Efficient Algorithm for Estimating Amoebae Using Archimedian Tropical Varieties

Sheridan Grant

July 26, 2013

Abstract

Given any complex Laurent polynomial f, Amoeba(f) is defined as the image of the complex roots of f under the coordinate-wise log absolute value map. The Archimedean tropical variety has been proposed as a means of approximating Amoeba(f), and there is now an explicit bound on the Hausdorff distance between the two sets. We illustrate a polynomial-time sub-algorithm that computes the connected components of ArchTrop(f) for a given query point. We then extend this sub-algorithm to an exponential-time algorithm for approximating the nearest root to a given query point for a system of polynomials and finding all intersections of Archimedean tropical varieties for polynomials of a given system.

1 Introductions

1.1 The Program

Coming into the REU, the only experience I had with Algebraic Geometry was a proof-based introductory linear algebra course. Fortunately, the first two weeks were spent well, by introducing our group to various Algebraic-Geometric concepts. We learned techniques for bounding the number of roots of polynomial systems (for instance, applying induction on a known bound for a single dimension or using Rolle's theorem, calculus and algebra to bound the number of real roots of a polynomial). Geometric concepts like the amoeba (which became integral to our project) and path-connectedness were discussed, among other things.

I chose the Tropical Geometry project as I thought my programming experience would allow me to contribute to the algorithmic aspects of the project, especially as a major goal of the project was to find fast/polynomial time algorithms. Sure enough, Matlab programming became a daily job, allowing me to add a sixth programming language to my lexicon.

We began the project with a "naive" algorithm that used the exponential-time convex hull algorithm to compute the entire Archimedean Newton polytope and, from there, the entire Archimedean tropical variety. Though the algorithm was inefficient, it was a good way to learn Matlab, as well as get a feel for the eccentric geometric structures that we would be working with going forward. Also, the ability to fully depict Archimedean tropical varieties in two dimensions proved useful in testing our later algorithms and in communicating the geometric concepts of our project in presentations. Following this, we transitioned into writing our final algorithm, detailed below.

1.2 Introduction to Algebraic Geometry

The traditional means of solving poynomial systems of equations, particularly with the use of Grobner Bases, unfortunately has a lower complexity bound of exponential space. Polyhedral geometry can be used to approximate the norms of roots of polynomials, a relationship first discovered by Isaac Newton over the field $\mathbb{C}$. We examine tropical geometry, specifically the Archimedean case, as a means of linking polyhedral geometry with algebraic geometry.

1.3 Definitions

We use the abbreviations $[N] := \{1, ..., N\}$, $x := (x_1, ..., x_n)$, and let $\text{Conv}(S)$ denote the convex hull of a set S. Let us then define the function $\text{Log}|x|$ to be $(log|x_1|, ..., log|x_n|)$ and, for any $f \in \mathbb{C}[x_1^{\pm 1}, ..., x_n^{\pm 1}]$, we define $\text{Amoeba}(f)$ to be $\{Log|x| \mid f(x) = 0,\, x \in (\mathbb{C}^*)^n\}$. Also, writing $f(x) = \sum_{i=1}^{t} C_i x^{ai}$ with $C_i \neq 0$ for all i, we define the support (or spectrum) of f to be $\text{Supp}(f) := \{a_i\}_{i \in t}$, the (ordinary) Newton polytope of f to be $\text{Newt}(f) :=\text{Conv}(\text{Supp}(f))$, and the Archimedean Newton polytope of f to be $\text{ArchNewt}(f) :=\text{Conv}(\{(a_i, -\log |c_i|)\}_{i \in t})$. We also define the Archimedean tropical variety of f, denoted

ArchTrop(f), to be the set of all $v \in \mathbb{R}^n$ with $(v, -1)$ an outer normal of a positive-dimensional face of ArchNewt(f). Finally, given any subsets R, $S \in R^n$, their Hausdorff distance, $\Delta(R, S)$, is defined to be the maximum of $\sup\limits_{\rho\in\mathbb{R}} \inf\limits_{\sigma\in S} |\rho-\sigma|$, and $\sup\limits_{\sigma\in S} \inf\limits_{\rho\in\mathbb{R}} |\rho-\sigma|$ where $|.|$ denotes the usual L_2 norm on $\mathbb{R}^n$.

1.4 Theorem

For any $f \in \mathbb{C}[x_1^{\pm 1}, ..., x_n^{\pm 1}]$ with exactly t monomial terms and Newt(f) of dimension k, we have $t \geq k+1$ and

$$\Delta(Amoeba(f), ArchTrop(f)) \leq (2t-3)log(t-1)$$

Furthermore:

(0) If $t = k+1$ or Supp(f) is the vertex set of Newt(f), then (a) ArchTrop(f) $\subseteq$ Amoeba(f) and (b) ArchTrop(f) and Amoeba(f) are homotopy equivalent.

(1) For $t \geq k+1$ we have $\sup\limits_{\rho\in Amoeba(f)} \inf\limits_{\sigma\in ArchTrop(f)} |\rho-\sigma| \leq log(t-1)$.

(2) Defining $g_{t,k}(x) := (x_1+1)^{t-k} + x_2 + \cdots + x_k$, we have

$$\Delta(Amoeba(g_{t,k}), ArchTrop(g_{t,k})) \geq log(t-k)$$

2 Algorithm

An algorithm that calculates the Archimedean tropical variety intersections lying on the connected component of the union of the Archimedean tropical varieties that surrounds a given query point for a system of k n-dimensional polynomial equations is detailed as follows.

Input: For k functions $f \in \mathbb{Q}[x_1, \ldots x_n]$, written as $f(x) = \sum_{i=1}^{t} C_i x^{ai}$ with $C_i \neq 0$ for all i and the a_i distinct, input each Supp(f) concatenated with -log$|C_i|$ for the k polynomials and a query point, $v \in \mathbb{Q}^n$.

Output: A matrix consisting of the intersections of the Archimedean tropical varieties of the k polynomials in the connected component of $\mathbb{R}^n$ surrounding the query point v and a matrix consisting of the distances from the query point, v, to the given ArchTrop intersections.

Description:

1. Using linear programming, determine which vectors $\{a_i, -log|C_i|\}$ in Supp(f), for each of the k polynomials, cannot be written as a convex linear combination of the other vectors in the set. I.e., for any $\{a_k, -log|C_k|\}$, with $k \in [t]$, $\alpha_1(a_i, -log|C_i|)+\cdots+\alpha_n(a_t, -log|C_t|) \neq (a_k, -log|C_k|)$, the vector $(a_i, -\log|c_i)$ does not appear on the left hand side of the equation, and the real numbers α_i satisfy $\alpha_i \geq 0$ and $\alpha_1 + \cdots + \alpha_n = 1$. These vectors are the vertices defining the Archimedean Newton polytope, or ArchNewt, of the function.

2. Now determine which vertex of the ArchNewt(f) yields the maximal inner product with the query point vector, v. Via the duality between ArchNewt(f) and ArchTrop(f) this vertex will correspond to the connected component of the ArchTrop(f) in which the given query point, v, will lie.

3. Using a second linear program, determine which vertices of ArchNewt(f), when paired with the maximizing vertex, define edges of the ArchNewt(f) for each of the k polynomials. Consider the midpoints of all the line segments formed by a vertex other than the maximizing vertex being connected to the maximizing vertex. If a midpoint cannot be written as a convex linear combination of any vertices (other than, trivially, the two vertices it is halfway between), the edge from which it was derived defines an outer edge of the ArchNewt(f). Thus, the outer edges of the hull emanating from the maximizing vertex are found.

4. In a third linear program, determine which edges defining ArchNewt(f) for each of the k polynomials are lower edges. Subtract any $\epsilon > 0$ from the last coordinate of the midpoint of each edge, and minimize a variable s that is added to the last coordinate of the linear combination. If the minimum distance s the midpoint's last coordinate may be raised while still remaining in the hull is less than ϵ, the midpoint originally resided on an upper edge of the hull. If $s = \epsilon$, the midpoint originally resided on a lower edge of the hull. Thus, the lower edges of ArchNewt(f) may be determined.

5. Now, for each of the k polynomials, subtract the maximizing vertex from each vertex vector that helps define a lower edge of ArchNewt(f) in order to obtain the vectors that define the lower edges of each ArchNewt(f) of interest for the query point, v. Compute the inner product of each of these vectors with the vector $X = (x_1, x_2, \dots, x_n, -1)$. Setting the product less than or equal to zero, the equations for the hyperplanes defining the connected component of the ArchTrop(f) in which the query point lies are obtained.

6. Find all k-tuples from the k sets of lower edge vectors in ArchNewt(f) for the k polynomials. Check each n-dimensional k-tuple for linear independence, projecting the vectors "down a dimension" (i.e. ignoring the last coordinate).

7. Determine which of those linearly independent lower edge k-tuples have a Minkowski sum that defines a lower face of the Minkowski sum of the k ArchNewt(f) they help define via a linear program similar to that in step 4.

8. Those k-tuples of edges whose Minkowski sums form lower faces on the Minkowski sum of the k Archimedean Newton Polytopes define intersections of the k Archimedean Tropical Varieties on the connected component of the union of the k Archimedean Tropical Varieties that surrounds the query point. Taking the dot product of each edge in the k-tuple with $(x_1, x_2, \ldots, x_n, -1)$ yields a system of k hyperplanes whose intersection is the intersection of the Archimedean Tropical Varieties we desire. From these intersecting hyperplanes, computing the intersections and (if $k = n$), the distance from v to the intersection points, is trivial.

3 Preliminary Results

3.1 The POSSO Suite

We aim to test the efficiency and accuracy of our algorithm, as well as gain an indication of what the generalization of the Hausdorff bound in section 1.2 to arbitrary dimensions might look like. The POSSO suite provides an excellent opportunity to do this, offering numerous polynomial systems of varying sizes and orders that have been used in research in areas like chemistry and economics.

3.2 Results

For the Reimer 5 and Rose systems, most roots appear to lie less than 2 units from an intersection of the system's corresponding Archimedean tropical varieties. $\sup_{a \in \log|roots|} \inf_{b \in ArchTropIntersections} |a - b|$ initially appears to be relatively small, being 5.3872 for "Chemequs" and 0.5012 for "Caprasse." However, $\sup_{b \in ArchTropIntersections} \inf_{a \in \log|roots|} |a - b|$ seems to be much larger: 111.1693 for "Chemequs" and 5.8575 for "Caprasse."

4 Conclusion

4.1 The Project

The project is far from over. Though we have a working algorithm, we must test it thoroughly, an endeavor that will take lots of time (both real and computational). Additionally, we plan to use our algorithm to develop heuristics for the Hausdorff distance between Archimedean Tropical Varieties and amoebae for systems. We will also explore the algorithm's implications in complexity theory, perhaps specifically in the context of Steve Smale's seventeenth problem, which calls for an average-case polynomial-time algorithm that approximates roots of polynomial systems. In the coming semester, we plan to write and submit for publishing a paper centered on this algorithm with Drs. Maurice Rojas and Peter Gritzmann.

4.2 The Program

The most significant impression the REU made on we was that research is an enjoyable and worthwhile use of time. I have always enjoyed working with others on math, so it was natural that I would like research. I also like getting wrapped-up in whatever I am doing, so having an enormous, difficult problem to always be pondering is great. My feelings about graduate education are more mixed. Over the course of the program, I learned that a graduate mathematics education is extremely important, even if one is aiming to work in industry rather than academia. Yet it seems to be a stressful struggle for many students, making me think that finding a field that one is truly passionate about is vital before attending graduate school. My hope is that, by senior year, I will have discovered such a field.

References

[1] Martín Avendaño, Roman Kogan, Mounir Nisse, and J. Maurice Rojas. *Metric Estimates For Archimedean Amoeba And Tropical Hypersurfaces.*

[2] Eleanor Anthony, Sheridan Grant, J. Maurice Rojas, and Alexander E. Whatley. *Polynomial-Time Ball-Amoeba Intersection.*