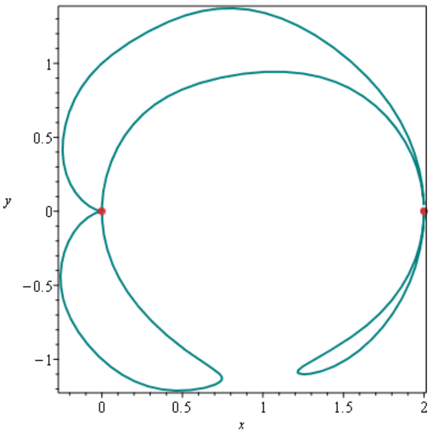
Validated Numerical Software For Algebraic Curves With Singularities



- **where ? CRIStAL, Université de Lille** — in the research group CFHP (Computer Algebra and HPC).
- **who ?** *student :* Master in CS or mathematics. *advisors :*
  - Adrien Poteaux (`adrien.poteaux@univ-lille.fr`),
  - Florent Bréhard (`florent.brehard@univ-lille.fr`).
- **when ?** spring 2025 (4 – 6 months)

I. Scientific context: Numerical computation with singular algebraic curves

Motivation. Real and complex algebraic curves are fundamental for many applications in computer science, mathematics and physics. Although they are simply defined *implicitly* by polynomial relations in their coordinates:

$$P_1(x_1,\dots,x_n) = \dots = P_r(x_1,\dots,x_n) = 0, \qquad P_j \in \mathbb{K}[x_1,\dots,x_n] \quad \text{with } \mathbb{K} = \mathbb{Q}, \mathbb{R}, \mathbb{C}\dots,$$

manipulating them efficiently and *explicitly* (parametrization, intersection, topology, etc.) requires sophisticated algorithms that have been continuously developed over last decades in computational algebraic geometry [8, 1], either purely symbolically or with the use of numerics [22, 13].

Singularities are the points where the curve is not similar to a line, like a pinch or a crossing of two branches (see the two red dots in the figure above). They occur in many situations like the plane projection of a space curve, or when a robot passes through a singular position. Particular care is needed at singularities since algorithms designed for regular curves may exhibit critical behavior at those points, like division by zero or numerical instability. This is a challenge for applications where maximum confidence is required, such as safety-critical engineering or computer-aided proofs in mathematics: a surgical robot is *not* safe up to erratic numerical behavior, *nor* is a geometry theorem true up to rounding errors.

Validated numerics [17, 25] aims at computing with numerical set-valued representations (real intervals, complex balls, set of functions described by a polynomial approximation and an error bound, etc.), thus exploiting the efficiency of floating-point arithmetic while guaranteeing actual mathematical statements: the solution is contained in the computed set. Such techniques have been successfully employed for critical applications and computer-assisted proofs (see, e.g., [24, 26, 6]). ***The goal of this internship is to treat singularities of algebraic curves using symbolic-numeric methods and validated numerics to combine efficiency and reliability.***

II. Objective: A validated symbolic-numeric Newton-Puiseux algorithm

The Newton-Puiseux algorithm computes parametrizations of the branches of a curve implicitly defined by $P(X,Y) = 0$ at a singularity x_0 under the form of a Puiseux series (i.e., power series with fractional exponents), with *algebraic* coefficients a_k:

$$Y(X) = \sum_{k \geqslant k_0} a_k (X - x_0)^{\frac{k}{e}}, \qquad e \in \mathbb{N}^*, \quad k_0 \in \mathbb{Z}, \quad a_k \in \mathbb{C} \quad (k \geqslant k_0).$$

Despite significant improvements made on Newton-Puiseux over the last years (see [20, 21] and references therein), the intrinsic representation size of the algebraic numbers a_k makes this symbolic algorithm not competitive, even for problems $P(X,Y) = 0$ of moderate degree (10 – 100). Consequently, in presence of singularities, many algorithms avoid the use of Puiseux series and prefer *turning around* such points when possible. Doing so, however, they ignore the crucial geometrical information encoded by singularities, and they increase the risk of numerical instability when working close to a singular point without exploiting it.

Yet, in many situations, computing accurate numerical approximations of the a_k together with rigorous and tight error bounds, rather than exact representations, is sufficient. Therefore, we propose to design a validated symbolic-numeric Newton-Puiseux algorithm, following these steps:

1. Compute the coefficients a_k numerically *following the structure* of the Puiseux series (i.e., its exponents), obtained using [19]. Note that, without this structure information, such a numerical method applied close to a singularity would be highly unstable.

2. Finally, design a validation method to compute rigorous and tight error bounds on the a_k's approximate values. This will involve techniques known as *fixed-point a posteriori validation* [5, §3.3], where error bounds are obtained afterwards from the application of a suitable fixed-point theorem. A specific difficulty here to tackle is that singular equations are typical examples of ill-posed problems: they become regular but highly ill-conditioned under infinitesimal perturbations.

Besides theoretical results, another important part of the internship is a neat implementation in Julia of this validated symbolic-numeric Newton-Puiseux algorithm, relying on the Arb library for validated numerics [16]. Our objective is treating examples of much higher degrees, say 1,000 – 10,000, than the purely symbolic Newton-Puiseux algorithm.

III. The future: Applications to computational algebraic geometry

We have a ***guaranteed PhD funding*** to continue the work initiated during this internship. The resolution of singularities of algebraic curves using the symbolic-numeric Newton-Puiseux algorithm will be the cornerstone for improvements of algorithms in real and complex algebraic geometry, which in the future (beyond this internship) will help tackling computationally challenging applications in the following domains:

- *Robotics* often involves polynomial systems describing real algebraic (or semi-algebraic) varieties representing, for example, the possible configurations of a robot. *Connectivity queries* are essential for motion planning, and they can be handled by computing a *roadmap* of the algebraic variety [7, 12], thus reducing the problem to connectivity queries on real algebraic curves. A possible approach for this [15, 14] is to project the curve onto a plane and to analyze the resulting 2D singular curve [11]. Such an analysis could be improved with the use of symbolic-numeric Puiseux series.

- *Homotopy methods* are used to compute numerical roots of polynomial systems by deformation (the homotopy) from simpler systems [3]. The curves tracking the roots may cross each other, resulting in singularities that can be treated rigorously with the validated symbolic-numeric Newton-Puiseux algorithm.

- The *Abel-Jacobi map* [4, §1] links crucial information of a complex algebraic curve (a Riemann surface) to computational data, namely contour integrals along paths connecting singularities. Computing them rigorously is a major step towards proofs of existence of particular solutions to nonlinear wave equations in physics [2, 10, 9, 18]: KdV (Korteweg-de Vries), KP (Kadomtsev-Petviashvili) and NLS (nonlinear Schrödinger). This also has applications in computer algebra, e.g. integrating algebraic functions [23].

IV. References

[1] S. Basu, R. Pollack, and M.-F. Roy. *Algorithms in Real Algebraic Geometry*. Springer Berlin, Heidelberg, 2003.

[2] E. D. Belokolos, A. I. Bobenko, V. Z. Enolskii, A. R. Its, and V. B. Matveev. *Algebro-geometric approach to nonlinear integrable equations*, volume 1994. Springer, 1994.

[3] C. Beltrán and A. Leykin. Robust certified numerical homotopy tracking. *Found. Comput. Math.*, 13:253–295, 2013.

[4] A. I. Bobenko and C. Klein. *Computational Approach to Riemann Surfaces*, volume 2013. Springer Science & Business Media, 2011.

[5] F. Bréhard. *Certified numerics in function spaces : polynomial approximations meet computer algebra and formal proof*. PhD thesis, University of Lyon, France, 2019.

[6] F. Bréhard, N. Brisebarre, M. Joldes, and W. Tucker. Efficient and validated numerical evaluation of abelian integrals. *ACM Trans. Math. Softw.*, pages 1 – 38, 2023. accepted for publication. `https://hal.science/hal-03561096`.

[7] J. Canny. *The complexity of robot motion planning*. PhD thesis, MIT, 1987.

[8] D. Cox, J. Little, D. O'Shea, and M. Sweedler. *Ideals, varieties, and algorithms*. Springer, 1994.

[9] B. Deconinck and M. S. Patterson. Computing the Abel map. *Physica D: Nonlinear Phenomena*, 237(24):3214–3232, 2008.

[10] B. Deconinck and H. Segur. The KP equation with quasiperiodic initial data. *Physica D: Nonlinear Phenomena*, 123(1):123–152, 1998.

[11] D. N. Diatta, S. Diatta, F. Rouillier, M.-F. Roy, and M. Sagraloff. Bounds for polynomials on algebraic numbers and application to curve topology. *Discrete & Computational Geometry*, 67(3):631–697, 2022.

[12] M. S. E. Din and E. Schost. A nearly optimal algorithm for deciding connectivity queries in smooth and bounded real algebraic sets. *Journal of the ACM*, 63(6), 2017.

[13] J. D. Hauenstein and A. J. Sommese. What is numerical algebraic geometry? *Journal of Symbolic Computation*, 79:499–507, 2017.

[14] M. N. Islam, A. Poteaux, and R. Prébet. Algorithm for connectivity queries on real algebraic curves. In *Proceedings of the 2023 International Symposium on Symbolic and Algebraic Computation*, page 345–353. ACM, 2023.

[15] K. Jin and J. Cheng. On the complexity of computing the topology of real algebraic space curves. *Journal of Systems Science and Complexity*, 34:809–826, 2021.

[16] F. Johansson. Arb: Efficient arbitrary-precision midpoint-radius interval arithmetic. *IEEE Transactions on Computers*, 66(8):1281–1292, 2017.

[17] R. E. Moore, R. B. Kearfott, and M. J. Cloud. *Introduction to Interval Analysis*. Society for Industrial and Applied Mathematics, 2009.

[18] M. S. Patterson. *Algebro-geometric algorithms for integrable systems*. PhD thesis, University of Washington, 2007.

[19] A. Poteaux and M. Rybowicz. Good reduction of Puiseux series and applications. *Journal of Symbolic Computation*, 47(1):32 – 63, 2012.

[20] A. Poteaux and M. Weimann. Computing Puiseux series: a fast divide and conquer algorithm. *Annales Henri Lebesgue*, 4:1061–1102, 2021.

[21] A. Poteaux and M. Weimann. Local polynomial factorisation: improving the Montes algorithm. In *Proceedings of the 2022 International Symposium on Symbolic and Algebraic Computation*, ISSAC '22. ACM, 2022.

[22] A. Sommese and C. Wampler. *The Numerical solution of systems of polynomials arising in engineering and science*. World Scientific, 2005.

[23] B. M. Trager. *Integration of Algebraic Functions*. PhD thesis, MIT, 1984.

[24] W. Tucker. A rigorous ODE solver and Smale's 14th problem. *Found. Comput. Math.*, 2:53–117, 2002.

[25] W. Tucker. *Validated Numerics: A Short Introduction to Rigorous Computations*. Princeton University Press, Princeton, 2011.

[26] J. B. van den Berg and J.-P. Lessard. Rigorous numerics in dynamics. *Notices Amer. Math. Soc.*, 62(9):1057–1061, 2015.