Topology of singular curves and surfaces

Scientific context

The description of singular varieties (self-intersecting curves or surfaces for example) arises in a wide range of applications, from scientific visualization to the design of robotic mechanisms. Currently, most software provide either (a) numerical approximations without topological guarantee; or (b) rely on symbolic computer algebra methods to handle singular varieties but suffer from efficiency in practice. Computing the topology of a variety consists in computing a piecewise-linear graph or triangulation that can be deformed continuously toward the input variety. In particular, it should preserve the number of connected components, the self-intersection points, etc. This PhD subject addresses the design and the implementation of algorithms based on interval arithmetic to compute the topology of restricted classes of singular varieties.

In the case of a smooth variety, several numerical methods can guarantee its topology. One can mention global subdivision [5, 9], or continuation approaches [6]. For a singular variety, computing its topology requires i) to detect its singularities, ii) to compute the topology in a neighborhood of those singularities, iii) to compute the topology of the smooth remaining part of the variety. State-of-the-art methods to compute it are based on symbolical computer algebra algorithms such as the Cylindrical Algebraic Decomposition (see [8] and references therein). In the general case, no numerical method can yet handle the detection of the singularities, neither the computation of their local topology.
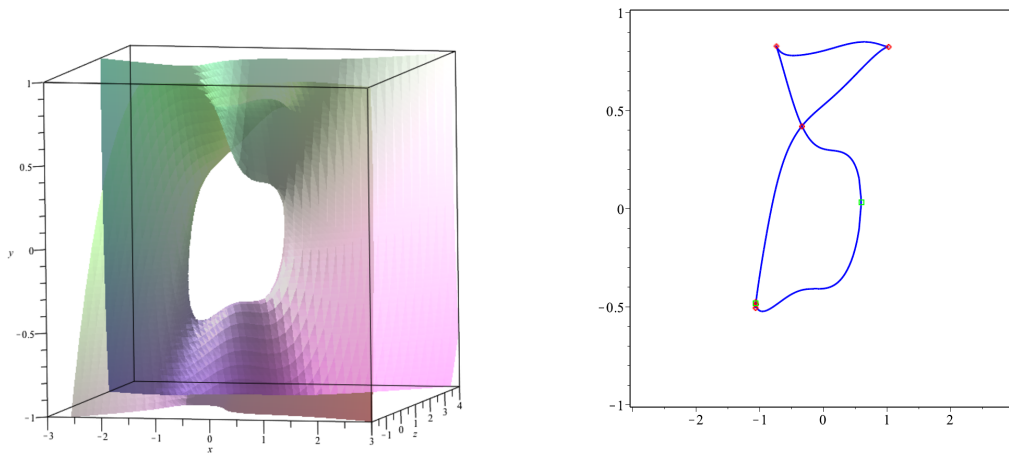

Figure 1. Left: a surface $f(x, y, z) = 0$. Its silhouette curve is defined by the system $f = \frac{\partial f}{\partial z} = 0$. Right: the projection of the silhouette is singular with node and cusp singularities.

Motivation

The singular curves and surfaces appearing in several applications often belongs to a restricted class of singular varieties. For instance, when visualizing a surface, it is natural to compute the boundary of its shadow. Although it might be a singular curve, it is often the projection of a smooth curve, called the silhouette or polar-variety of the surface (Figure 1). In this case, we are interested in describing singular curves in the plane that are projection of smooth curves.

Another example in mechanical design is the description of the workspace of robots. For example, consider a manipulator in the plane with two degrees of freedom (Figure 2). The points A and B are fixed and two actuators control the end effector M of position (x,y) in $\mathbb{R}^2$ through the lengths of two links l_1 and l_2. Its workspace is the surface in $\mathbb{R}^4$ of the values (l_1, l_2, x, y) for which the robot can be assembled. It is a smooth surface and to visualize it, we project it in $\mathbb{R}^3$. This projection is a singular surface (Figure 3).

More generally, a natural class of singular surfaces is the projection of smooth surfaces of $\mathbb{R}^n$ on $\mathbb{R}^3$. These surfaces appear naturally when visualizing surfaces and modelling robotic mechanisms with one or two degrees of freedom. From a mathematical point of view, their singularities have been studied and classified [1, 2]. Yet, no numerical method handles the detection of these singularities nor the computation of their local topology.
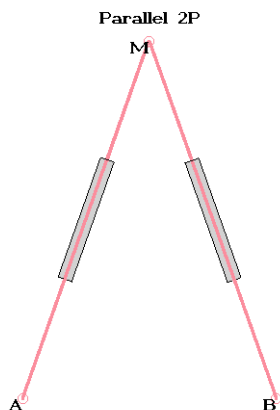

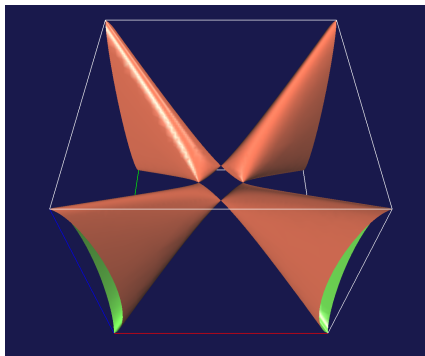Figure 2. Parallel manipulator with 2 legs.

Figure 3. Projection of the workspace of the 2-P manipulator in (r_1, r_2, x).

Missions

If we consider a curve $\mathcal{C}$ defined by $f(x, y) = 0$, its singular points are the solutions of the three equations in 2 variables $f = 0, \frac{\partial f}{\partial x} = 0, \frac{\partial f}{y} = 0$.

To isolate these singularities, we want to use a numerical method based on Newton algorithm and interval arithmetic. More precisely, given a square polynomial system (as many equations as variables) and a box B, the evaluation of a Newton operator on B can allow us to decide if B contains or not a solution (see [7, 10] and references therein). Unfortunately the system defining the singularities of f is not square and we cannot guarantee its solutions with numerical methods based on Newton directly.

Considering a more restricted case, let $f_1 = \cdots = f_{n-1} = 0$ be a system of $n-1$ equations defining a smooth curve of $\mathbb{R}^n$, and let $\mathcal{C}_{proj}$ be its projection on the first two variables. In [3, 4], we extracted square systems encoding the singularities of $\mathcal{C}_{proj}$ in the case $n = 3$ and we showed that this system can be used in numerical isolation algorithms. However the problem remains open for $n \geq 4$.

The first challenge will be to design a numerical algorithm computing a piecewise linear curve with the same topology as $\mathcal{C}_{proj}$ (Figure 4) for $n \geq 4$. In particular, the algorithm should always return a correct result, or eventually not terminate if some assumptions on the f_i are not verified.

A second related challenge is to handle the case where the curve in the plane is the silhouette of a surface embedded in higher dimension (Figure 1). This case arises notably in robotic applications and the method developed will be tested on some mechanism modelings coming from the robotic field.

Finally, the candidate will also address the problem of triangulating singular surfaces $\mathcal{S}$. In the general case there is no method to extract a square system of polynomial equations that vanish exactly on the singularities of $\mathcal{S}$. However, we think that the problem is tractable in the restricted case where $\mathcal{S}_{proj}$ is the projection in $\mathbb{R}^3$ of a smooth surface in $\mathbb{R}^n$. The challenge in this case is to compute a triangulation with the same topology as $\mathcal{S}_{proj}$. Compared to the previous challenges, one of the difficulties in this case is that we need to guarantee that the self-intersection curves of $\mathcal{S}_{proj}$ have the same topology as a piecewise-linear graph included in the edges of its triangulation.
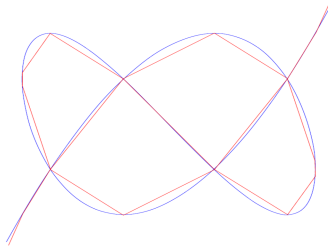

Figure 4. The red piecewise-linear graph is a topology preserving vectorial representation of the blue curve (computed with the software Isotop).

Supervision and contacts

Sylvain Lazard and Guillaume Moroz (firstname.name@inria.fr)

Team: Gamble (LORIA laboratory, Lorraine University, Inria, CNRS), http://gamble.loria.fr/

Profile of the candidate

The candidate should have a taste for both mathematics (geometry or numerical analysis) and computer science. Programming skills would be appreciated.

Bibliography

[1] Demazure, M. 2000. *Bifurcations and catastrophes: Geometry of solutions to nonlinear problems.* Springer.

[2] Hobbs, C.A. and Kirk, N.P. 2001. On the classification and bifurcation of multigerms of maps from surfaces to 3-space. *Math. Scand.* 89, 1 (2001), 57–96.

[3] Imbach, R., Moroz, G. and Pouget, M. 2017. A certified numerical algorithm for the topology of resultant and discriminant curves. *Journal of Symbolic Computation.* 80, Part 2, (2017), 285–306.

[4] Imbach, R., Moroz, G. and Pouget, M. 2015. Numeric and certified isolation of the singularities of the projection of a smooth space curve. *Proceedings of the 6th international conferences on mathematical aspects of computer and information sciences* (Berlin, Germany, 2015).

[5] Liang, C., Mourrain, B. and Pavone, J. 2008. Subdivision methods for 2d and 3d implicit curves. *Geometric modeling and algebraic geometry.* Springer. 171–186.

[6] Martin, B., Goldsztejn, A., Granvilliers, L. and Jermann, C. 2013. Certified parallelotope continuation for one-manifolds. *SIAM J. Numerical Analysis.* 51, 6 (2013), 3373–3401.

[7] Moore, R.E., Kearfott, R.B. and Cloud, M.J. 2009. *Introduction to interval analysis.* Siam.

[8] Mourrain, B., Pion, S., Schmitt, S., Técourt, J.-P., Tsigaridas, E.P. and Wolpert, N. 2006. Algebraic issues in Computational Geometry. *Effective Computational Geometry for Curves and Surfaces.* J.-D. Boissonnat and M. Teillaud, eds. Springer. 117–155.

[9] Plantinga, S. and Vegter, G. 2004. Isotopic approximation of implicit curves and surfaces. *SGP '04: Eurographics/acm siggraph symposium on geometry processing* (Nice, France, 2004), 245–254.

[10] Rump, S.M. 2010. Verification methods: Rigorous results using floating-point arithmetic. *Acta Numerica.* 19, (May 2010), 287–449.