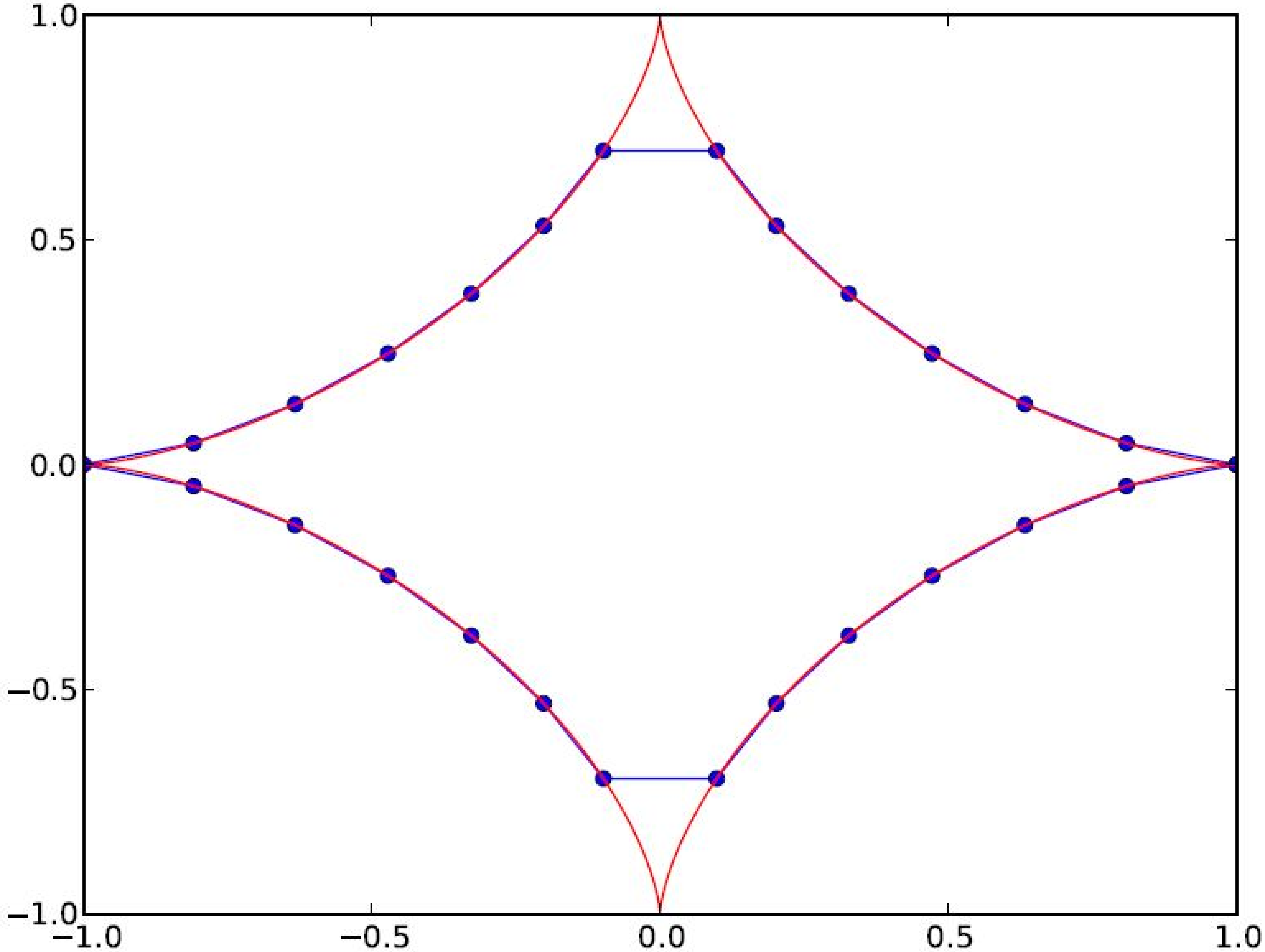
$$p(t) = (\cos(t), \sin(t)),\ 0 \le \mathrm{t} \le 2\pi$$



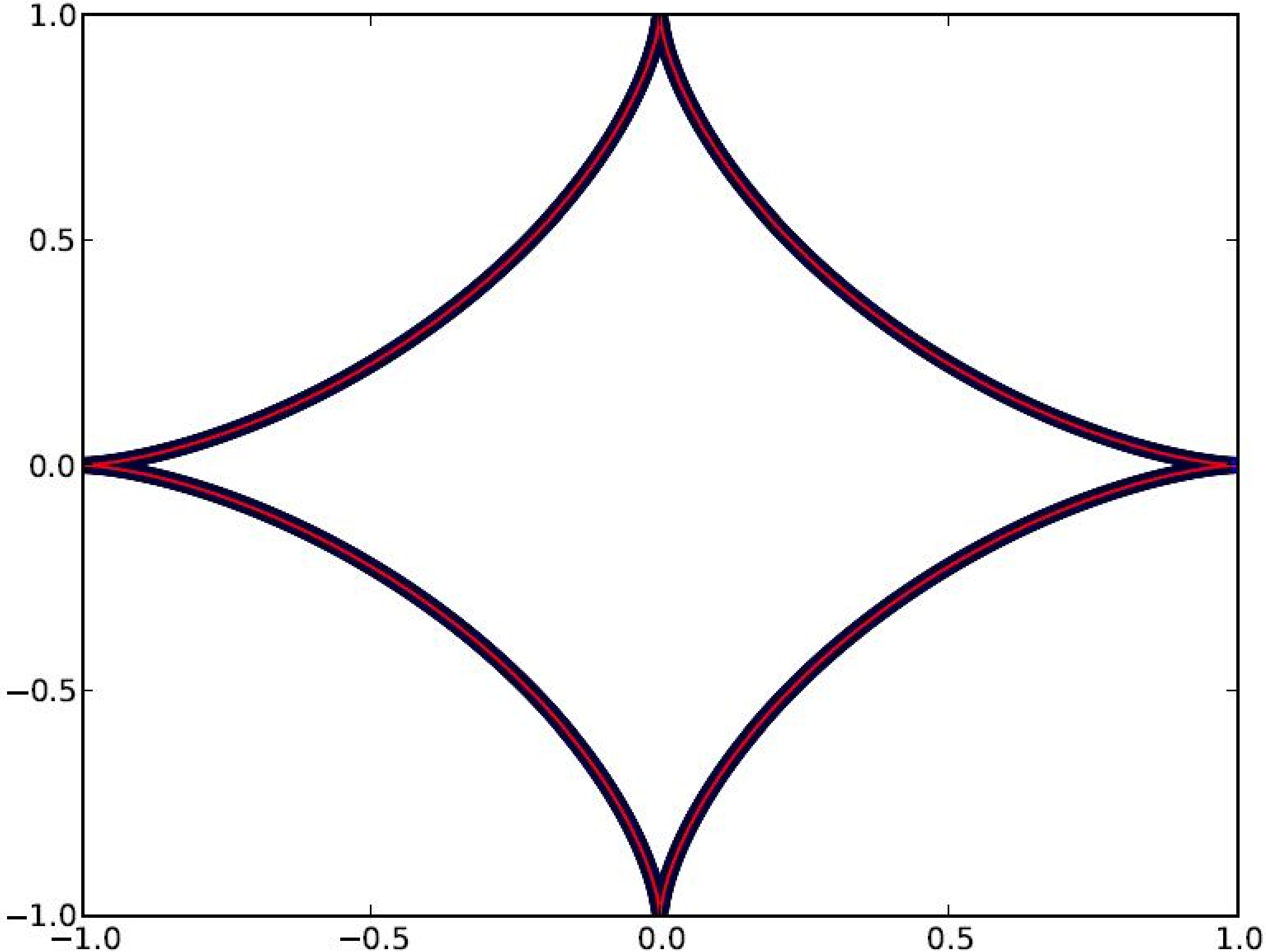
$$p(t) = (t\cos(t), t\sin(t)),\ 0 \le \mathrm{t} \le 4\pi$$



$$p(t) = (\cos(t), \sin(t)\,(1 + (1/10)\sin(30\,t))),\ 0 \le \mathrm{t} \le 2\pi$$



$$p(t) = (\sin(t)^3, \cos(t)^3),\ 0 \le \mathrm{t} \le 2\pi$$


1.0
0.5
0.0
−0.5
−1.0
−1.0
−0.5
0.0
0.5
1.0

Case 2: piecewise parameterization

$$p(t) = p_j(t)$$

$$\begin{cases} t \in \left[a_{j-1,} a_j \right] \\ \mathrm{a} = \mathrm{a}_0 < \mathrm{a}_1 < \ldots < \mathrm{a}_{M-1} < \mathrm{a}_M = b \text{ partition of [a,b]} \\ p_j : \left[a_{j-1} - \varepsilon, a_j + \varepsilon \right] \to \mathbb{R}^2 \text{ at least twice cont. diff. (j = 1, \ldots, M)} \end{cases}$$

It is assumed that p is at least twice cont. diff.

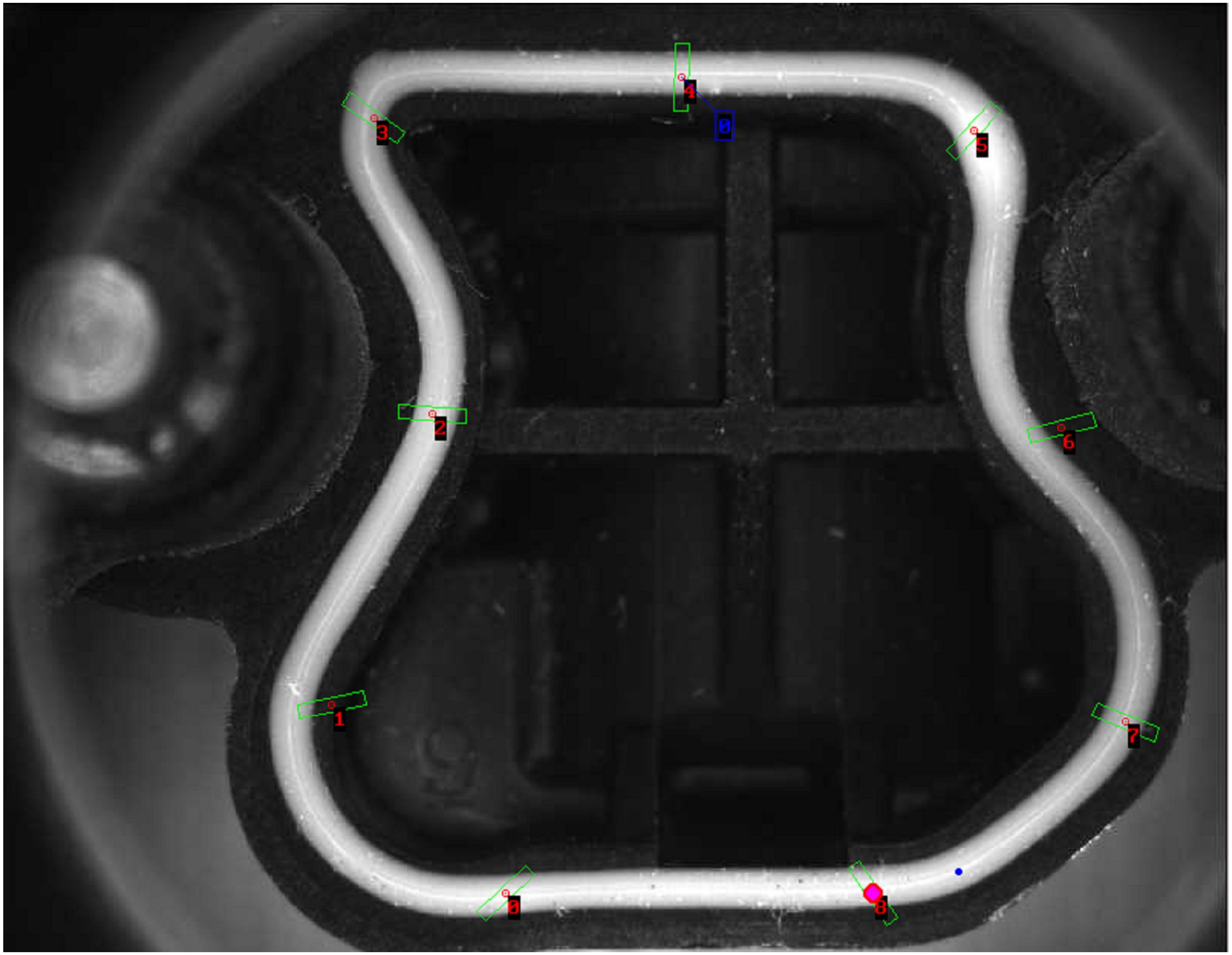
Approaches

- NLP with black-box constraints corresponding to parameterization
- MINLP with integer variables corresponding to each interval of the partition
- MPEC (NLP)

$$\max_{(t_1,\ldots,t_N)\in[a,b]^N} \sum_{i=1}^{N-1} \left\| x_{i+1} - x_i \right\|^2$$

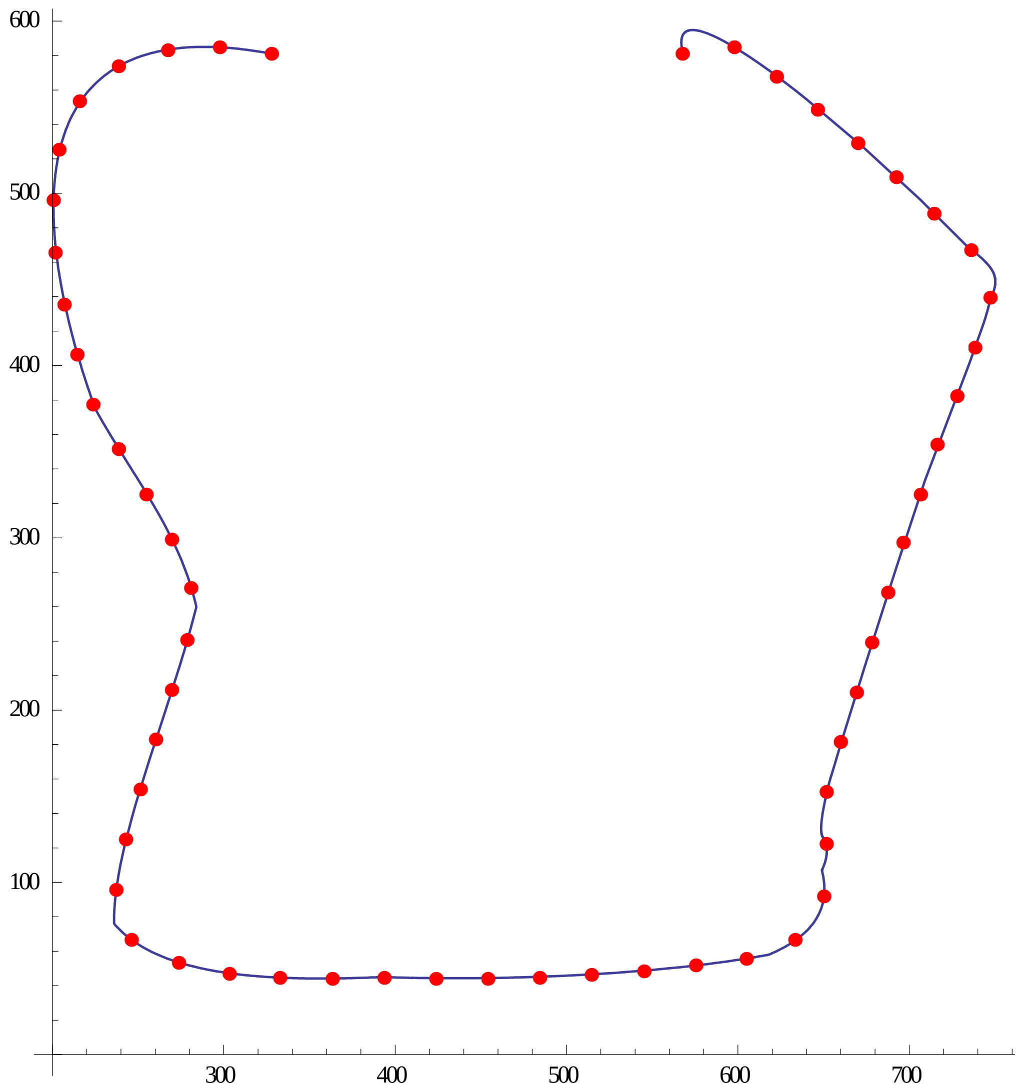
subject to

$$\begin{cases} t_1 = a \\ t_N = b \\ t_i \le t_{i+1},\ i = 1, \ldots, N-1 \\ \left\| x_{i+1} - x_i \right\|^2 = \left\| x_2 - x_1 \right\|^2,\ i = 2, \ldots, N-1 \\ x_i = \sum_{j=1}^{M} y_{i,j}\, p_j(t_i),\ i = 1, \ldots, N \\ \sum_{j=1}^{M} y_{i,j} = 1,\ i = 1, \ldots, N \\ y_{i,j}\lambda_{i,j} = 0,\ i = 1, \ldots, N,\ j = 1, \ldots, M \\ (t_i - a_{j-1})(t_i - a_j) = \lambda_{i,j} + \mu_i,\ i = 1, \ldots, N,\ j = 1, \ldots, M \\ y_{i,j}, \lambda_{i,j} \ge 0,\ i = 1, \ldots, N,\ j = 1, \ldots, M \\ \mu_i \in \mathbb{R},\ i = 1, \ldots, N \\ x_i \in \mathbb{R}^2,\ i = 1, \ldots, N \end{cases}$$

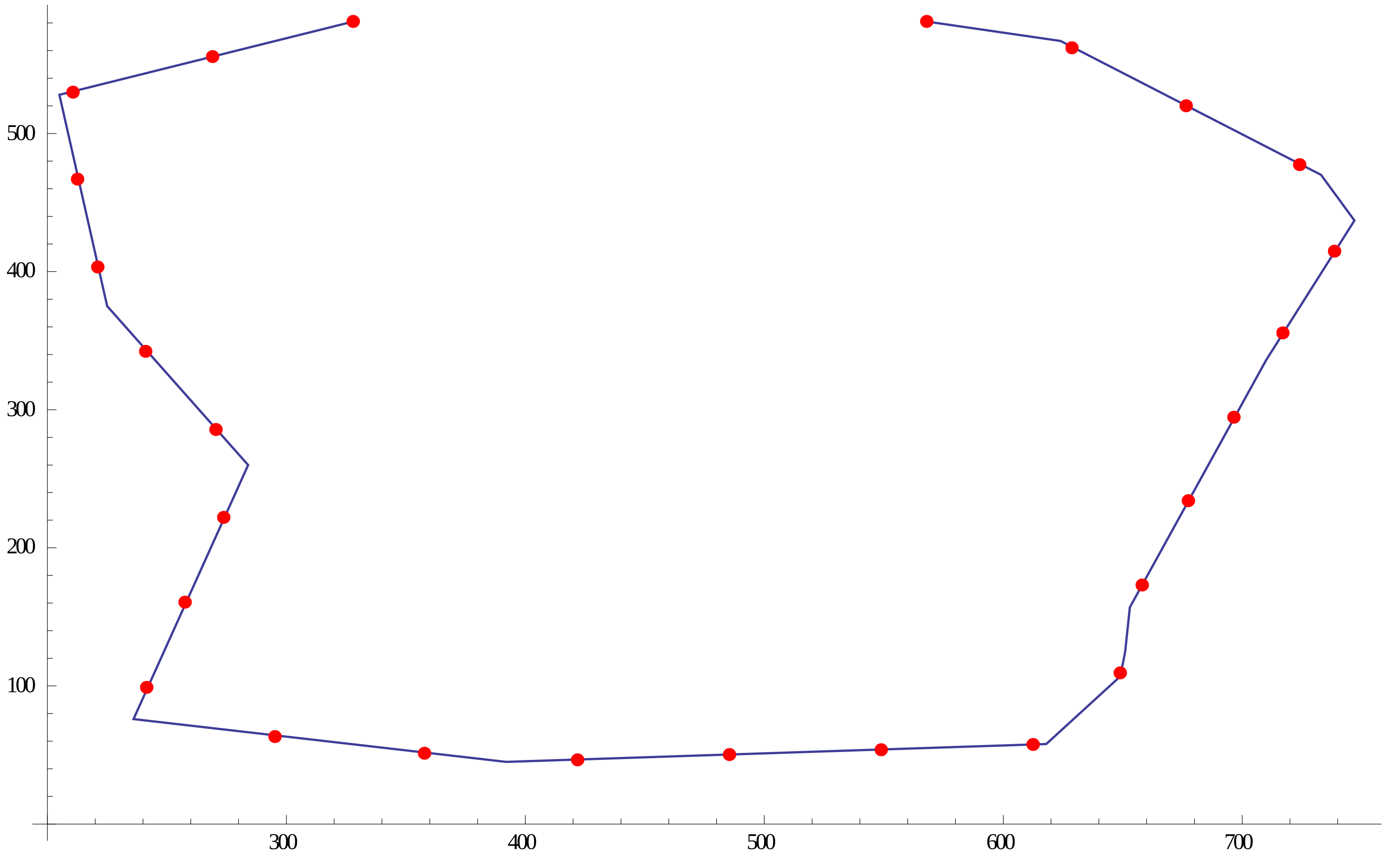


4
0
3
5
2
6
1
7
0
8

Piecewise parameterization (black-box & MPEC)



PL parameterization (MPEC)



Geometric approach: summary of results

- A closed method which guarantees it will find a solution if one exists.
- An algebraic characterization of the solution set for polylines with two or three segments.
- A geometric algorithm based on the mean value theorem.

Geometric approach: a bisection algorithm

It's obvious that the distance **d** between two consecutive points is bounded above by the length of the curve divided by **N+1** and bounded from below by **|AB|/(N+1)**

lb = |AB|/(N+1) ≤ d ≤ arc(AB)/(N+1)=up

- First approximation of **d** is taken as **d1 = (lb + ub)/2**

- Starting from the point **A** we compute all possible equidistant distributions of **a{0}, a{1}, ... a{N}** [finding coordinates of **a{i}** for **i =1..N** as intersections between line segments of the given curve and a circle of radius **d** centered at **a{i-1}** points consequently, for multiple intersections we collect a complete tree of all solutions.

- We mark each computed solution as an **undershoot** or an **overshoot** depending on the location of the point **a{N}** with respect to the point **a{N+1} = B**

Geometric approach: a bisection algorithm

- On each **i > 1** iteration we adjust **d** as **d{i} = (lb+d{i-1})/2**

for each an **overshoot** type solution and as **d{i} = (ub+d{i-1})/2**

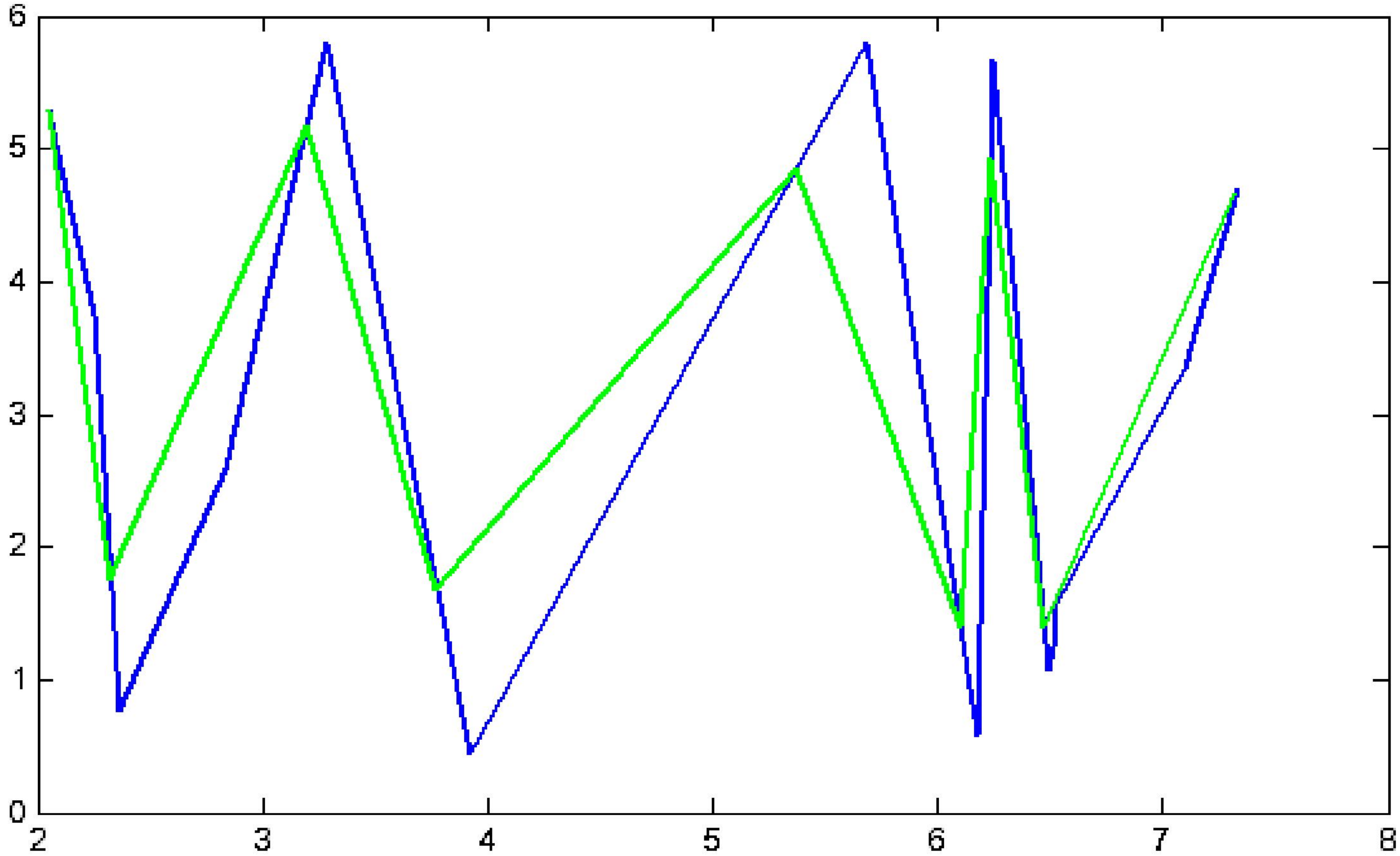
for each an **undershoot** type solution. And repeat computations of all solutions for the new distance **d = d{i}.**

- We stop and claim that a numerical solution is found when we reach a tolerance **|a{N} - B| ≤ 0.001 arc(AB).**

- Getting a finite number of numerical solutions we choose the one that satisfies a given optimization criteria.

Downside of the proposed algorithm: the total number of solutions can be relatively large if the original curve has spiral type pieces but at the same time the total number of solutions tends to one as the number of points **N** goes to infinity.

Bisection results



N = 7

Geometric approach analysis: two segment case

Let $\mathrm{N} \in \mathbb{N}$, R, L $>$ 0, $\alpha, \beta \geq 0$ such that $\alpha + \beta = \pi / 2$.

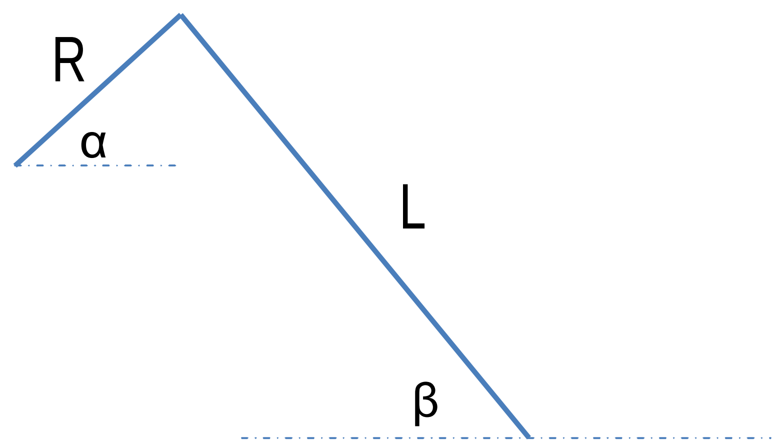
For each $\mathrm{N}_1, \mathrm{N}_2 \in \mathbb{N}$, $0 < \delta < 1$ such that $\mathrm{N}_1 + \mathrm{N}_2 = \mathrm{N}$

and $R / L = (N_1 + \delta)/(N_2 + \sqrt{1 - \delta^2})$, there exists a unique

solution to the equidistance problem with N_1 and N_2

points on the first and second segment (resp.) and spacing

$\mathrm{R}/(N_1 + \delta)$.



Geometric approach analysis: three segment case

Equally distribute N points (with respect to the Euclidean norm) on a polyline consisting of three segments of lengths $L_1,\ L_2\ ,L_3$:
three equations for three unknowns (h, d_1, d_2)

$$\left(d_1 - N_1 h\cos\alpha\right)^2 + \left(d_2 - N_1 h\sin\alpha\right)^2 =$$

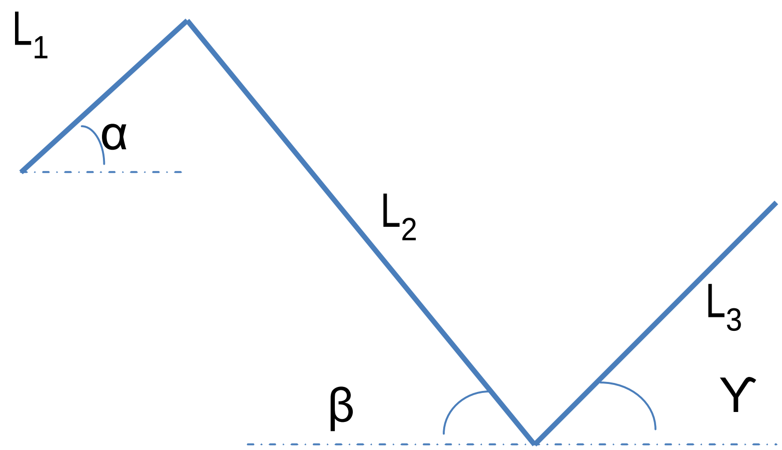
$$= \left(d_1 - \left(L_2 - N_2 h\right)\cos\beta - \left(L_3 - N_3 h\right)\cos\gamma - L_1\cos\alpha\right)^2 +$$

$$\left(d_2 + \left(L_2 - N_2 h\right)\sin\beta - \left(L_3 - N_3 h\right)\sin\gamma - L_1\sin\alpha\right)^2$$

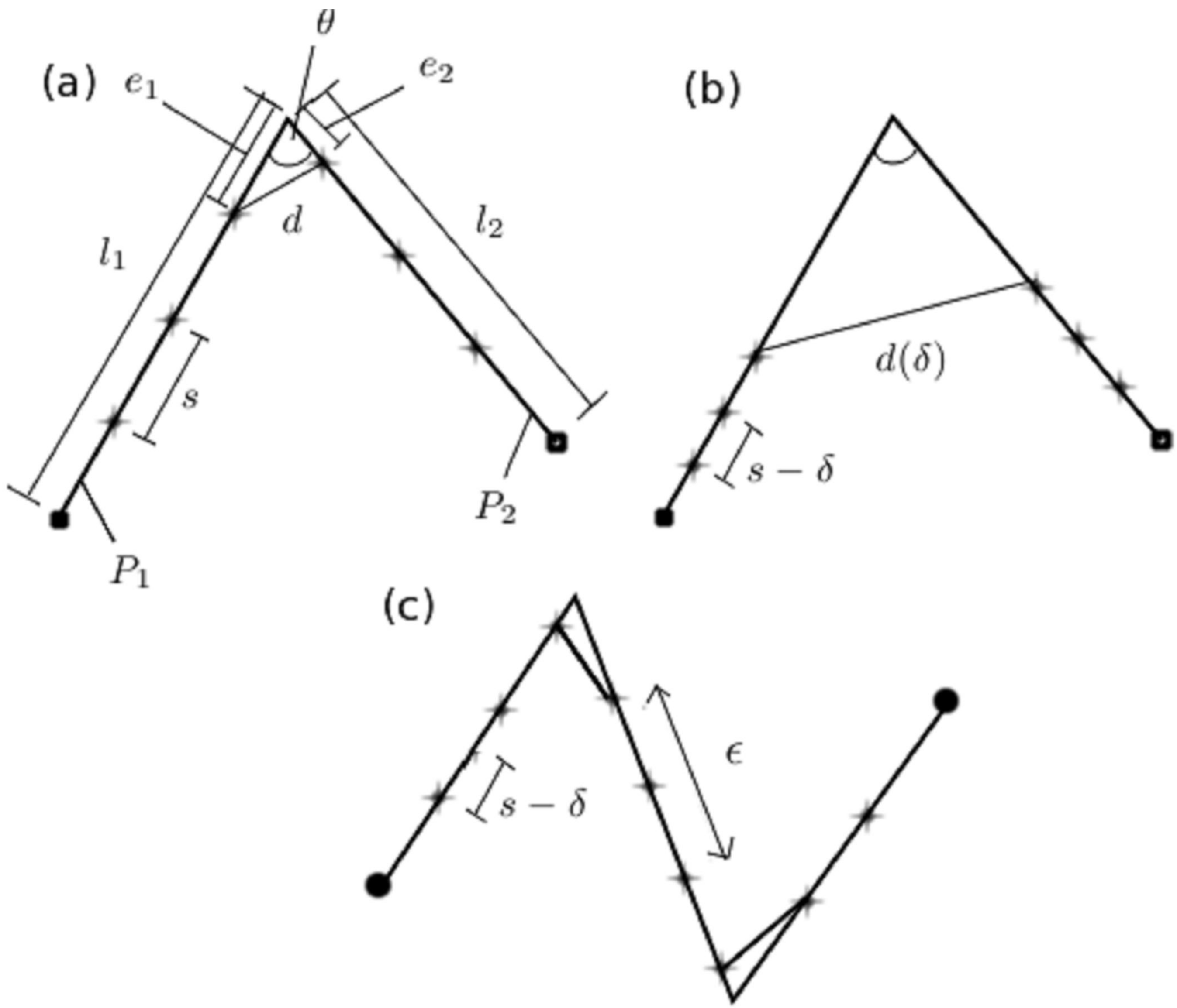
For each $0<\delta<1$

$$\left(d_1 - N_1 h\cos\alpha\right)^2 = \delta^2 h^2$$

$$\left(d_2 - N_1 h\sin\alpha\right)^2 = \left(1-\delta^2\right)h^2$$



Alternate approach to existence



Conclusions and future work

- Equidistance condition can be relaxed.
- Existence in general case ?
- More efficient implementation of bisection algorithm.
- Alternative objective functions.