Polyhedral Homotopies

Polyhedral homotopies provide proof that mixed volumes count the roots of random coefficient polynomial systems. Mixed-cell configurations store the supports of all start systems in polyhedral homotopies. Our application concerns the construction of Runge-Kutta formulas and we end by sketching Viro’s method.

1 Bernstein’s First Theorem

We consider a system \( f(x) = 0 \) of \( n \) equations \( f = (f_1, f_2, \ldots, f_n) \) in \( n \) unknowns \( x = (x_1, x_2, \ldots, x_n) \) and complex coefficients. The sparse structure of the system is modeled by its supports \( A = (A_1, A_2, \ldots, A_n) \), so we write

\[
  f_i(x) = \sum_{a \in A_i} c_{ia} x^a, \quad c_{ia} \in \mathbb{C}^*, \quad \mathbb{C}^* = \mathbb{C} \setminus \{0\}, \quad i = 1, 2, \ldots, n. \tag{1}
\]

The supports \( A \) span the Newton polytopes \( P = (P_1, P_2, \ldots, P_n) \), \( P_i = \text{conv}(A_i) \), \( P_i \) is the Newton polytope of \( f_i \), \( i = 1, 2, \ldots, n \). We denote the mixed volume of \( P \) by \( V(P) \).

**Theorem 1.1 (Bernstein [2])** The number of isolated solutions of \( f(x) = 0 \) in \((\mathbb{C}^*)^n\) is bounded by \( V(P) \).

The proof of this theorem as given in [2] is constructive, i.e.: we can implement it as an algorithm to solve polynomial systems. Following the more general [6], we outline the proof via polyhedral homotopies. In [9] another self-contained proof and introduction to polyhedral methods is given.

Let \( g(x) = 0 \) be a generic system with the same Newton polytopes as \( f(x) = 0 \). The polynomials in \( g \) are generic in the sense that their coefficients are randomly chosen complex numbers, typically homotopy

\[
h(x, t) = (1 - t)g(x) + tf(x) = 0, \quad t \in [0, 1]. \tag{2}
\]

If we can show that \( g(x) = 0 \) has exactly as many solutions in \((\mathbb{C}^*)^n\) as \( V(P) \), then the observation that (2) is a coefficient-parameter homotopy proves the theorem.

That \( g(x) = 0 \) has exactly as many solution in \((\mathbb{C}^*)^n\) as \( V(P) \) follows from the application of Bernstein’s second theorem. For any nonzero direction \( v \in \mathbb{Z}^n \), the initial form system \( \text{in}_v g(x) = 0 \) is an overdetermined polynomial system. Since all monomials \( x^a \) of \( \text{in}_v g \) yield the same value \( \langle a, v \rangle \), we may define a unimodular coordinate transformation which eliminates one variable. After this transformation, we are left with \( n \) equations in \( n - 1 \) unknowns in a system whose coefficients were chosen at random. Such an overdetermined random coefficient system has no solutions.

We will solve \( g(x) = 0 \) via polyhedral homotopies, defined via the application of a lifting function \( \omega = (\omega_1, \omega_2, \ldots, \omega_n) \) on the supports. This leads to the system \( \hat{g}(x, t) \) with equations

\[
  \hat{g}_i(x, t) = \sum_{a \in A_i} \hat{c}_{ia} x^a t^{\omega_i(a)}, \quad \hat{c}_{ia} \in \mathbb{C}^*, \tag{3}
\]

where the coefficients \( \hat{c}_{ia} \) are random complex numbers. The system \( \hat{g}(x, t) = 0 \) is a system of \( n \) equations in \( n + 1 \) variables. To solve \( \hat{g}(x, t) = 0 \), we look for inner normals \( v = (u, 1) \) for which the corresponding initial form system \( \text{in}_v \hat{g}(x, t) = 0 \) has a solution in \((\mathbb{C}^*)^n\). Change coordinates as follows: \( x_j = y_j t^{u_j}, \ j = 1, 2, \ldots, n \), then \( \hat{g}_i(y, t) \)

\[
  = \sum_{a \in A_i} \hat{c}_{ia} (y_1 t^{u_1} y_2 t^{u_2} \cdots y_n t^{u_n})^a t^{\omega_i(a)} = \sum_{a \in A_i} \hat{c}_{ia} y^a t^{u_1 a_1 + u_2 a_2 + \cdots + u_n a_n + \omega_i(a)} = \sum_{a \in A_i} \hat{c}_{ia} y^a t^{(a, u) + \omega_i(a)}. \tag{4}
\]

Denoting the result of this coordinate change as \( \tilde{g}_v(y, t) = 0 \), and \( m_i = \min_{a \in A_i}(a, v) \), then monomials of \( \tilde{g}_v \) with lowest exponent \( m_i \) belong to \( \text{in}_v \tilde{g}_v \). Thus \((t^{-m_i} \tilde{g}_v, i)(y, 0) = \text{in}_v (t^{-m_i} \tilde{g}_v, i)(y) \). So initial forms of \( \tilde{g}(x, t) \) are start systems. The \( v \) are the leading powers of Puiseux series [1].
2 Runge-Kutta Formulas

To solve an initial value problem \( y'(t) = f(t, y(t)) \) with \( y(t_0) = y_0 \), Runge-Kutta formulas are widely used numerical methods to approximate the solution \( y(t) \), at discrete time steps \( t_{k+1} = t_k + h_k, \ k = 0, 1, \ldots \) 

A Runge-Kutta formula of order \( s \) has the form

\[
y_{k+1} = y_k + h \sum_{i=1}^{s} b_i f_i \approx y(t_{k+1}) = y(t_k + h_k),
\]

with

\[
f_i = f \left( t_k + c_i h, y_k + \sum_{j=1}^{s} a_{ij} f_j \right), \quad i = 1, 2, \ldots, s.
\]

The coefficients \( c_i \) and \( a_{ij} \) are determined so that the approximation \( y_{k+1} \) in (5) for \( y(t_{k+1}) \) matches the Taylor expansion of \( y(t_{k+1}) \).

For \( s = 2 \), we have the following Runge-Kutta formula:

\[
\begin{align*}
y_{k+1} &= y_k + b_1 k_1 + b_2 k_2 \\
k_1 &= hf(t_k, y_k) \\
k_2 &= hf(t_k + \alpha h, y_k + \beta k_1),
\end{align*}
\]

using a fixed step size \( h \). We can then derive

\[
b_1 + b_2 = 1, \quad \alpha b_2 = \frac{1}{2}, \quad \beta b_1 = \frac{1}{2},
\]

which defines a whole family of formulas with local error \( O(h^2) \). Any member in the family defines a second-order Runge-Kutta method.

See [3] for a symbolic derivation of conditions on Runge-Kutta formulas. Note that we are interested only in those solutions for which all coordinates are nonzero.

3 Regular Mixed-Cell Configurations

A mixed-cell configuration collects the supports of all start systems in the polyhedral homotopies \( \hat{g}(x, t) = 0 \) used to solve a random coefficient system \( g(x) = 0 \).

Because we are interested only in solutions in \((C^\ast)^n\) only those faces \( \partial_v \hat{g} \) that have solutions with all their components different from zero are interesting to us. This condition imposes conditions on the supports of \( \partial_v \hat{g} \). In particular, every equation of \( \partial_v \hat{g} \) must have at least two monomials. The exponent of those two monomials in \( \partial_v \hat{g} \) span an edge on the lower hull of \( \partial_v \hat{P} \), for \( i = 1, 2, \ldots, n \).

Denote the edge which spans \( \partial_v \hat{P} \) by \( \partial_v \hat{A}_i = \{\hat{a}, \hat{b}\} \). Then the inner normal \( v \) to this edge satisfies

\[
\begin{align*}
\langle \hat{a}, v \rangle &= \langle \hat{b}, v \rangle \\
\langle \hat{a}, v \rangle &\leq \langle \hat{c}, v \rangle, \quad \text{for all } c \in A_i.
\end{align*}
\]

Enumerating all edges of a polytope is thus equivalent to enumerating all feasible solutions to the system (9). Letting \( i \) range from 1 to \( n \) in (9) applied to the lifted point sets \( \hat{A}_i \) provides the dual linear-programming model to enumerate all inner normals to the mixed cells in a regular mixed subdivision. We call this collection of mixed cells a mixed-cell configuration.

Figure 1 illustrates the computation of the mixed cells of two polygons \( P_1 \) and \( P_2 \) as the intersection of the outward oriented normal cones to their edges. The supports \( A_1 \) and \( A_2 \) spanning respectively \( P_1 \) and \( P_2 \) are
\[ A_1 = \{(2, 2), (1, 2), (2, 1), (0, 1), (1, 0), (0, 0)\} \]
\[ A_2 = \{(2, 3), (1, 3), (3, 2), (0, 2), (3, 1), (1, 1), (2, 0), (1, 0)\}. \]

The arrows in Figure 1 that are perpendicular to the edges of the polygons represent the outer normal cones to the edges. These cones contain all those vectors for which the inner product with every point on an edge is maximal, compared to any other point of the polygon.

Figure 1: Mixed area computation for two polygons \( P_1 \) and \( P_2 \). The positioning of \( P_1 \) relative to \( P_2 \) emphasizes the intersection of the outer normal cones to the edges. There is a one-to-one correspondence between the intersections at the left and the mixed cells in the mixed subdivision of \( P_1 + P_2 \) at the right.

Figure 1 is a projection of a three dimensional picture, shown in Figure 2.

Figure 2: A regular mixed subdivision of two polygons \( P_1 \) and \( P_2 \). The edges of \( P_2 \) are thicker than those of \( P_1 \). \( P_2 \) stayed at height 0, while \( P_1 \) was lifted using the function \( \omega : (a_1, a_2) \mapsto 4 - a_1/2 - 3a_2/2 \).

If the lifting function \( \omega \) is sufficiently generic, then the conditions on the inner normal \( \nu \) imposed by any edge will determine \( \nu \) uniquely. In this case the inequalities \( \leq \) in (9) will become \( < \) for all \( c \in A_i \setminus \{a, b\} \). Thus for generic lifting functions, the support \( \partial_{\nu}A \) of \( \partial_{\nu}\hat{g} \) consists of two points in every support set. Such systems \( \partial_{\nu}\hat{g} \) are called binomial systems and can be solved very efficiently. Linear programming plays an
important role to compute all mixed cells [8]. MixedVol [4] is an efficient software package to calculate mixed volumes.

The proof of Bernstein’s first theorem relies on Puiseux series and applies the argument of the second theorem. In the polyhedral homotopy, as we let \( t \) go to zero, we keep in mind that we are interested only in the solutions in \((\mathbb{C}^*)^n\). Just like in Bernstein’s second theorem, solutions in \((\mathbb{C}^*)^n\) can occur only at systems supported on at least two monomials in every equation. Computing all mixed cells in a regular mixed subdivision leads to finding all binomial systems to start the deformations.

4 Patchworking Algebraic Curves

Mixed subdivisions were originally introduced by Bernd Sturmfels [12] to generalize Viro’s method to complete intersections, see also [13]. Viro’s method [7] is a deformation method to construct algebraic curves with a topology prescribed by the Newton polytope of the defining polynomial equation.

Consider any regular triangulation of the Newton polygon. We mark the edges which connect vertices of the corresponding coefficient. The polygon is transferred to the other quadrants by flipping over the coordinate axes and flipping the signs for the corresponding odd exponents if necessary. Consider any regular triangulation of the Newton polygon. We mark the edges which connect vertices of opposite sign. A line is drawn between the center of each marked edged and the barycenter of the cell its belongs to. These lines provide a piecewise linear sketch of a planar curve. A polyhedral homotopy with small values for \( t \) will realize the topology of this planar curve.

Figure 3 shows three examples of sign assignments and Figure 4 shows the patchworking.

![Figure 3: Sign assignments to the vertices of a triangle with corresponding piecewise-linear curve.](image)

![Figure 4: At the left we see a triangulation with signs and piecewise-linear curve in the positive real quadrant. Then we flip the left triangulation. The constructed piecewise-linear curve is at the right.](image)

The polynomial that defines the algebraic curve shown in Figure 4 is

\[
    f(x, y) = +c_{2,0}tx^2 + c_{0,2}y^2 - c_{1,0}x - c_{0,1}y + c_{0,0}t, \quad c_{i,j} > 0, \quad t = \epsilon.
\] (12)

The \( t = \epsilon \) means that \( t \) is small enough to realize the topology prescribed by the signed Newton polygon.
5  Newton polytopes in polymake

We can visualize Newton polytopes with the software polymake [5]. Below is a session with version 2.9.9.

```
polytope > $r = new Ring(qw(x y t));
polytope > ($x,$y,$t) = $r->variables;
polytope > $f = $t*$x*$x + $y*$y - $x - $y + $t;
polytope > print($f);
-1*x -1*y + t + y^2 + x^2*t
polytope > $p = newton($f);
polytope > print $p->VERTICES;
polymake: used package cddlib
Implementation of the double description method of Motzkin et al.
Copyright by Komei Fukuda.
http://www.ifor.math.ethz.ch/~fukuda/cdd_home/cdd.html
1 1 0 0
1 0 1 0
1 0 0 1
1 0 2 0
1 2 0 1
polytope > $p->VISUAL;
```

6  a Totem Pole of Homotopies

Figure 5 shows a hierarchy of homotopies. This figure is a simplification of [11, Fig 8.1] and is published in [10]. Except for the polynomial products, we have covered all methods. The polynomials in a start system of the polynomial products class are products of polynomials. Unlike linear-product start systems, polynomial-product start systems are harder to setup and solve.

![Totem Pole Diagram](image)

Figure 5: A totem pole of homotopies. All homotopies below the dashed line A can be done automatically. Above the line, apply special ad-hoc methods or bootstrapping. Homotopies at the bottom of the totem pole are often used to find solutions for generic instances of parameters in a coefficient-parameter homotopy.

One more specific form of coefficient-parameter polynomial continuation is called the cheater’s homotopy.
The name cheater refers to the requirement of having first solutions of a system with random coefficients before the path following can start. Polyhedral homotopies have removed this part of the cheating.

7 Exercises

1. Derive the conditions for the third and fourth order Runge-Kutta formulas, i.e.: $s = 3$ and $s = 4$.

2. Consider the polynomial system 

$$f(x) = \begin{cases} 
  x_1 x_2 + x_1 + x_2 + 1 = 0 \\
  x_1^2 x_2^2 + x_1 + x_2 = 0.
\end{cases}$$

Choose a lifting of the points in the supports of $f$ and compute its mixed-cell configuration. List all homotopies and start systems used to solve a system $g$ with the same supports as $f$ but with random coefficients.

3. Give the system of the previous exercise as input to `phc -m`. Compute a mixed-cell configuration.

4. Consider the polygons in Figure 1. Select one intersection of outward pointing normal vectors and setup the corresponding system of linear inequalities. Compute the inner normal to the corresponding mixed cell of the regular mixed subdivision induced by the lifting used in Figure 2.

References


