Counting Real Roots of Tetranomials Faster via A-discriminants

Lisa Soule

29 July 2021

Abstract

Given a univariate polynomial with degree d and all coefficients having absolute value at most H, there are decades-old algorithms that can count the real roots in time polynomial in d $log(H)$. We show how to count the roots in time polynomial in $log(dH)$, for a large fraction of inputs, when there are just 4 terms. The case of 2 terms is elementary, while the case of 3 terms was only discovered around 2009. For 4 terms, there is a geometric explanation for why real roots are hard to count for a small fraction of input tetranomials. We explore some of this geometry via some Matlab experiments.

1. Introduction

We explore, via some Matlab experiments, how to speed up real root counting on average for univariate tetranomials. A key tool we use is the theory of A-discriminants and tropical deformations [**DFS07**].

There are some key motivations for trying to better understand and work with discriminant varieties. The discriminant variety [**GKZ94**] is the set of coefficients that define a polynomial with degenerate roots and degenerate roots help describe transitions in number of real roots. Closeness to degeneracy governs hardness of numerical solving. The application of these findings touch on a wide swath of areas including algebraic statistics, computational biochemistry and partial differential equations.

Too see just how quickly discriminants can become algebraic nightmares in the tetranomial case, observe the following two examples:

First, we see that using a simple tetranomial:

$$c_0 + c_1 x + c_2 x^2 + c_3 x^3 \tag{1}$$

yields a manageable discriminant:

$$-27c_0^2c_3^2 + 18c_0c_1c_2c_3 - 4c_0c_2^3 - 4c_1^3c_3 + c_1^2c_2^2.$$

But when we summon a slightly more sparse tetranomial:

$$c_0 + c_1 x^3 + c_2 x^5 + c_3 x^{19} \tag{2}$$

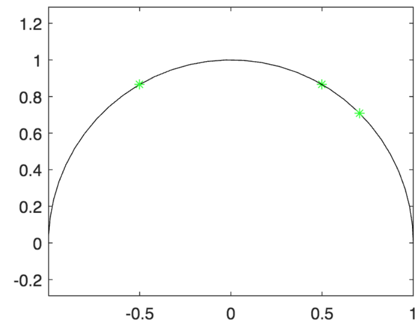
we are left with a nastier result!:

$1978419655660313589123979c_0^{16}c_3^5 + 6093825838807983035604992c_0^{12}c_1^3c_2^2c_3^4 - 4166308590611436407824000c_0^{10}c_1c_2^7c_3^3 + 4136784303514917397331968c_0^8c_1^6c_2^4c_3^3 - 1680626254018160036413444c_0^6c_1^{11}c_2c_3^3 + 5465538956966243292282888c_0^6c_1^4c_2^9c_3^2 + 3040596925589240487608320c_0^4c_1^9c_2^6c_3^2 + 9103573347707241984000c_0^4c_1^2c_2^{14}c_3 + 2441097252432707688857 6c_0^2c_1^{14}c_2^3c_3^2 - 1103132840914428362752c_0^2c_1^7c_2^{11}c_3 + 347250213298688000000c_0^2c_2^{19} + 498062089990157893632c_1^{19}c_3^2 - 48896735641570639872\ c_1^{12}c_2^8c_3 + 12000967371602657 28c_1^5c_2^{16}.$

Looking at this example we can see that we need better way to plot the zero sets of complicated polynomials! We will use the clever Horn-Kapranov Uniformization (HKU) to reduce the dimension of the parameter space!

Coding the HKU into Matlab takes a polynomial and runs it through two loops. These loops essentially turn our unmanageable discriminants into a nice summation of logs which are much simpler to work with.

If we change the input points $(\lambda_1 : \lambda_2)$ to equal $(cos\theta, sin\theta)$ this brings our plots from $[\lambda_1 : \lambda_2]$ in the projecive aspace $\mathbb{P}^1_{\mathbb{R}}$ to the unit semi-circle. This provides a clear visual of the coefficient space of our polynomial's discriminant:



Simplifying our visual even further, we can transform the semi-circle into an amoeba by using logs of the polynomial coefficients [**Kap91**]. The image of the semicircle gives green points where asymptotes lie in the amoeba. We can now look at each chamber of discriminant coefficient space more effectively:

Figure: This is the Ameoba for $1 + x_1 + x_2$.

While the amoeba shape is a cool-looking shape it is not trivial to draw. Deciding if a rational point (x,y) lies in a 2-dimensional amoeba is already NP-hard while deciding if a rational point lies on or near such a curve gets us into interesting problems involving Diophantine approximation!

The aim is to then approximate the curves of the amoeba with tropical approximations. Once an appropriate approximation has been made, those curves will be used as inequalities to test for certain "points" in the amoeba. These points will be specific polynomials that will land in the quadrant that contains the correct number of real roots akin to it.

The Matlab code runs through a version of the HKU to plot the amoeba and its quadrants. Each quadrant is also a sign chamber, signaling the sign of the coefficients in ascending order of exponents. There are four chambers for our tetranomial despite sixteen possible combinations of signs for four terms. The reason is because of certain homogeneities. First, multiplying f(x) by -1 will not change the number of real roots. This gets rid of half of the possibilities. The second is that the number of real roots is also not affected by changing x to -x, so we lose another half of the entries, leaving us with four possible sign chambers and thus four quadrants.

Examining each quadrant, specific coefficients can be given to the polynomial to see where it is plotted in the coefficient space. The point is plotted with logs of the absolute value of the coefficients, so it is important to remember the signs to keep track of the proper chamber for each specified polynomial.

Running through each quadrant, one can easily find a set of coefficients that would put the chosen polynomial on both sides of each quadrant. These specific polynomials can then be put into Maple to determine the number of real roots. With this information in hand, now any future polynomial of the same family can have the number of real roots determined by merely plotting it as a point in the amoeba and seeing where it lies in its associated quadrant!

---

2. Key Techniques

The following definitions and theorems will help us along the way:

Definition 1 (A-Discriminant Variety). *For* $A = \begin{bmatrix} a_1 & a_2 & a_3 & a_4 \end{bmatrix}$ *we set* ∇_A *to be the set of all* $[c_1 : c_2 : c_3 : c_4]$ *in* $\mathbb{P}^3_{\mathbb{C}}$ *such that* $c_1x^{a_1} + ... + c_4x^{a_4}$ *has a degenerate root in* $\mathbb{P}^1_{\mathbb{C}}$*. Note: The intersection of* ∇_A *with the line* $c_1 = c_4 = 1$ *in* $\mathbb{P}^3_{\mathbb{C}}$ *is then just the set of all* $[1 : c_2 : c_3 : 1]$ *such that there is a* z *in* $\mathbb{C}$ *with* $f(z) = z^{a_1} + c_2z^{a_2} + c_3z^{a_3} + z^{a_4} = 0$ *and* $f'(z) = 0$*.*

Definition 2 (A-Discriminant Polynomial). *For* $A = \begin{bmatrix} 0 & a_2 & a_3 & a_4 \end{bmatrix}$ *(so we assume* $a_1 = 0$*) we define* Δ_A *to be* $(1/c_4)^{a_4-a_3}$ *times the determinant of* *the Sylvester matrix of format* $(a_4, a_4 - a_2)$ *corresponding to the polynomials* f *and* f'/x^{a_2-1}*, i.e.,* Δ_A *is the determinant of an explicit structured* $(2a_4 - a_2)$ *by* $(2a_4 - a_2)$ *matrix with entries that are either 0 or a coefficient of* f*.*

Definition 3 (Horn-Kapranov Uniformization)**.** A way to efficiently parameterize discriminant varieties. *For* $A = \begin{bmatrix} a_1 & a_2 & a_3 & a_4 \end{bmatrix}$*, let* $\hat{A}$ *be the 2x4 matrix defined by appending a row of 1s to the top of* A *and let* B *in* $\mathbb{Z}^{4x2}$ *be any matrix whose columns form a basis for the right nullspace of* $\hat{A}$*. Then the (logarithmic, reduced) Horn-Kapranov Uniformization for* A *is the function*

$$\xi_A([\lambda_1 : \lambda_2]) := (Log|[\lambda_1, \lambda_2]B^T|)B$$

which defines a map from $\mathbb{P}^1_{\mathbb{R}}$ *to* $\mathbb{R}^2$*.*

Definition 4 (Amoeba)**.** f is any polynomial in $C[x_1, \ldots, x_n]$ then its amoeba is the set

$$\{(\log|x_1|, \ldots, \log|x_n|) \mid f(x_1, \ldots, x_n) = 0 \ , \ x_i \in \mathbb{C} \setminus \{0\}\}.$$

Theorem 1. *If* $A = \begin{bmatrix} a_1 & a_2 & a_3 & a_4 \end{bmatrix}$ *then the A-discriminant variety is exactly the zero set of the A-discriminant polynomial in* $\mathbb{P}^3_{\mathbb{C}}$*.*

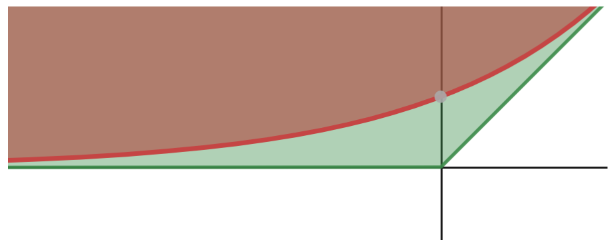
Theorem 2. *If* $A = \begin{bmatrix} a_1 & a_2 & a_3 & a_4 \end{bmatrix}$ *then the resulting reduced A-discriminant contour is exactly the image of* $\mathbb{P}^1_{\mathbb{R}}$ *under the (reduced, logarithmic) Horn-Kapranov Uniformization,* ξ_A*.*

---

3. Results

The first part of the Matlab code will check to see if the tropical approximation is close enough to the amoeba curve to yield an accurate result. Choosing a simple, linear piece-wise approximation is the first step.

Here, we look at how well the tropical, linear piece-wise function $y >= 0$ and $y >= x$ (in green) approximates the known curve of an amoeba. This curve is $y >= log(1 + e^x)$ (in red) and lies in projective space, but brought into the reals to be seen in two dimensions, $\mathbb{P}^1_{\mathbb{R}} \to \mathbb{R}^2$:



Testing random points for accuracy of the approximation leaves us underwhelmed:

Testing 1000 i.i.d. random points:

Trials:	1	2	3	4	5
%:	62%	65%	63%	60%	65%

Testing 10,000 i.i.d. random points:

Trials:	1	2	3	4	5
%:	63%	63%	65%	64%	64%

Testing 100,000 i.i.d. random points:

Trials:	1	2	3	4	5
%:	64%	63%	64%	64%	64%

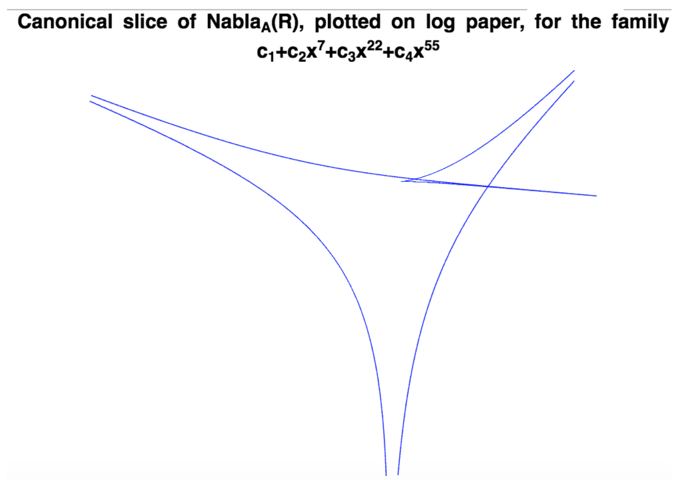
Deciding a polynomial inequality, involving a polynomial of degree d in n variables with coefficients all of absolute value $<= H$, at an input rational point $p = (a_1/b_1, ..., a_n/b_n)$, is a highly non-trivial problem! So what can be done to make a more accurate approximation? Since the arcs of the discriminant contour are defined by linear combinations of logarithms, each arc can be approximated by just 2 logarithms. This should also yield easier Diophantine approximation.

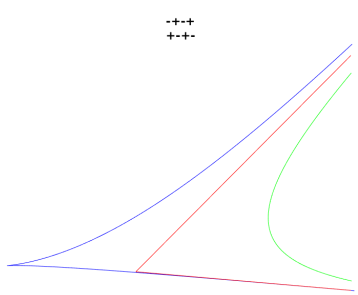
Testing this new approximation for accuracy could be done in future research with more time, but looking at the images produced in the Matlab code will show just how much closer this new curve approximates the contour of the amoeba.

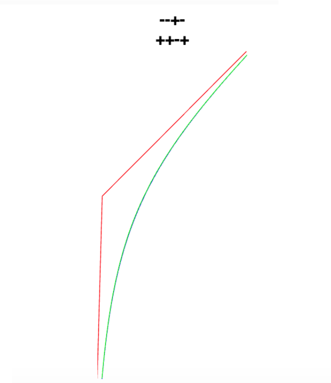
The second part of the Matlab code will take in exponent values, run through two loops to apply the HKU, and produce the associated amoeba to the polynomial along with each quadrant. The family of polynomials that will be used as the example is:

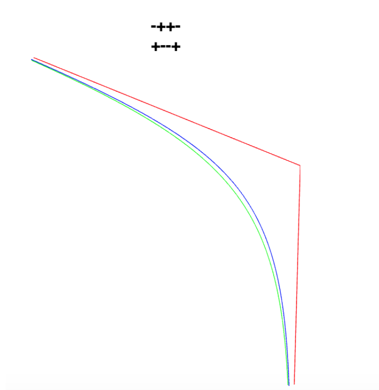
$$c_1 + c_2x^7 + c_3x^{22} + c_4x^{55}.$$

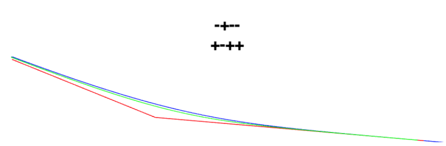
This produces the following amoeba and its four quadrants:





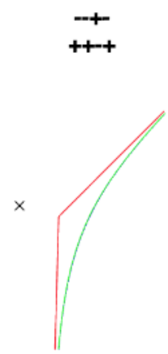






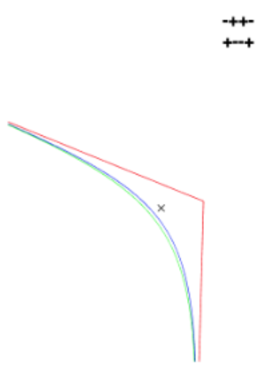
Each of the four quadrants is labeled with the signs relating to the coefficients of the polynomials within the example family. The amoeba curves are in blue while the linear approximations are in red and the newer, logarithmic approximations in green. In all quadrants, except for the first one, the green curve approximates the discriminant contours more accurately. The less accurate approximation of the first quadrant, however, will not negatively affect the results. Points can still be checked using the same inequalities and can be visually spotted within the targeted region to determine the number of real roots.

Now, coefficients can be chosen to plot specific polynomials as points in the quadrants. It will be pertinent to hit both sides of the curve with a point:

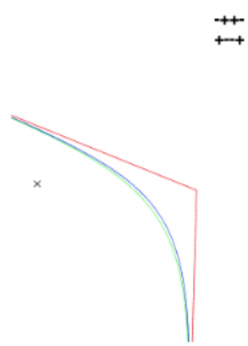


$-1 - x^7 + x^{22} - x^{55}$

$-2 - 2x^7 + 10x^{22} - x^{55}$



$-1 + x^7 + x^{22} - x^{55}$



$-20 + 10x^7 + 2x^{22} - 10x^{55}$

These polynomials are part of the example of family of polynomials with specific coefficients that give them a particular point in the quadrants. Now, these polynomials can be plugged into a computer algebra system, such as Maple, where the number of real roots can easily be calculated:

$$f1 := -1 - x^7 + x^{22} - x^{55};$$

$$realroot(f1);$$

$$f2 := -2 - 2 \cdot x^7 + 10 \cdot x^{22} - x^{55};$$

$$realroot(f2);$$

$$f3 := -1 + x^7 + x^{22} - x^{55};$$

$$realroot(f3);$$

$$f4 := -10 + 2x^7 + 10x^{22} - 20x^{55};$$

$$realroot(f4)$$

$$f1 := -x^{55} + x^{22} - x^7 - 1$$

$$\left[\left[-\frac{30953}{32768}, -\frac{61903}{65536}\right]\right]$$

$$f2 := -x^{55} + 10x^{22} - 2x^7 - 2$$

$$\left[\left[-\frac{118191}{131072}, -\frac{472761}{524288}\right], \left[\frac{61}{64}, \frac{977}{1024}\right], \left[\frac{139993}{131072}, \frac{279989}{262144}\right]\right]$$

$$f3 := -x^{55} + x^{22} + x^7 - 1$$

$$\left[[-1, -1], \left[\frac{125029}{131072}, \frac{250335}{262144}\right], [1, 1]\right]$$

$$f4 := -20x^{55} + 10x^{22} + 2x^7 - 10$$

$$\left[\left[-\frac{1001}{1024}, -\frac{125}{128}\right]\right]$$

Observing these results shows the number of real roots for the tested polynomials:

- $-1 - x + x^{22} - x^{55}$ has 1 negative root
- $-2 - 2x + 10x^{22} - x^{55}$ has 2 positive roots and 1 negative root
- $-2 - 2x + 10x^{22} - x^{55}$ has 2 positive roots and 1 negative root
  $-1 + x + x^{22} - x^{55}$ has 2 positive roots and 1 negative root
- $-20 + 10x + 2x^{22} - 10x^{55}$ has 1 negative root

This process can be done with the remaining two quadrants to label the coefficient space of the entire amoeba. With this information, any future polynomial with any coefficients within the family can be easily plotted to instantly know the number of real roots that specific polynomial has.

Acknowledgments

I would like to express my deepest gratitude to Dr. Maurice Rojas and TA Joshua Goldstein for providing me with tireless support throughout this entire process. I would like to individually thank Dr. Rojas for providing me with the Matlab code on which this research is based.
I also would like to extend appreciation to TAMU and the NSF who made this research experience possible.

References

[DFS07] Dickenstein, Alicia; Feichtner, Eva Maria; and Sturmfels, Bernd, *"Tropical discriminants,"* J. Amer. Math. Soc., **20** (2007), pp. 1111–1133.

[GKZ94] Gel'fand, Israel Moseyevitch; Kapranov, Misha M.; and Zelevinsky, Andrei V.; *Discriminants, Resultants and Multidimensional Determinants,* Birkhäuser, Boston, 1994.

[Kap91] Kapranov, Misha, *"A characterization of A-discriminantal hypersurfaces in terms of the logarithmic Gauss map,"* Mathematische Annalen, 290, 1991, pp. 277–285.