Communications in Information and Systems
Volume 15, Number 2, 119–307, 2015

Homotopy continuation method for solving systems of nonlinear and polynomial equations

Tianran Chen and Tien-Yien Li

1. Introduction

The problem of solving a system of n nonlinear equations in n variables $F(\mathbf{x}) = \mathbf{0}$ numerically where

$$F : \mathbb{R}^n \to \mathbb{R}^n$$

appears widely in scientific computations. We assume F is smooth, i.e., it has as many continuous partial derivatives as the discussion requires. A well known algorithm for this problem is the Newton's iterations:

$$\mathbf{x}^{(n+1)} = \mathbf{x}^{(n)} - DF(\mathbf{x}^{(n)})^{-1}F(\mathbf{x}^{(n)}), \quad n = 0, 1, 2, \ldots, \quad \mathbf{x}^{(0)} \in \mathbb{R}^n$$

where $DF(\mathbf{x}^{(n)})$ is the Jacobian matrix of F at $\mathbf{x}^{(n)}$. This algorithm is *local* in the sense that a very good estimate of the correct solution is required for the convergence of the algorithm. Unfortunately, such knowledge concerning zero points of F is usually unavailable *a priorily*. As a possible remedy, one may define a *homotopy function* $H(\mathbf{x}, t) : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^n$ so that

$$H(\mathbf{x}, 0) = G(\mathbf{x}), \qquad H(\mathbf{x}, 1) = F(\mathbf{x}),$$

where $G : \mathbb{R}^n \to \mathbb{R}^n$ is a smooth map whose zeros can easily be obtained, and H is also smooth (in both $\mathbf{x}$ and t). Let $\mathbf{x}_0$ be a solution of $G(\mathbf{x}) = \mathbf{0}$. Then at $t = 0$, $H(\mathbf{x}, 0) = \mathbf{0}$ has a solution $\mathbf{x} = \mathbf{x}_0$. At $t = 1$, the system $H(\mathbf{x}, 1) = \mathbf{0}$ agrees with $F(\mathbf{x}) = \mathbf{0}$. If for arbitrary $t \in \mathbb{R}$, $\mathbf{x}(t)$ solves $H(\mathbf{x}(t), t) = \mathbf{0}$, then, under certain mild conditions, $\mathbf{x}(t)$ will generate a smooth curve. If one can successfully trace this smooth curve $\mathbf{x}(t)$ from $t = 0$ where $\mathbf{x}(0) = \mathbf{x}_0$ continuously, then when t reaches 1, a solution $\mathbf{x}(1) = \mathbf{x}^*$ of $F(\mathbf{x}) = \mathbf{0}$ is attained.

Under this framework, this method, the so-called *homotopy continuation method*, has been substantially developed during the last few decades and proved to be a reliable and efficient numerical algorithm for solving solutions of nonlinear systems of equations numerically. It has become an important tool for this problem as illustrated by the extensive bibliography listed in [4]. Most importantly, this method is *global* in the sense that solutions of $G(\mathbf{x}) = \mathbf{0}$ may not need to be anywhere close to the solution of $F(\mathbf{x}) = \mathbf{0}$.

In this article, a special category of nonlinear systems we choose to deal with by employing the homotopy continuation method is the systems of polynomial equations. The problem of solving polynomial systems has been, and will continue to be, one of the most important subjects in both pure and applied mathematics. The need to solve systems of polynomial equations arises very frequently in various fields of science and engineering, such as, formula construction, geometric intersection, inverse kinematics, robotics, vision and the computation of equilibrium states of chemical reaction equations, etc. Many of those applications have been well documented in [107]. Solving polynomial systems is an area where numerical computations arise almost naturally. Given the complexity of the problem, we must

use standard machine arithmetic to develop efficient algorithms. Moreover, by Galois theory explicit formulas for the solutions are unlikely to exist. We are concerned with the robustness of our methods and want to be sure that *all* isolated solutions are obtained, i.e., we want exhaustive methods. These criteria are met by homotopy continuation methods. In 1977, Garcia and Zangwill [33] and Drexler [25] independently presented theorems suggesting that homotopy continuation methods could be used to find the full set of isolated solutions of n polynomial equations in n variables numerically. Afterwards, with few decades of developments, this method has advanced to a most powerful and widely used procedure in approximating *all* isolated zeros of a polynomial system. Along the way, a new discipline *Numerical Algebraic Geometry* [102] has emerged.

While the natural setting for studying polynomial systems is the product of complex (or projective) spaces, in practice, polynomial systems almost always appear with real coefficients, and most importantly, only real solutions are on the wish list. One may, of course, find *all* solutions in the complex setting first, and then filter out all the non-real solutions. However, to deal with those systems in real spaces directly would certainly be beneficial numerically. In this article, we will pay a special attention in solving *real* polynomial systems by real homotopies. Indeed, we shall introduce solving nonlinear systems of real equations by the real homotopies in the first place.

There is no shortage of the demand of solving larger and larger polynomial systems in applications. To attain more computing resources for solving large polynomial systems, the parallelization of the homotopy method becomes inevitably essential. The biggest advantage of the homotopy continuation method in solving polynomial systems is perhaps its natural parallelism in the sense that each isolated zero is computed totally independent of the others.

The landscape of computation hardware has seen an extremely active development in recent years making a wide spectrum of exciting new technologies available. First, developments in new processor design and network technology have allowed supercomputers and computer clusters to grow larger and faster than ever. Second, new ideas such as cycle-scavenging and grid computing has led to the creation of virtual supercomputers out of large numbers of individual computers around the globe. Another exciting development is the advent of parallel computing on GPUs (Graphical Processing Units). While originally designed for rendering graphics rendering only, over the years GPUs has become sufficiently sophisticated to handle a much wider range of problems. Highly parallel by design, GPUs are more efficient than

general purpose CPUs in carrying out a range of complex algorithms. In this article we will also describe the adaptation of homotopy continuation algorithms to a variety of parallel computation environments. Of course, we can only present the most current parallel computing technologies for solving very large polynomial systems. While the specific details of these technologies can be volatile and are likely to change in the near future, we hope the general idea to remain valid.

2. Curve tracing

The idea of tracing a solution curve of a homotopy equation to reach a solution of a nonlinear system asked to be solved underlines this entire article. We therefore start by establishing its theoretical framework.

Let $F : \mathbb{R}^n \to \mathbb{R}^p$ be smooth. A point $\mathbf{y} \in \mathbb{R}^p$ is said to be a *regular value* of F if

$$\text{Range}\, DF(\mathbf{x}) = \mathbb{R}^p \quad \text{for all } \mathbf{x} \in F^{-1}(\{\mathbf{y}\}) = \{\mathbf{x} \in \mathbb{R}^n \mid F(\mathbf{x}) = \mathbf{y}\}$$

where $DF(\mathbf{x})$ denotes the $n \times p$ Jacobian matrix of F, consisting of all the partial derivatives of $F(\mathbf{x})$ with respect to $\mathbf{x} = (x_1, \ldots, x_n)$. Sard's Theorem [94] states that if F is smooth, then almost all $\mathbf{y} \in \mathbb{R}^p$ are regular values.

In the context of our homotopy constructions, a differentiable homotopy $H(\mathbf{x}, t) : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^n$ with

$$H(\mathbf{x}, 0) = G(\mathbf{x}) \quad \text{and} \quad H(\mathbf{x}, 1) = F(\mathbf{x}),$$

can be thought of as a deformation between two systems $G(\mathbf{x})$ and $F(\mathbf{x})$ among a family of systems. The point $\mathbf{0} = (0, \ldots, 0) \in \mathbb{R}^n$ is a regular value for H if the $n \times (n+1)$ Jacobian matrix DH of H with respect to $(\mathbf{x}, t) = (x_1, \ldots, x_n, t)$ is of rank n (full row rank) for all $(\mathbf{x}, t) \in H^{-1}(\{\mathbf{0}\}) \subset \mathbb{R}^n \times \mathbb{R}$. This mild condition enables the "continuation" of a solution of a single system in the deformation into a solution curve.

2.1. The continuation of solution curves

Assuming $\mathbf{0}$ is a regular value of H, if a solution $\mathbf{x}_0$ to the system $G(\mathbf{x}) = H(\mathbf{x}, 0) = \mathbf{0}$ is known then the point $(\mathbf{x}_0, 0)$ is in $H^{-1}(\{\mathbf{0}\})$ and hence, by assumption, DH is of rank n at this point. By the Implicit Function Theorem, locally (i.e., in a neighborhood $(\mathbf{x}_0, 0)$) $H^{-1}(\{\mathbf{0}\})$ consists of a segment of a smooth curve (however short) that passes through $(\mathbf{x}_0, 0)$. Actually this

segment of curve cannot terminate: by continuity, any limit point of this segment must also be inside $H^{-1}(\{\mathbf{0}\})$, and the regular value assumption of $\mathbf{0}$ hence ensures DH to be of rank n at this limit point. The very same application of the Implicit Function Theorem then extends the curve a little further. As this process repeats itself, the smooth curve starting from $(\mathbf{x}_0, 0)$ can be extended indefinitely. This is the essence of homotopy continuation method. We say H satisfies the *smoothness condition* if $\mathbf{0}$ is a regular value of H and $H^{-1}(\{\mathbf{0}\}) \neq \varnothing$.

The solution curve defined by $H(\mathbf{x}, t) = \mathbf{0}$ starting from the point $(\mathbf{x}_0, 0)$ will lead to solutions to the system of interest $F(\mathbf{x}) = \mathbf{0}$ when this smooth curve crosses the plane of $\mathbb{R}^n \times \mathbb{R}$ defined by $t = 1$ (as illustrated in Figure 1).

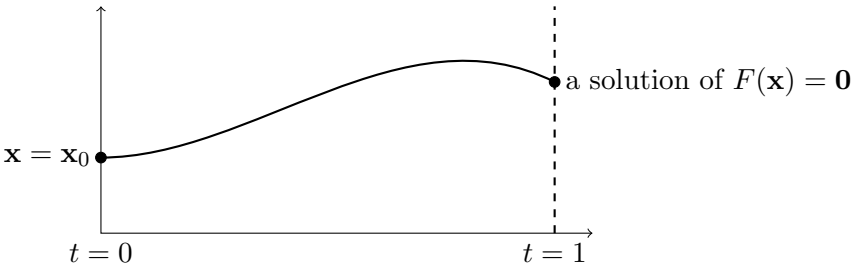


Figure 1. A smooth curve defined by $H(\mathbf{x}, t) = \mathbf{0}$ reaching a solution $\mathbf{x}^*$ of the target system.

2.2. The local description of a solution curve

In the following, we assume the homotopy H satisfies the smoothness condition, and our focus will concentrate on tracing those smooth curves in $H^{-1}(\{\mathbf{0}\})$ numerically. To trace a smooth curve $\gamma \subset \mathbb{R}^n \times \mathbb{R}$ that contains the starting point $(\mathbf{x}_0, 0)$, there is a wide range of variations on the basic methods. For brevity, we only discuss one specific method and refer to [4] for a comprehensive list of existing methods.

In the construction of the homotopy $H(\mathbf{x}, t) = \mathbf{0}$, it may seem natural to use t as the designated parameter for the solution curve as originally suggested by Davidenko [24]. However, this parametrization has a severe limitation, as t cannot be used as a smooth parameter in certain situations. For example, as shown in Figure 2, at points where $\partial H/\partial \mathbf{x}$ is singular the solution curve of $H(\mathbf{x}, t) = \mathbf{0}$ cannot be parametrized by t directly. Actually,

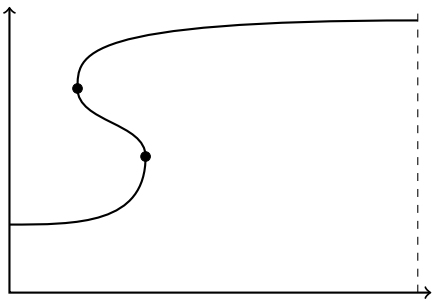


Figure 2. Points at which curves cannot be parametrized by t

one may avoid such difficulties by considering both $\mathbf{x}$ and t as independent variables and parametrize the smooth curve γ by the arc-length.

To facilitate our discussion, we shall use the notation $\mathbf{y} = (\mathbf{x}, t)$ and write $H(\mathbf{x}, t) = H(\mathbf{y})$, where none of the variables will be used as a parameter of solution curve γ of $H(\mathbf{y}) = \mathbf{0}$. An arc-length parametrization of γ is a smooth function $\mathbf{y} : \mathbb{R}^+ \to \gamma$ such that $\mathbf{y}(0) = (\mathbf{x}_0, 0)$, $H(\mathbf{y}(s)) = \mathbf{0}$, and the tangent vector $\dot{\mathbf{y}}$ always has unit length for all $s \in \mathbb{R}^+$. Those conditions for $\mathbf{y}(s)$ make it a solution of the following system of differential equations:

$$\begin{aligned} DH(\mathbf{y}(s)) \cdot \dot{\mathbf{y}}(s) &= \mathbf{0} \\ \|\dot{\mathbf{y}}(s)\| &= 1 \\ \mathbf{y}(0) &= (\mathbf{x}_0, 0). \end{aligned} \tag{2.1}$$

Apparently, solution of this system is not unique, since on every point of the smooth curve, there are always two different arc-length parametrizations moving towards opposite directions. Therefore, to trace a curve along arc-length parametrization without backtracking, one must determine and maintain a consistent orientation.

As in (2.1), $\dot{\mathbf{y}}(s)$ is in the null space of the $n \times (n+1)$ matrix $DH(\mathbf{y}(s))$, which is of rank n by assumption. Thus, the $(n+1) \times (n+1)$ square matrix $\begin{bmatrix} DH(\mathbf{y}(s)) \\ \dot{\mathbf{y}}(s) \end{bmatrix}$ must be nonsingular, and its determinant will never vanish for all $s \in \mathbb{R}^+$. So the sign of this determinant never changes along the curve, therefore

$$\sigma(\mathbf{y}) := \operatorname{sgn} \det \begin{bmatrix} DH(\mathbf{y}(s)) \\ \dot{\mathbf{y}}(s) \end{bmatrix}$$

can be used to determine the orientation of the parametrization. Once an orientation $\sigma_0 = \pm 1$ has been decided, this orientation $\sigma(\mathbf{y}) = \sigma_0$ must be

maintained in the process of tracing the curve by adjusting the sign of $\dot{\mathbf{y}}$. With this additional orientation constraint, equations of the arc-length parametrization for the smooth curve $\gamma \subset H^{-1}(\{\mathbf{0}\})$ become

$$\begin{aligned} DH(\mathbf{y}(s)) \cdot \dot{\mathbf{y}}(s) &= \mathbf{0} \\ \operatorname{sgn} \det \begin{bmatrix} DH(\mathbf{y}(s)) \\ \dot{\mathbf{y}}(s) \end{bmatrix} &= \sigma_0 \\ \|\dot{\mathbf{y}}(s)\| &= 1 \\ \mathbf{y}(0) &= (\mathbf{x}_0, 0) \end{aligned} \quad . \tag{2.2}$$

for chosen sign $\sigma_0 = +1$ or -1.

In principle, any of the available ODE solvers capable of integrating the above system can be used to trace the curve and obtain a solution to the target system at $t = 1$. However, due to numerical stability concerns, a more preferable method to trace the curve is the"prediction-correction scheme". In this scheme, one traces along the smooth curve γ via a series of discrete "prediction-correction" steps. Starting from a point known to be on or very close to the curve γ, the "prediction" step produces an approximation of a point "one step" further in the right orientation for a proper step size. Such a prediction step will be followed by a "correction" step in which a Newton-like method is applied to refine the prediction to within a close proximity of the curve γ. These steps may be repeated to move forward in the sense of increasing arc-length along the curve γ. Among many different choices for the "predictors" as well as "correctors", we shall present here a commonly used combination of the (generalized) Euler's method for the predictor and Newton's method for the corrector. They form a simple but effective prediction-correction duet. (See [4] for a comprehensive discussion of a wide range of variations.)

2.3. Euler's predictor

Starting from a point $\mathbf{y}_0 \in \mathbb{R}^{n+1}$ on the curve γ, the Euler's predictor simply moves one step along the tangent direction of γ at $\mathbf{y}_0$ for certain step size as depicted in Figure 3. The tangent direction $\dot{\mathbf{y}}$ is given by (2.2), which can be computed numerically via the QR-decomposition of $DH(\mathbf{y}_0)^\top$: If

$$(DH(\mathbf{y}_0))^\top = Q \begin{bmatrix} R \\ \mathbf{0} \end{bmatrix}$$

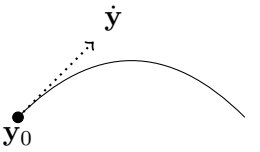


Figure 3. Euler's predictor moves one step along the tangent direction

where Q is orthogonal and R is upper triangular with full rank, let $\Delta\mathbf{y} = Q\,\mathbf{e}_{n+1}$ where $\mathbf{e}_{n+1} = (0,\ldots,0,1)^T \in \mathbb{R}^{n+1}$. Clearly, $\|\Delta\mathbf{y}\|_2 = 1$ and $DH(\mathbf{y}_0)\cdot\Delta\mathbf{y} = \mathbf{0}$. So, up to a sign change, $\Delta\mathbf{y}$ satisfies (2.2). To choose the correct sign, note that

$$\begin{bmatrix} DH(\mathbf{y}_0) \\ \Delta\mathbf{y} \end{bmatrix} = \begin{bmatrix} R^T \\ \mathbf{e}_{n+1}^T \end{bmatrix} Q^T,$$

thus $\det\begin{bmatrix} DH(\mathbf{y}_0) \\ \Delta\mathbf{y} \end{bmatrix} = \det R \cdot \det Q$. Signs of both $\det R$ and $\det Q$ are easy to compute: with R being upper triangular, the sign of its determinant is simply the product of the signs of its diagonal entries. If the QR-decomposition is computed via Householder transformations, as we usually do, then $Q = Q_1 \cdots Q_k$ where each factor represents a Householder transformation with determinant being -1. Hence the sign of $\det Q = (-1)^k$, and a proper assignment of the sign σ, the sign of $\det\begin{bmatrix} DH(\mathbf{y}_0) \\ \sigma\cdot\Delta\mathbf{y} \end{bmatrix}$ will agree with the original orientation σ_0 of γ. With step size Δs, the point

$$\hat{\mathbf{y}} = \mathbf{y}_0 + \Delta s \cdot \sigma \cdot \Delta\mathbf{y}$$

is designated as the resulting *prediction*.

2.4. Generalized Newton's corrector

The resulting *prediction* produced by a predictor may not be exactly on or even very close to the curve γ. If the next prediction step starts from such a poor approximatiom, the error can quickly build up to an unacceptable level. To curb such error accumulation, a "correction" step is necessary to return the resulting prediction $\hat{\mathbf{y}}$ to (or close to) the curve γ. Let $\mathbf{y}$ denote the point on the curve γ that is nearest to the prediction $\hat{\mathbf{y}}$. The point $\mathbf{y}$ solves the following optimization problem:

$$\begin{aligned} &\min \|\mathbf{y} - \hat{\mathbf{y}}\| \\ &H(\mathbf{y}) = \mathbf{0} \end{aligned} \tag{2.3}$$

as illustrated in Figure 4. Indeed, it can be shown that a unique solution exists as long as $\hat{\mathbf{y}}$ is sufficiently close to γ.

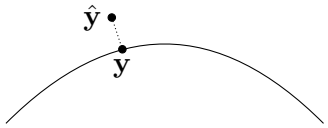


Figure 4. The generalized Newton's corrector finds a point on the curve that is closest to the prediction

From the theory of Lagrange multipliers, under the smoothness condition, the solution to the above minimization problem must satisfy the *Lagrangian equations*

$$\begin{aligned} H(\mathbf{y}) &= \mathbf{0} \\ \mathbf{y} - \hat{\mathbf{y}} \in \operatorname{range}(DH)^\top &= (\operatorname{Ker} DH)^\perp. \end{aligned}$$

Alternatively, we write

$$\begin{aligned} H(\mathbf{y}) &= \mathbf{0} \\ (\mathbf{z}(\mathbf{y}))^\top (\mathbf{y} - \hat{\mathbf{y}}) &= 0 \end{aligned} \tag{2.4}$$

where $\mathbf{z}(\mathbf{y}) \in \ker DH(\mathbf{y})$ with $\|\mathbf{z}\| = 1$. Linearizing (2.4) at $\hat{\mathbf{y}}$ by taking the first two terms of the Taylor series expansions leads to

$$\begin{aligned} H(\hat{\mathbf{y}}) + DH(\hat{\mathbf{y}})(\mathbf{y} - \hat{\mathbf{y}}) &= \mathbf{0} \\ \mathbf{z}(\hat{\mathbf{y}})^\top (\mathbf{y} - \hat{\mathbf{y}}) &= 0. \end{aligned} \tag{2.5}$$

Since $\begin{bmatrix} DH(\hat{\mathbf{y}}) \\ \mathbf{z}(\hat{\mathbf{y}})^\top \end{bmatrix}$ is nonsingular, so

$$\begin{bmatrix} DH(\hat{\mathbf{y}}) \\ \mathbf{z}(\hat{\mathbf{y}})^\top \end{bmatrix} \begin{bmatrix} DH^+(\hat{\mathbf{y}}) & \mathbf{z}(\hat{\mathbf{y}}) \end{bmatrix} = I_{n+1}$$

where $DH^+(\hat{\mathbf{y}})$ is the Moore-Penrose inverse of the $n \times (n+1)$ matrix $DH(\hat{\mathbf{y}})$ of H at $\hat{\mathbf{y}}$; i.e., $DH^+(\hat{\mathbf{y}}) = DH(\hat{\mathbf{y}})^T(DH(\hat{\mathbf{y}})DH(\hat{\mathbf{y}})^T)^{-1}$. From (2.5),

$$\begin{bmatrix} DH(\hat{\mathbf{y}}) \\ \mathbf{z}(\hat{\mathbf{y}})^\top \end{bmatrix} (\mathbf{y} - \hat{\mathbf{y}}) = - \begin{bmatrix} H(\hat{\mathbf{y}}) \\ 0 \end{bmatrix}.$$

Consequently,

$$\mathbf{y} - \hat{\mathbf{y}} \;=\; -(DH(\hat{\mathbf{y}}))^+ H(\hat{\mathbf{y}}).$$

This leads to the generalized Newton's iteration: the **Newton point** $\mathcal{N}(\hat{\mathbf{y}})$ for approximating the solution of (2.3) is given by

$$\mathcal{N}(\hat{\mathbf{y}}) := \hat{\mathbf{y}} - (DH(\hat{\mathbf{y}}))^+ H(\hat{\mathbf{y}}).$$

The map $\mathcal{N} : \mathbb{R}^{n+1} \to \mathbb{R}^{n+1}$ will also be called the **Newton map**.

Starting from the potentially inaccurate prediction $\hat{\mathbf{y}}$ produced by the Euler's predictor, the map $\mathcal{N}$ is iteratively applied to $\hat{\mathbf{y}}$ until certain criterion of convergence is met. Stated formally, the result of the correction is

$$\mathbf{y} = \mathcal{N}^k(\hat{\mathbf{y}}) = \mathcal{N} \circ \mathcal{N} \circ \cdots \circ \mathcal{N}(\hat{\mathbf{y}})$$

where $k \in \mathbb{Z}^+$ is determined by convergence criterion. It is well known that this generalized Newton method converges quadratically for $\hat{\mathbf{y}}$ sufficiently close to γ under the smoothness assumption (see, e.g., [88] and [89]). Therefore, the shrinking distance $\|\mathcal{N}^j(\hat{\mathbf{y}}) - \mathcal{N}^{j-1}(\hat{\mathbf{y}})\|$ between successive points produced by the iterations can be used as a criterion for convergence.

Of course, if the iterations fail to converge, one must go back to adjust the step size for the Euler's predictor.

3. Varieties of homotopies

In the above, we introduced the notion of commencing at a given known point $(\mathbf{x}_0, 0)$ and tracing the solution curve of a homotopy equation $H(\mathbf{x}, t) = \mathbf{0}$ to a solution of the nonlinear system of equations asked to be solved. The exact solution curve that occurs will depend directly upon the selected homotopy function $H(\mathbf{x}, t)$. A great number of different types of homotopy functions have been developed over the last several decades. Here we do not intend to provide an exhaustive list. Instead, this section will highlight three of the commonly used constructions.

3.1. Fixed point homotopy

Let $F : \mathbb{R}^n \to \mathbb{R}^n$ and $\mathbf{x} = (x_1, \ldots, x_n)$. One of the simplest homotopy for finding solutions of $F(\mathbf{x}) = \mathbf{0}$ is the **fixed point homotopy** given by

$$H(\mathbf{x}, t) = (1 - t)(\mathbf{x} - \mathbf{a}) + tF(\mathbf{x})$$

where $\mathbf{a} \in \mathbb{R}^n$ and $t \in \mathbb{R}$. At $t = 0$, the starting system is $H(\mathbf{x}, 0) = \mathbf{x} - \mathbf{a} = \mathbf{0}$ for which the only solution is $\mathbf{x} = \mathbf{a}$. At $t = 1$, the system $H(\mathbf{x}, 1) = F(\mathbf{x}) = \mathbf{0}$ is the system of equations of interest. The smoothness of this homotopy construction is warranted by *Generalized Sard's theorem*[1] for which we shall state the simplified form:

Theorem 3.1 (Generalized Sard's Theorem). *For two open sets $X \subset \mathbb{R}^{n_1}$ and $Y \subset \mathbb{R}^{n_2}$, let $\phi : X \times Y \to \mathbb{R}^m$ be a smooth function. If $\mathbf{v} \in \mathbb{R}^m$ is a regular value of ϕ, then for almost all $\mathbf{y} \in Y$, $\mathbf{v}$ is a regular value of $\phi_{\mathbf{y}} = \phi(\bullet, \mathbf{y}) : X \to \mathbb{R}^m$.*

To apply this to the fixed point homotopy defined above, consider $\mathbf{a}$ as a variable and define $\phi : \mathbb{R}^n \times (\mathbb{R} \setminus \{1\}) \times \mathbb{R}^n \to \mathbb{R}^n$ by $\phi(\mathbf{x}, t, \mathbf{a}) = (1 - t)(\mathbf{x} - \mathbf{a}) + tF(\mathbf{x})$. Then

$$D\phi(\mathbf{x}, t, \mathbf{a}) = \left[(1 - t)I + tDF(\mathbf{x}),\ -(\mathbf{x} - \mathbf{a}) + F(\mathbf{x}),\ -(1 - t)I\right].$$

With the last n columns being $-(1 - t)I$, this matrix has rank n for any $(\mathbf{x}, t, \mathbf{a}) \in \mathbb{R}^n \times (\mathbb{R} \setminus \{1\}) \times \mathbb{R}^n$ satisfying $\phi(\mathbf{x}, t, \mathbf{a}) = \mathbf{0}$. Namely, $\mathbf{0} \in \mathbb{R}^n$ is a regular value of the map $\phi : \mathbb{R}^n \times (\mathbb{R} \setminus \{1\}) \times \mathbb{R}^n \to \mathbb{R}^n$. By Generalized Sard's theorem, for almost all $\mathbf{a} \in \mathbb{R}^n$, $\mathbf{0}$ is a regular value of $\phi(\bullet, \bullet, \mathbf{a}) = H(\bullet, \bullet)$. Thus, when $\mathbf{a} \in \mathbb{R}^n$ is chosen at random then, with probability one, H satisfies the *smoothness condition*, for $t \neq 1$, since $H(\mathbf{a}, 0) = \mathbf{0}$ (so, the inverse image $H^{-1}(\{\mathbf{0}\})$ is not empty). The system $H(\mathbf{x}, t) = \mathbf{0}$ defines (disjoint) smooth *solution curves* in $\mathbb{R}^n \times (\mathbb{R} \setminus \{1\})$. The general method for tracing smooth curves described in §2 can then be used to trace the unique smooth solution curve emanating from $(\mathbf{x}, t) = (\mathbf{a}, 0)$. When this curve passes through the hyperplane defined by $t = 1$, a solution of the target system $F(\mathbf{x}) = \mathbf{0}$ would be attained. It is, of course, possible for the curve passing through $t = 1$ multiple times and we shall obtain multiple solutions of $F(\mathbf{x}) = \mathbf{0}$.

This homotopy construction was first introduced in [24]. Partially due to its easily established smoothness condition, it has been used and studied

intensively since 1950s. We highlight here two classes of theorems related to this homotopy.

3.1.1. Fixed point theorems. In the above, the target functions $F(\mathbf{x})$ were defined on the entire space $\mathbb{R}^n$. In most occasions, however, they were restricted to inhabit a certain domain, such as $F : U \to \mathbb{R}^n$ where $U \subset \mathbb{R}^n$ is open. Similarly, sometimes it is useful to restrict the variable t to the unit interval $I = [0, 1]$. With the restricted domains U and I, the framework of our homotopy will be

$$H : U \times [0, 1] \to \mathbb{R}^n.$$

Same as before, if $\mathbf{0}$ is a regular value of H with nonempty inverse image, that is, H satisfies the smoothness condition, then $H^{-1}(\{\mathbf{0}\})$ consists of smooth curves in $U \times [0, 1]$. The underlying principle for finding a solution of nonlinear system of equations $F(\mathbf{x}) = \mathbf{0}$ of interest with selected homotopy H is to trap a solution curve $\gamma \subset H^{-1}(\{\mathbf{0}\})$ inside U with starting point $(\mathbf{a}, 0)$ where $\mathbf{a} \in U$ and force the curve to proceed to the appropriate place. For instance, if we ensure the solution curve γ never pierce the boundary of U, denoted by ∂U, then γ cannot escape from U. It will then be forced to go to a proper location — a solution of the target system at $t = 1$.

Given a homotopy $H : U \times [0, 1] \to \mathbb{R}^n$ and a $t_0 \in [0, 1]$, let

$$H|^{-1}_{t=t_0}(\{\mathbf{0}\}) = \{\, \mathbf{x} \in U \mid H(\mathbf{x}, t_0) = \mathbf{0} \,\}.$$

We say H is *boundary-free* [32] at $t_0 \in [0, 1]$ if $\mathbf{x} \notin \partial U$ for any $\mathbf{x} \in H|^{-1}_{t=t_0}(\{\mathbf{0}\})$. In general we say H is boundary-free for t in a subset $S \subset [0, 1]$ if H is boundary-free for all $t \in S$. Accordingly, when H is boundary-free for $S = [0, 1)$, then the only place the solution curve γ could possibly touch the boundary of U is at $t = 1$.

For $U \subset \mathbb{R}^n$, we use $\operatorname{int} U$ to denote the interior of U.

Theorem 3.2. *Given smooth function $F : U \to \mathbb{R}^n$, let $U \subset \mathbb{R}^n$ be compact and $\operatorname{int} U \neq \varnothing$. For some $\mathbf{a} \in \operatorname{int} U$, if $H : U \times [0, 1] \to \mathbb{R}^n$ is boundary-free for $0 \leq t < 1$, where*

$$H(\mathbf{x}, t) = (1 - t)(\mathbf{x} - \mathbf{a}) + t(\mathbf{x} - F(\mathbf{x})), \tag{3.1}$$

then F has a fixed point, i.e., there exists an $\mathbf{x}^* \in U$ such that $F(\mathbf{x}^*) = \mathbf{x}^*$.

Proof. From what had been developed before, the proof of this theorem is quite straightforward. Since by Generalized Sard's Theorem, for almost

all $\mathbf{a} \in U$, H in (3.1) satisfies the smoothness condition for $t \neq 1$. Therefore, the smooth solution curve $\gamma \subset H^{-1}(\{\mathbf{0}\})$ starting at $(\mathbf{a}, 0)$ exists. Prohibited by the assumption that H is boundary free, the curve cannot meet ∂U for $0 \leq t < 1$. Since U is compact and thus bounded, the curve must reach $t = 1$. A point $(\mathbf{x}^*, 1) \in H^{-1}(\{\mathbf{0}\})$ therefore exists for which $F(\mathbf{x}^*) = \mathbf{x}^*$. □

There are two very important conditions both ensure H is boundary-free for $t \in [0, 1)$: Given $\mathbf{a} \in \operatorname{int} U$

Leray-Shauder: $\mathbf{x} \neq tF(\mathbf{x}) + (1-t)\mathbf{a}$ for $0 < t < 1$ and $\mathbf{x} \in \partial U$.

Geometrically, this condition means that given any $\mathbf{x} \in \partial U$, the point $\mathbf{x}$ does not lie in the interior of the line segment joining $\mathbf{a}$ and $F(\mathbf{x})$. Since $\mathbf{a} \in \operatorname{int} U$ and $\mathbf{a} \in H|_{t=0}^{-1}(\{\mathbf{0}\})$ is unique, H is boundary-free at $t = 0$. Simplifying (3.1) yields that any point $(\mathbf{x}, t)$ on the solution curve must satisfy $\mathbf{x} = t\,F(\mathbf{x}) + (1-t)\,\mathbf{a}$ for $0 < t < 1$. Therefore, H is boundary-free for $0 \leq t < 1$.

Brouwer: $F : U \to U$ is smooth and U is convex.

Since $F : U \to U$, for any $\mathbf{x} \in U$, $F(\mathbf{x}) \in U$. Hence the point $\hat{\mathbf{x}} = t\,F(\mathbf{x}) + (1-t)\,\mathbf{a}$ for $0 < t < 1$ must be in $\operatorname{int} U$ since U is convex. Thus if $\mathbf{x} \in \partial U$, then $\mathbf{x} \neq t\,F(\mathbf{x}) + (1-t)\,\mathbf{a}$ for $0 < t < 1$, and therefore H is boundary-free for $0 \leq t < 1$.

As a corollary of Theorem 3.2, under either of the above conditions, a fixed point of F exists when U is compact and $\operatorname{int} U$ is nonempty.

Remark 3.3. The celebrated Brouwer fixed point theorem given above was developed very early in the last century. It was used widely in economics, biology, ecology, medicine, physics, chemistry, and other fields to classify the *equilibrium*. Most importantly, in the proof of this theorem via the *fixed point homotopy* as described above actually provides a means to calculate those *equilibrium* constructively.

3.1.2. Existence of solutions to systems of nonlinear equations. Given a smooth function $F : \mathbb{R}^n \to \mathbb{R}^n$, a condition that ensures the existence of a solution to the system of nonlinear equations $F(\mathbf{x}) = \mathbf{0}$ in a domain $U \subset \mathbb{R}^n$ is for U to be compact and having an interior point $\mathbf{a}$ such that for all $\mathbf{x} \in \partial U$, $F(\mathbf{x})^\top(\mathbf{x} - \mathbf{a}) > 0$ (or, equivalently, $F(\mathbf{x})^\top(\mathbf{x} - \mathbf{a}) < 0$).

Theorem 3.4. *Let $F : \mathbb{R}^n \to \mathbb{R}^n$ be smooth, and $U \subset \mathbb{R}^n$ be a compact subset with nonempty interior. Let $\mathbf{a} \in \operatorname{int} U$ be arbitrary. If for all $\mathbf{x} \in \partial U$, $F(\mathbf{x})^\top(\mathbf{x} - \mathbf{a}) > 0$, then there exists an $\mathbf{x}^* \in \operatorname{int} U$ such that $F(\mathbf{x}^*) = \mathbf{0}$.*

Proof. Let the homotopy $H : \mathbb{R}^n \times [0,1] \to \mathbb{R}^n$ be given as

$$H(\mathbf{x}, t) = (1-t)(\mathbf{x} - \mathbf{a}) + t\,F(\mathbf{x}).$$

As before, for almost all $\mathbf{a}$, H satisfies the smoothness condition for $t \neq 1$. This warrants the solution set of $H(\mathbf{x}, t) = \mathbf{0}$ consists of all smooth curves. Simply by inspection, $\mathbf{a}$ is the unique solution of $H(\mathbf{x}, 0) = \mathbf{0}$. Also, if $\mathbf{x} \in \partial U$, then

$$H(\mathbf{x}, t)^\top (\mathbf{x} - \mathbf{a}) = (1-t)\|\mathbf{x} - \mathbf{a}\|^2 + tF(\mathbf{x})^\top (\mathbf{x} - \mathbf{a}) > 0.$$

So, the solution curve defined by $H(\mathbf{x}, t) = \mathbf{0}$ that emanates from $(\mathbf{a}, 0)$ cannot return to $t = 0$, and it cannot reach the boundary of U. Thus it must extend to $t = 1$ within the interior of U, yielding a point $(\mathbf{x}^*, 1) \in H^{-1}(\{\mathbf{0}\})$ with $\mathbf{x}^* \in U$ which satisfies $F(\mathbf{x}^*) = \mathbf{0}$. □

Note that the set U in the above theorem can be selected quite arbitrarily, so the condition $F(\mathbf{x})^\top (\mathbf{x} - \mathbf{a}) > 0$ in the theorem is very flexible.

3.2. Newton homotopy

Another commonly used homotopy function is the **Newton homotopy** $H : \mathbb{R}^{n+1} \to \mathbb{R}^n$ given by

$$\begin{aligned} H(\mathbf{x}, t) &= (1-t)[F(\mathbf{x}) - F(\mathbf{a})] + tF(\mathbf{x}) \\ &= F(\mathbf{x}) - (1-t)F(\mathbf{a}) \end{aligned} \tag{3.2}$$

where $F : \mathbb{R}^n \to \mathbb{R}^n$ is the smooth system of interest, and $\mathbf{a}$ is a generically chosen point in $\mathbb{R}^n$. Clearly, at $t = 1$, $H(\mathbf{x}, 1) \equiv F(\mathbf{x}) = \mathbf{0}$, and at $t = 0$, the starting system is $H(\mathbf{x}, 0) = F(\mathbf{x}) - F(\mathbf{a}) = \mathbf{0}$ for which $\mathbf{a}$ is a solution. As before, if H satisfies the smoothness condition, then the solution set of $H(\mathbf{x}, t) = \mathbf{0}$ in $\mathbb{R}^{n+1}$ consists of smooth curves, and the predictor-corrector scheme developed in §2 can be used to trace the smooth solution curve emanating from $(\mathbf{a}, 0) \in \mathbb{R}^{n+1}$.

Let $U \subset \mathbb{R}^n$ be open and bounded with a smooth and connected boundary. The smoothness of H as well as its boundary-free property with respect to U can be established via Smale's boundary conditions:

1) $F : \mathbb{R}^n \to \mathbb{R}^n$ is a smooth map;

2) $\mathbf{0}$ is a regular value of F;

3) F has no zero on ∂U;

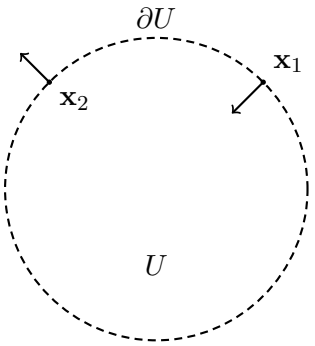


Figure 5. Smale's boundary condition requires that the *Newton direction* at any point on the boundary of U to point either in or out of U but not tangent to ∂U.

4) $DF(\mathbf{x})$ is nonsingular for all $\mathbf{x} \in \partial U$; and

5) at any $\mathbf{x} \in \partial U$, the *Newton direction*

$$-(DF(\mathbf{x}))^{-1}F(\mathbf{x})$$

is not tangent to ∂U .

Under these assumptions, Percell proved [90] that for almost all $\mathbf{a} \in \partial U$, the Newton homotopy (3.2) satisfies the smoothness condition, that is, $\mathbf{0}$ is a regular value of H. Furthermore, Smale [98] showed that in this circumstance, the smooth solution curve of $H(\mathbf{x}, t) = \mathbf{0}$ passing through the starting point $(\mathbf{a}, 0)$ and pointing into U must reach at least one point $(\mathbf{x}^*, 1)$ at the level $t = 1$ and a solution of $F(\mathbf{x}) = \mathbf{0}$ is found. Although this Smale's boundary conditions is generally difficult to verify, for those functions F having only isolated nonsingular solutions it can be shown (see [89]) that these conditions are satisfied by some sufficiently small ball around each solution of F. This can be considered as a generalization of the well known Newton-Kantorovitch Theorem [47].

It is worth noting the close connection between the Newton homotopy and the well known Newton's method for solving nonlinear equations: Differentiating both sides of $H(\mathbf{x}, t) = \mathbf{0}$ given by (3.2) yields the initial value problem

$$\begin{aligned} \dot{\mathbf{x}}(t) &= -(DF(\mathbf{x}(t)))^{-1}F(\mathbf{a}) \\ \mathbf{x}(0) &= \mathbf{a} \end{aligned} \tag{3.3}$$

on domains in which $DF(\mathbf{x})$ is nonsingular. Applying Euler's method at $t = 0$ with step size 1 to the above ODE from the initial point $\mathbf{x} = \mathbf{a}$, the approximation of $\mathbf{x}(1)$ becomes

$$\mathbf{x}^{(1)} = \mathbf{a} - (DF(\mathbf{a}))^{-1} F(\mathbf{a})$$

which is precisely a single iteration of Newton's method. Hence, Newton's iteration can be considered as the application of Euler's method with step size 1 on the solution curve given by the Newton homotopy (3.2).

However, in contrast to Newton's method which is generally a local method, the Newton homotopy exhibits certain global convergence property via Smale's boundary conditions for instance. The detailed comparison between the two can be found in [4].

3.3. *d*-homotopy method

The *d*-homotopy (also known as *d*-trick homotopy) method is designed to find additional solutions after an isolated regular solution of the system $F(\mathbf{x}) = \mathbf{0}$ has been found. Assuming $\mathbf{x}^*$ is a known isolated regular solution of $F(\mathbf{x}) = \mathbf{0}$, the *d*-homotopy is given by

$$H(\mathbf{x}, t) = F(\mathbf{x}) + t\,\mathbf{d} \tag{3.4}$$

for some chosen $\mathbf{d} \in \mathbb{R}^n$. The Newton homotopy can be considered a special case of this *d*-homotopy with $\mathbf{d} = -F(\mathbf{a})$. For simplicity, we further assume all the solutions of the system $F(\mathbf{x}) = \mathbf{0}$ are isolated and regular, that is, $DF(\mathbf{x})$ is nonsingular for all $\mathbf{x} \in F^{-1}(\{\mathbf{0}\})$. With this assumption, the regularity of the *d*-homotopy can be established via Generalized Sard's theorem by the same technique used in §3.1 for the fixed point homotopy: Take $\mathbf{d}$ as a variable and let $\phi : (\mathbb{R}^n \times \mathbb{R}) \times \mathbb{R}^n \to \mathbb{R}^n$ be given by

$$\phi(\mathbf{x}, t, \mathbf{d}) = F(\mathbf{x}) + t\mathbf{d}.$$

Then

$$D\phi = \begin{bmatrix} DF & \mathbf{d} & tI_n \end{bmatrix}$$

where I_n is the $n \times n$ identity matrix.

Since the last n columns is tI_n, the rank of the above matrix is n for $t \neq 0$. At $t = 0$, $\phi(\mathbf{x}, 0, \mathbf{d}) \equiv F(\mathbf{x})$ which, by assumption, has only isolated regular solutions. Hence DF is nonsingular for all $\mathbf{x} \in \mathbb{R}^n$ satisfying $\phi(\mathbf{x}, 0, \mathbf{d}) = \mathbf{0}$. Therefore the matrix $D\phi$ has rank n for all $(\mathbf{x}, t, \mathbf{d}) \in (\mathbb{R}^n \times \mathbb{R}) \times \mathbb{R}^n$ for

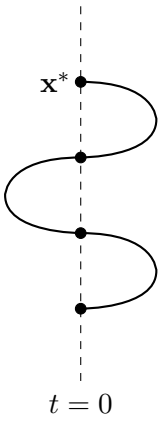


Figure 6. The d-homotopy defines a curve that turns around and intersect the hyperplane at $t = 0$ again.

which $\phi(\mathbf{x}, t, \mathbf{d}) = \mathbf{0}$. Rephrasing: $\mathbf{0}$ is a regular value of ϕ. By Generalized Sard's theorem, for almost all $\mathbf{d} \in \mathbb{R}^n$, $\mathbf{0}$ is a regular value of $H(\bullet, \bullet) = \phi(\bullet, \bullet, \mathbf{d})$. Consequently, when $\mathbf{d} \in \mathbb{R}^n$ is chosen at random, with *probability one*, H satisfies the smoothness condition, and solution set of $H(\mathbf{x}, t) = \mathbf{0}$ consist of (disjoint) smooth curves in $\mathbb{R}^n \times \mathbb{R}$. Starting from $(\mathbf{x}^*, 0)$ with the known solution $\mathbf{x} = \mathbf{x}^*$ of $F(\mathbf{x}) = \mathbf{0}$, we trace the unique smooth curve defined by $H(\mathbf{x}, t) = \mathbf{0}$ that passes through $(\mathbf{x}^*, 0)$. Slightly different from the framework outlined in §2 and §3.1, we wish this solution curve will turn back and intersect the hyperplane at $t = 0$ again, producing an additional solution to $F(\mathbf{x}) = \mathbf{0}$ at the intersection point. Figure 6 is a depiction of the situation. Similar to all homotopies in the previous sections, it is possible for the curve to pass through $t = 0$ multiple times afterwards, obtaining even more solutions. The following proposition says actually the *boundary-free* condition ensures the *turning back* of the solution curve and its intersection with the hyperplane at $t = 0$ again.

Proposition 3.5. *For a smooth function $F : \mathbb{R}^n \to \mathbb{R}^n$, let $U \subset \mathbb{R}^n$ be a bounded open set which contains a known solution $\mathbf{x}^*$ of $F(\mathbf{x}) = \mathbf{0}$. If*

(1) *$\mathbf{0}$ is a regular value of F,*

(2) *For chosen $\mathbf{d}$, the* boundary-free *condition, that is, $H(\mathbf{x}, t) = F(\mathbf{x}) - t\mathbf{d} \neq \mathbf{0}$ holds for all $\mathbf{x} \in \partial U$ and $t \in \mathbb{R}$.*

Then the curve $\gamma \subset H^{-1}(\{\mathbf{0}\})$ *which contains* $(\mathbf{x}^*, 0)$ *will intersect* $U \times \{0\}$ *again.*

Proof. The boundary-free condition (2) makes the curve γ lying strictly inside the cylinder $\operatorname{int} U \times \mathbb{R}$. A solution $(\mathbf{x}, t) \in \gamma$ must satisfy $F(\mathbf{x}) = t\mathbf{d}$ and $\mathbf{x} \in U$. Hence,

$$|t| = \frac{\|F(\mathbf{x})\|}{\|\mathbf{d}\|} \tag{3.5}$$

remains bounded. The assertion of the proposition thus follows because the solution curve γ has "no place to run". □

In the proof above, the solution curve γ may be traced from $(\mathbf{x}^*, 0)$ in either direction: increasing in t or decreasing in t. When γ runs into the region with $t > 0$, (3.5) becomes

$$t = \frac{\|F(\mathbf{x})\|}{\|\mathbf{d}\|}.$$

Therefore under the relaxed boundary-free condition

(2a) $H(\mathbf{x}, t) = F(\mathbf{x}) - t\mathbf{d} \neq \mathbf{0}$ for all $\mathbf{x} \in \partial U$ and $t \geq 0$, the assertion of the proposition is still valid.

Similarly, the boundary-free condition (2) in the proposition can also be relaxed to

(2b) $H(\mathbf{x}, t) = F(\mathbf{x}) - t\mathbf{d} \neq \mathbf{0}$ for all $\mathbf{x} \in \partial U$, $t \leq 0$.

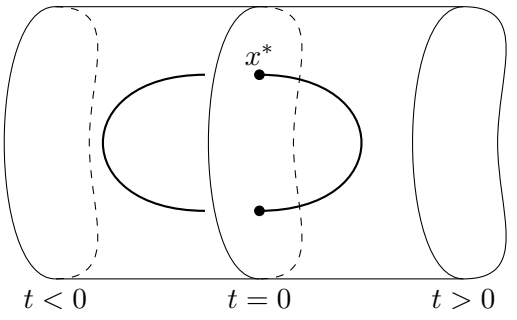


Figure 7. The d-homotopy defines a solution curve that is "trapped" inside a finite cylinder.

3.4. Strengths of homotopy methods and case studies

In contrast to local methods, such as Newton's method, for solving nonlinear systems, one great advantage of homotopy methods is the global nature manifested in Theorems 3.2, 3.4, the Leray-Shauder, Brouwer fixed point conditions, and the Smale's boundary conditions listed above. This section highlights other noteworthy strengths of (real) homotopy methods through concrete examples.

3.4.1. Handling singular solutions. Unlike Newton's method that generally fail near singular solutions, the homotopy continuation method has a particularly valuable advantage in handling singular solutions. We shall illustrate this via two simple examples.

Example 3.6 (A trivial example). We start with a trivial yet illuminating example: $f(x) = x^2$. It is immediate that the only solution of $f(x) = 0$ is $x = 0$, and it is singular since $f(x)$ and $f'(x)$ both vanish at $x = 0$. Thus a direct application of Newton's iteration on a point near $x = 0$ would face numerical difficulties as $f'(x)$ is close to zero.

However, with homotopy continuation method, it is possible to obtain the singular solution $x = 0$ with no difficulties. Consider for example the Newton homotopy construction given by

$$h(x, t) = f(x) - (1 - t)f(a) = x^2 - (1 - t)a^2$$

for a randomly chosen nonzero $a \in \mathbb{R}$. The equation $h(x, t) = 0$ actually defines a smooth curve that passes though the singular solution at $(x, t) = (0, 1)$ shown in Figure 8. As the curve tracing algorithm march towards the singular solution, nothing extraordinary happens. After all, the singular solution is simply a smooth point on the smooth solution curve, exhibiting no additional numerical difficulties than any other points on the curve.

While numerical analysts are mostly familiar with the modifications to Newton's method (see standard text such as [104]) which can be used to locate the singular solution $x = 0$ of this equation ($f(x) = x^2 = 0$), there are, however, situations where no direct modification of Newton's method can salvage, but a homotopy-based method would experience no numerical difficulties. The next example is one such system — the Griewank-Osborne system.

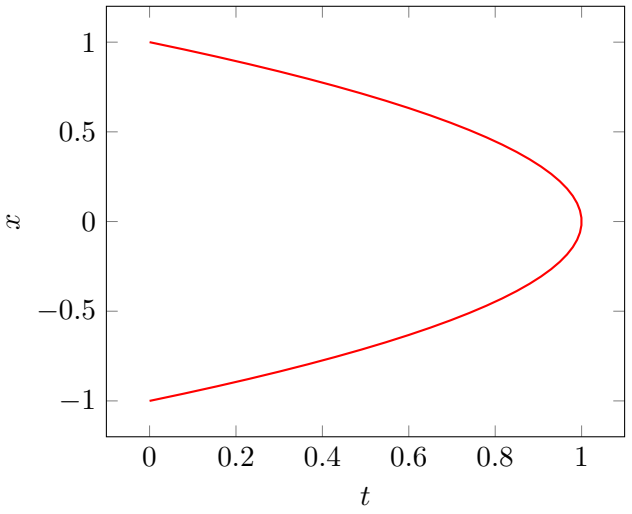


Figure 8. The smooth curve defined by Newton homotopy applied to the equation $f(x) = x^2 = 0$.

Example 3.7 (The Griewank-Osborne system). The Griewank-Osborne system, analyzed in [35] is given by:

$$F(x,y) = \begin{cases} \frac{29}{16}x^3 - 2xy = 0 \\ y - x^2 = 0. \end{cases} \tag{3.6}$$

The only solution of this system in $\mathbb{R}^2$ is $(0,0)$ which is singular since Jacobian matrix DF of F is singular at $(0,0)$. It is notoriously known that even starting from points arbitrarily close to the solution $(0,0)$ of the system Newton's method can exhibit chaotic behaviors: It may converge infinitely slowly or even diverge completely.

In contrast, we shall show that the Newton homotopy $H : \mathbb{R}^3 \to \mathbb{R}^2$ constructed for the Griewank-Osborne system given by

$$H(x,y,t) = \begin{cases} \frac{29}{16}x^3 - 2xy - t\left(\frac{29}{16}a^3 - 2ab\right) \\ y - x^2 - t\,(b - a^2) \end{cases} \tag{3.7}$$

satisfies the smoothness condition with plenty choices of $(a,b) \in \mathbb{R}^2$ and is capable of finding the singular solution $(x,y) = (0,0)$ with no difficulties. Parts of these analysis first appeared in [72, 73].

Proposition 3.8. *For almost all $(a, b) \in \mathbb{R}^2$ in the sense of Lebesgue measure, the Newton homotopy H, defined in (3.7), satisfies the smoothness assumption. Consequently, for almost all $(a, b) \in \mathbb{R}^2$, $H^{-1}((0, 0))$ consists of disjoint smooth curves.*

Proof. Points $(a, b) \in \mathbb{R}^2$ that violate the smoothness assumption are those points for which there exists $(x, y, t) \in \mathbb{R}^3$ such that $H(x, y, t) = \mathbf{0}$ but $\operatorname{rank} DH(x, y, t) < 2$. In other words, all 2×2 minors of the 2×3 matrix $DH(x, y, t)$ vanish, thus providing a system of 5 equations in 5 unknowns (x, y, t, a, b):

$$\begin{aligned}
\frac{29}{16}x^3 - 2xy - t\,(\frac{29}{16}a^3 - 2ab) &= 0 \\
y - x^2 - t\,(b - a^2) &= 0 \\
\frac{23}{16}x^2 - 2y &= 0 \\
\frac{1}{16}(a^2 - b)(87x^2 - 32y) - \frac{1}{8}(29a^3 - 32ab)x &= 0 \\
\frac{29}{16}a^3 - 2ab - 2(a^2 - b)x &= 0.
\end{aligned}$$

The associated primes of the ideal induced by this system (in $\mathbb{R}[x, y, t, a, b]$) are $\langle x, y, t, 29a^2 - 32b\rangle, \langle x, y, a, b\rangle, \langle x, y, t, a\rangle$. Therefore the only choices of $(a, b) \in \mathbb{R}^2$ that violate the smoothness assumption are those with $29a^2 = 32b$ or $a = 0$ which forms a (nowhere dense set) of measure zero. As a result, for almost all $(a, b) \in \mathbb{R}^2$, (3.7) is regular. □

Proposition 3.9. *For all $(a, b) \in \mathbb{R}^2$ with $b > \frac{119}{128}a^2$, there is an open and unbounded set $U \subset \mathbb{R}^2$ of positive Lebesgue measure such that for all $(a, b) \in U$, the equation $H(x, y, t) = (0, 0)$ defines a smooth curve containing both $(a, b, 1)$ and $(0, 0, 0)$.*

By this proposition, the Newton homotopy method via curve tracing starting from any point $(a, b) \in U$, will reach the (only) solution $(0, 0)$ of the original system (3.6). This is supported by our numerical experiments. Figure 9 shows the scatter plot of the starting points $(a, b) \in \mathbb{R}^2$ for which the Newton homotopy method using floating point arithmetic were successful in obtaining the solution $(0, 0)$ of the original system within the machine precision (produced using data presented in [72] with permission). An unbounded region is clearly visible.

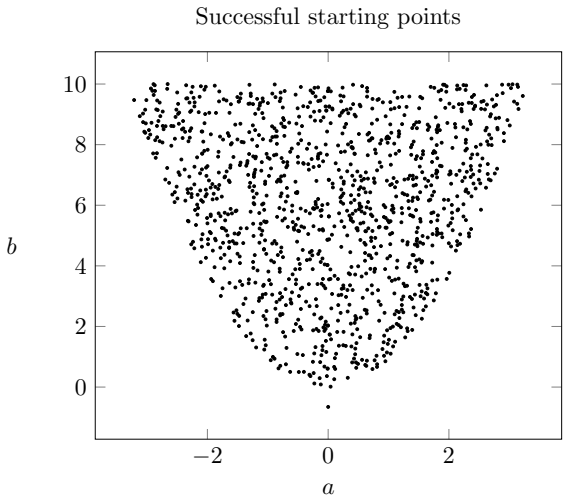


Figure 9. Scatter plot of some $(a, b) \in \mathbb{R}^2$ for which the Newton homotopy (3.7) was successful in obtaining the singular solution $(x, y) = (0, 0)$ of the target Griewank-Osborne system (3.6).

Proof. (Appendix of [73]) Eliminating y from $H(x, y, t) = (0, 0)$ provides

$$h(x,t) = -\frac{3}{16}x^3 - 2x(x^2 + tb - ta^2) - t\left(\frac{29}{16}a^3 - 2ab)\right).$$

As a cubic polynomial in x only, its discriminant is

$$\Delta = -4 - \frac{3}{16}[-2t(b-a^2)]^3 - 27\left(-\frac{3}{16}\right)^2\left[-t\left(\frac{29}{16}a^3 - 2ab)\right)\right]^2.$$

Substituting $b = \dfrac{119+r}{128}a^2$ in the above yields

$$\Delta = \frac{t^2a^6}{2^{20}}\left[-243r^2 - 3(r^3 - 27r^2 + 243r - 729)t - 1458r - 2187\right].$$

We are interested in the sign of Δ as t goes from 1 to 0. Note that

$$\frac{2^{20}\Delta}{t^2a^6} = -243r^2 - 3(r^3 - 27r^2 + 243r - 729)t - 1458r - 2187$$

is a linear function in t. A straightforward calculation shows that this function takes negative values at $t = 0$ and $t = 1$ for any $r > 0$. Therefore $\Delta < 0$

for all $t \in [0,1]$ and $r > 0$. Since $b = \frac{119+r}{128}a^2$, so for $U = \{(a,b) \mid b > \frac{119}{128}a^2\}$ and $t \in [0,1]$, the discriminant of $h(x,t)$, as a univariate polynomial in x, is negative indicating that it has a unique real root for each fixed t. Combining with the smoothness of H, the equation $h(x,t) = 0$ defines a single smooth curve in $\mathbb{R} \times [0,1]$ and, by extension, $H(x,y,t) = (0,0)$ also defines a single smooth curve in $\mathbb{R}^2 \times [0,1]$ which necessarily connects the starting point and the target solution $(0,0)$ of the Griewank-Osborne system. □

Though the smoothness condition for the Newton homotopy is difficult to establish in general, toward the real world problems in physics, chemistry, and a variety of other fields the great usefulness of the Newton homotopy is undeniable as the example in the following section shows.

3.4.2. Obtaining multiple solutions using one curve. When a local method such as Newton's iterations is used to solve a nonsingular system, with an appropriate choice of the starting point, the method may converge to a solution, and the story ends there. A distinct advantage of the (real) homotopy continuation methods is their ability to obtain solutions one after another by tracing just one solution curve.

As noted in §3, for a homotopy $H(\mathbf{x},t)$, if the solution curve defined by $H(\mathbf{x},t) = \mathbf{0}$ passes through $t = 1$, then a solution of the target system is obtained. However, it is possible for the same curve to pass through $t = 1$ multiple times, each time producing a distinct solution. In many real world problems, a large number of solutions can lie on the same solution curve, making homotopy method a particularly appealing choice. Below we examine a specific example from theoretical physics that has been studied with Newton homotopy method in [72].

Example 3.10 (The nearest-neighbor ϕ^4 model [72]). The two-dimensional nearest-neighbor ϕ^4 model is an important model in theoretical physics that has been widely studied. For an $N \in \mathbb{Z}^+$, the model, in N^2 variables $\mathbf{x} = (x_{00}, x_{01}, x_{10}, \ldots, x_{NN})$, is the real-valued "potential" function given by

$$V(\mathbf{x}) = \sum_{\Lambda} \left(\frac{3}{5 \cdot 4!} x_{ij}^4 - x_{ij}^2 + \frac{J}{4} \sum_{(i',j') \in \mathcal{N}(i,j)} (x_{ij} - x_{i'j'})^2 \right), \tag{3.8}$$

where $\Lambda = \{(i,j)\}_{i,j=0}^{N-1} \subset \mathbb{Z}^2$ is a square lattice with N^2 points, the set $\mathcal{N}(i,j) = \{(i+1,j),(i-1,j),(i,j+1),(i,j-1)\}$ modulo Λ is the subset of Λ that consists of four nearest neighboring points of (i,j), and J is a

chosen real value. The stationary system of this model is the system of N^2 equations in N^2 variables obtained by setting each partial derivative of V to zero, i.e.,

$$\frac{\partial V(\mathbf{x})}{\partial x_{ij}} = \frac{1}{10}x_{ij}^3 + (4J-2)x_{ij} - J \sum_{(i',j')\in\mathcal{N}(i,j)} x_{i'j'} = 0. \tag{3.9}$$

for each pair of $i, j = 0, \ldots, N-1$. Given the physical context, only real solutions are needed.

A variety of computational methods have been used to study this model. However, this family of problems, parametrized by $J \in \mathbb{R}$, poses a particularly tough computational challenge, especially for larger N. In particular, it is shown in [74] that the total number of solutions of the above system in $\mathbb{C}^{N^2}$ always equals to its "total degree" 3^{N^2} which grows quickly as N increases. For example, at $N = 5$ the total number of complex solutions is 847288609443 and for $N = 6$ the number exceeds 10^{17}. Direct computation of all complex solutions is clearly infeasible with current technology for larger N values. However, numerical experiments with this system reveals an interesting property: by varying the parameter J from 0 to 1, the number of real solutions decreases drastically while the number of complex solutions stays the same. In particular, for J values close to 1, only an extremely small fraction of the solutions are real. In this case, the Newton homotopy, which directly targets the real solutions, has a clear advantage over methods that compute all complex solutions in the first place, followed by filtering out all the real solutions.

In practical experiments, the Newton homotopy, as defined in (3.2), was applied to the above system with varying values for N and J. From a *single* randomly chosen starting point, the Newton homotopy can find multiple real solutions. Tables 1 and 2 show the capability and efficiency of the Newton homotopy in finding multiple real solutions for a range of N and J values. Remarkably, *all* real solutions were found in the cases with $N = 3$ and $J = 0.9, 0.8, 0.7$. Each of these systems has only 3 real solutions, and all of them can be found by tracing just one single solution curve of the Newton homotopy with a generically chosen starting point. The time consumption information is listed in Table 1.

Fixing $J = 0.9$, Newton homotopy in solving (3.9) with increasing value of $N = 3, 4, 5, 6, 7$ have also been investigated. Table 2 exhibits the strength of the Newton homotopy: a large number of real solutions can be found very quickly. In particular, when $N = 6$ and $N = 7$, these two systems have a total number of more than 10^{17} and 10^{23} complex solutions respectively. Solving

J	N.o. $\mathbb{R}$-solutions obtained	% $\mathbb{R}$-solutions	Time
0.9	3	(All) 100%	0.009s
0.8	3	(All) 100%	0.009s
0.7	3	(All) 100%	0.012s

Table 1. Number of real solutions obtained and timing information when the Newton homotopy is applied to the ϕ^4 model with $N = 3$ and varying values of J. A *single* randomly chosen starting point was used. The timing information is computed based on the performance on a workstation with Intel Core i5-3570K running at 3.4GHz.

N	N.o. $\mathbb{R}$-solutions	% $\mathbb{R}$-solutions	Time
3	3	(All) 100%	0.01s
4	83	(All) 100%	0.62s
5	102	-	2.00s
6	208	-	23.95s
7	358	-	29.66s

Table 2. Number of real solutions obtained and the time consumption when the Newton homotopy is applied to the ϕ^4 model [72] with $J = 0.9$ and varying values of $N = 3, 4, 5, 6$. A *single* randomly chosen starting point was used. The timing information is computed based on the performance on a workstation with Intel Core i5-3570K running at 3.4GHz.

all those complex solutions first seems particularly infeasible. However, the Newton homotopy, using a single starting point, can find 208 and 358 real solutions for these two systems respectively *within 30 seconds*. Figure 10 showcases this ability of the Newton homotopy. In the Figure, the t-value (horizontal axis) is plotted against the arc-length (vertical axis). Notice the numerous crossing of the solution curve with the plane at $t = 1$. Actually, each of them produces a distinct real solution of the target system.

3.4.3. Preserving Morse indices for gradient systems. In applications, one important source of nonlinear systems of equations is the *gradient systems* derived from partial derivatives of real-valued functions. More precisely, the corresponding gradient system of a real-valued differentiable

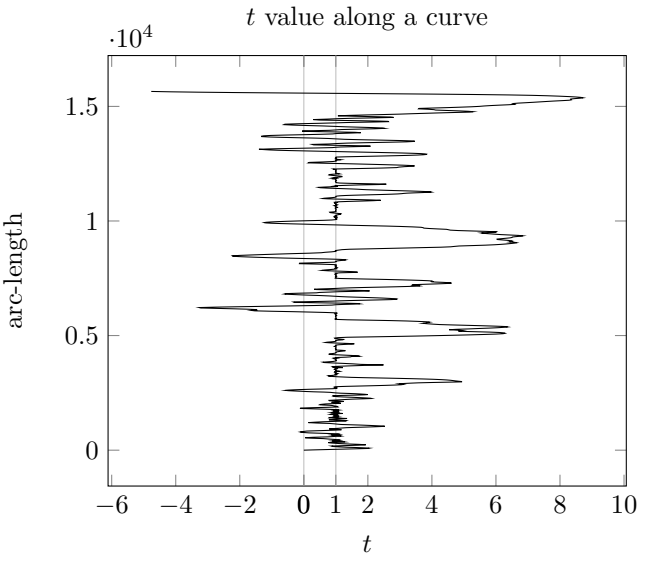


Figure 10. The t value along a solution curve defined by $H(\mathbf{x}, t) = \mathbf{0}$ where H is the Newton homotopy for the nearest neighborhood ϕ^4 problem (3.9) with $N = 6$ and $J = 0.9$. The light vertical line in the middle represents the plane at $t = 1$ whose intersections with the curve produces real solutions of the system.

function $V : \mathbb{R}^n \to \mathbb{R}$ in the variables $\mathbf{x} = (x_1, \ldots, x_n)$ is the system of equations

$$\begin{aligned} \frac{\partial V}{\partial x_1} &= 0 \\ &\vdots \\ \frac{\partial V}{\partial x_n} &= 0. \end{aligned}$$

For brevity, we also use the notation $DV(\mathbf{x}) = \mathbf{0}$ for this system. The solutions of this system are known as *critical points* of V, and the problem of finding such critical points is a fundamental problem that arises in such diverse fields as physics, economics, engineering, optimal control, etc. For simplicity, V is assumed to be smooth in the following discussion. Those critical points are classified by the eigenvalues of its Hessian matrix $D^2V = D(DV)$. With V being smooth, D^2V, as an $n \times n$ matrix with real entries, is necessarily symmetric, hence all its eigenvalues are real. By Sylvester's Law of

Inertia, the signs of these eigenvalues are independent from the coordinate systems. A critical point is said to be *degenerate* if its Hessian D^2V has a zero eigenvalue and *nondegenerated* otherwise. Moreover the number of negative eigenvalues is known as the *Morse-index* of the critical point. This Morse-index is invariant under a nonsingular smooth change of coordinate and hence a geometric property of the critical point itself. If we further assume V to be locally analytic near a critical point $\mathbf{x}^*$, then

$$V(\mathbf{x}) = V(\mathbf{x}^*) + DV(\mathbf{x}^*) \cdot (\mathbf{x} - \mathbf{x}^*) + (\mathbf{x} - \mathbf{x}^*)^\top \cdot D^2V(\mathbf{x}^*) \cdot (\mathbf{x} - \mathbf{x}^*) + \mathcal{O}(\|\mathbf{x} - \mathbf{x}^*\|^3).$$

Evidently, a nondegenerated critical point $\mathbf{x}^*$ of V is a local minimum if and only if the Morse-index of $D^2V(\mathbf{x}^*)$ is 0, that is, all its eigenvalues are positive. Local minima are of great importance in many applications. For example, if V models a potential energy of a physical system, the local minima of V then correspond to stable states of the system. Critical points of other Morse-indices also have meaningful interpretation in certain types of applications.

While there are many numerical methods for locating such critical points, the homotopy continuation method may have a great advantage in many situations. In particular, when one considers a family of real-valued functions parametrized by a new variable t:

$$\hat{V}(\mathbf{x}, t) = (1 - t)U(\mathbf{x}) + t\,V(\mathbf{x}) \tag{3.10}$$

for some smooth function $U : \mathbb{R}^n \to \mathbb{R}$. Clearly, $\hat{V}(\mathbf{x}, 0) \equiv U(\mathbf{x})$, $\hat{V}(\mathbf{x}, 1) = V$, and $\hat{V}(\mathbf{x}, t)$ can be interpreted as an objective function that smoothly deforms over time (represented by the variable t) from U to V. Now, the function

$$H(\mathbf{x}, t) := (1 - t)DU(\mathbf{x}) + t\,DV(\mathbf{x}) \tag{3.11}$$

can be taken as a homotopy between DU and DV. The system of equations $H(\mathbf{x}, t) = \mathbf{0}$ then defines the critical points of the "deforming objective function" given in (3.10). If one critical points of U is known, one can start from this point and trace the critical point of $\hat{V}(\mathbf{x}, t)$ as t varies. When t reaches 1, then a critical point of $\hat{V}(\mathbf{x}, 1) \equiv V(\mathbf{x})$ is located.

The potential advantage of this homotopy-based approach is the possibility of preserving the Morse-index. Suppose the homotopy satisfies the smoothness condition. Thus, $H^{-1}(\{\mathbf{0}\}) \subset \mathbb{R}^{n+1}$ is a disjoint union of smooth curves. Let $(\mathbf{x}(s), t(s))$ be the solution curve defined by $H(\mathbf{x}, t) = \mathbf{0}$ that

passes through a given starting point $\mathbf{x}^0$ and parametrized by arc-length s. Obviously, eigenvalues of $H_{\mathbf{x}} = D_{\mathbf{x}}^2 \hat{V}$ vary continuously along the curve, and since eigenvalues of $H_{\mathbf{x}}$ are all real, they must reach zero before changing signs. As a consequence, if $H_{\mathbf{x}}$ remains nonsingular (i.e., the product of eigenvalues remain nonzero) along the solution curve, then the Morse-index is preserved. Geometrically, if the solution curve contains no turning point (see Figure 2), then the Morse-index is preserved along the entire curve.

The effectiveness of this scheme depends on the choice of the starting function U in (3.11). A particularly common choice of U is

$$U = \frac{1}{2}(x_1 - a_1)^2 + \cdots + \frac{1}{2}(x_n - a_n)^2 = \frac{1}{2}\|\mathbf{x} - \mathbf{a}\|_2^2,$$

which has a unique local minimum at $\mathbf{x} = \mathbf{a}$. Also, at $\mathbf{x} = \mathbf{a}$, $DU = (x_1 - a_1, \ldots, x_n - a_n) = \mathbf{x} - \mathbf{a}$, and the homotopy, as constructed in (3.11), is therefore

$$H(\mathbf{x}, t) = (1 - t)(\mathbf{x} - \mathbf{a}) + t\, DV(\mathbf{x})$$

which is precisely the fixed point homotopy (3.1) applied to solving the nonlinear system $DV(\mathbf{x}) = \mathbf{0}$. If the solution curve defined by $H(\mathbf{x}, t) = \mathbf{0}$ never encounters any turning points, then the Morse-index of the critical point is preserved along the curved. Since the starting point $\mathbf{x} = \mathbf{a}$ at $t = 0$ is the unique local minimum (with Morse-index 0), the resulting solution of the target system $DV(\mathbf{x}) = H(\mathbf{x}, 1) = \mathbf{0}$ must therefore be a local minimum of $V(\mathbf{x})$.

More generally, for any desired Morse-index m with $0 \leq m \leq n$, one may construct the starting objective function

$$\begin{aligned} U = &-\frac{1}{2}(x_1 - a_1)^2 - \cdots - \frac{1}{2}(x_m - a_m)^2 \\ &+ \frac{1}{2}(x_{m+1} - a_{m+1})^2 + \cdots + \frac{1}{2}(x_n - a_n)^2. \end{aligned}$$

Clearly, $\mathbf{x} = \mathbf{a}$ is the unique critical point of U and its Morse-index is m. Consequently, the homotopy (3.11) (defined with this choice of U as the starting objective function) can then locate a critical point of Morse-index m if the solution curve reaches the hyperplane at $t = 1$ without encountering a turning point. General homotopy constructions exploiting this feature have been studied in [2, 6].

Remark 3.11. Though the procedure described above requires the solution curve to never encounter a turning point in order to preserve the Morse-index. Computational experiments [19] suggests that the Morse-index may be preserved under much more relaxed conditions.

4. Homotopy continuation methods for finding all isolated complex solutions of polynomial systems

Though homotopy continuation methods in numerical computation were first developed as tools for finding solutions to nonlinear systems of equations, in late 1970s' Garcia and Zangwill [33] as well as Drexler [25] independently presented theorems suggesting that homotopy continuation methods could be used to find the full set of isolated zeros of polynomial systems numerically. Finding solutions of polynomial systems is a classical problem that has occupied a special place in mathematics over its long history. Moreover, this is an area where numerical computations arise almost naturally since by Galois theory explicit formulas for the solutions are unlikely to exist. To deal with this problem, the homotopy continuation method has attracted a considerable attention in the last few decades. It has been deeply developed and proved to be a reliable and efficient numerical algorithm for approximating all isolated zeros of polynomial systems and identifying positive dimensional solutions sets with their degrees.

This part of the article will focus on the methods for finding *all* isolated solutions of system of polynomial equations, or polynomial systems, rather than finding one or some solutions as in the real cases discussed in previous sections. Obviously, to have any hope for finding all the solutions, the working framework must be in the complex spaces.

4.1. An important feature of the homotopy constructed in $\mathbb{C}^n$

For complex space $\mathbb{C}$, $\mathbb{C}[x_1, \ldots, x_n]$ denotes the set of all polynomials in the n variables $x_1, \ldots, x_n$ with complex coefficients, which forms a commutative ring under polynomial addition and multiplication. Given a system of n polynomials $P = (p_1, \ldots, p_n)$ where $p_i \in \mathbb{C}[x_1, \ldots, x_n]$ for $i = 1, \ldots, n$, to find all isolated solutions $\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{C}^n$ to the system of n equations $P(\mathbf{x}) = \mathbf{0}$, we construct, as before, a smooth homotopy $H(\mathbf{x}, t) : \mathbb{C}^n \times [0, 1] \to \mathbb{C}^n$ to deform P to a polynomial system $G = (g_1, \ldots, g_n)$ where $g_i \in \mathbb{C}[x_1, \ldots, x_n]$

for $i = 1, \ldots, n$ having known (or easily found) zeros, namely,

$$H(\mathbf{x}, 0) = G(\mathbf{x}) \quad \text{and} \quad H(\mathbf{x}, 1) = P(\mathbf{x}).$$

By identifying $\mathbb{C}^n$ with $\mathbb{R}^{2n}$ via the map

$$(z_1, \ldots, z_n) \mapsto (\operatorname{Re} z_1, \operatorname{Im} z_1, \ldots, \operatorname{Re} z_n, \operatorname{Im} z_n), \tag{4.1}$$

H can be considered as a map from $\mathbb{R}^{2n} \times [0, 1]$ to $\mathbb{R}^{2n}$. With this interpretation, we still say $\mathbf{0} \in \mathbb{C}^n$ is a regular value of H, or H is regular, if the Jacobian matrix $DH(\mathbf{x}, t) \in M_{2n \times 2n+1}(\mathbb{R})$ with respect to both $\mathbf{x}$ and t is of rank $2n$ for all $(\mathbf{x}, t) \in \mathbb{R}^{2n} \times [0, 1]$ satisfying $H(\mathbf{x}, t) = \mathbf{0}$. When $\mathbf{0}$ is a regular value of H, then $H(\mathbf{x}, t) = \mathbf{0}$ defines (disjoint) smooth solution curves in $\mathbb{R}^{2n+1}$, and any curve $\gamma \subset \mathbb{C}^n \times [0, 1]$ defined by the homotopy $H(\mathbf{x}, t) = \mathbf{0}$ can be parametrized by the arc length s. In the content of solving polynomial systems in $\mathbb{C}^n$, there is a special feature that has a profound effect which will significantly alter the choices of the underlying numerical methods. We will show below that for any point on the smooth homotopy curve $(\mathbf{x}(s), t(s))$ of $H(\mathbf{x}, t) = \mathbf{0}$ parametrized by the arc length s, $\frac{dt}{ds}$ is always nonzero, and therefore $\frac{dt}{ds} > 0$. Meaning: those curves do not "turn back in t". In other words, they extend across the interval $0 \leq t < 1$ and can always be parametrized by t. Accordingly, standard procedures in tracing general homotopy paths need to be adjusted to capitalize this special feature.

Lemma 4.1. *Regard the $n \times n$ complex matrix M as a linear transformation of complex variables $(x_1, \ldots, x_n)$ in $\mathbb{C}^n$ into itself. If this transformation is regarded as one on the space $\mathbb{R}^{2n}$ of real variables $(u_1, v_1, \ldots, u_n, v_n)$ where $x_j = u_j + \mathbf{i}\, v_j, j = 1, \ldots, n$, (here, $\mathbf{i} = \sqrt{-1}$) and is represented by the $2n \times 2n$ real matrix N then*

$$\det N = \left| \det M \right|^2 \geq 0$$

and

$$\dim_{\mathbb{R}}(\ker N) = 2 \times \dim_{\mathbb{C}}(\ker M) \text{ is even.}$$

Here, $\dim_{\mathbb{R}}$ *and* $\dim_{\mathbb{C}}$ *refer to real and complex dimension respectively.*

Proof. The relation between M and N is the following: if the (j, k)-entry of M is the complex number $m_{jk} = \xi_{jk} + \mathbf{i}\, \eta_{jk}$, and N is written in block

form as an $n \times n$ array of 2×2 blocks, then the (j,k)-block of N is the real matrix

$$\begin{pmatrix} \xi_{jk} & -\eta_{jk} \\ \eta_{jk} & \xi_{jk} \end{pmatrix}.$$

Denote this relation by $\alpha(M) = N$. It is clear that $\alpha(AB) = \alpha(A)\alpha(B)$ for complex matrices A and B, and $\alpha(A^{-1}) = \alpha(A)^{-1}$.

Now when M is upper triangular, the assertion is immediate. For general M, there exists complex nonsingular matrix A for which $A^{-1}MA$ is upper triangular. Because

$$\alpha(A^{-1}MA) = \alpha(A^{-1})\alpha(M)\alpha(A) = \alpha(A)^{-1}N\alpha(A),$$

we have

$$\det(\alpha(A^{-1}MA)) = \det(\alpha(A))^{-1} \times \det N \times \det(\alpha(A)) = \det N.$$

The assertion holds, since

$$\det(\alpha(A^{-1}MA)) = \left|\det(A^{-1}MA)\right|^2 = \left|\det M\right|^2. \qquad \square$$

Proposition 4.2. *If $(\mathbf{x}_0, t_0)$ is a point on any smooth homotopy paths $(\mathbf{x}(s), t(s))$ of the homotopy $H(\mathbf{x}, t) = \mathbf{0}$ defined on $\mathbb{C}^n \times [0,1]$ with $t_0 \in [0,1)$, then $H_{\mathbf{x}}(\mathbf{x}_0, t_0)$ is nonsingular. Hence, $\dfrac{dt}{ds} \neq 0$ at $(\mathbf{x}_0, t_0)$.*

Proof. Regard H as a map from $\mathbb{R}^{2n} \times \mathbb{R}$ to $\mathbb{R}^{2n}$. Since the $2n \times (2n+1)$ Jacobian matrix $DH = [H_{\mathbf{x}}, H_t]$ must be of full rank at $(\mathbf{x}_0, t_0)$ (otherwise it would be a bifurcation point [5]), its kernel is at most one-dimensional. By the above lemma, the matrix $H_{\mathbf{x}}$ must have zero kernel, so it is nonsingular. Hence, $\dfrac{dt}{ds} \neq 0$ at $(\mathbf{x}_0, t_0)$, because

$$H_{\mathbf{x}}\frac{d\mathbf{x}}{ds} + H_t\frac{dt}{ds} = 0. \qquad \square$$

4.2. Path tracking in $\mathbb{C}^n$

So, homotopy paths defined by $H(\mathbf{x}, t) = \mathbf{0}$ in $\mathbb{C}^n \times [0,1]$ can always be parametrized by t. Let $\mathbf{x}(t)$, $0 \leq t \leq 1$, be a path in $\mathbb{C}^n$ satisfying the homotopy equation $H(\mathbf{x}, t) = \mathbf{0}$, namely,

$$H(\mathbf{x}(t), t) = \mathbf{0} \qquad 0 \leq t \leq 1. \tag{4.2}$$

In the following, we shall denote $\frac{d\mathbf{x}}{dt}$ by $\mathbf{x}'(t)$. Now, differentiating (4.2) with respect to t yields

$$H_{\mathbf{x}}\mathbf{x}'(t) + H_t = \mathbf{0} \qquad 0 \le t \le 1,$$

or

$$\mathbf{x}'(t) = -H_{\mathbf{x}}^{-1} H_t \qquad 0 \le t \le 1. \tag{4.3}$$

This ordinary differential equation is commonly known as the Davidenko differential equation [24], which forms the basis of the numerical path tracking algorithms with which one can trace a solution path of the homotopy equation (4.2) from its starting point. While any numerical ordinary differential equation solver can, in principle, be applied to Equation (4.3) and thus be used for path tracking, just like in the real curve tracing discussed in §2 a special class of *predictor-corrector* method is generally preferred. In such a scheme, an efficient but potentially inaccurate "predictor" accounts for producing a rough estimate of the next point on the path using the information of known points on the path. Then a series of Newton-like "corrector" iterations is employed to bring the point back to the path approximately.

One of the most basic predictor-corrector configuration is the duet of Euler's method with Newton's iterations. For a fixed $0 \le t_0 < 1$, to proceed from a point $\mathbf{x}(t_0)$ that is approximately on the path $\mathbf{x}(t)$, one takes the following steps:

- **Euler Prediction:**

  For an adaptive step size $\delta > 0$, let $t_1 = t_0 + \delta < 1$ and

$$\tilde{\mathbf{x}}(t_1) = \mathbf{x}(t_0) + \delta\, \mathbf{x}'(t_0). \tag{4.4}$$

- **Newton's Correction:**

  For fixed t_1, $H(\mathbf{x}, t_1) = \mathbf{0}$ becomes a system of n equations in n unknowns. So, Newton's iteration can then be employed to solve the solution of $H(\mathbf{x}, t_1) = \mathbf{0}$ with starting point $\tilde{\mathbf{x}}(t_1)$, i.e.,

$$\mathbf{x}^{(m+1)} = \mathbf{x}^{(m)} - [H_{\mathbf{x}}(\mathbf{x}^{(m)}, t_1)]^{-1} H(\mathbf{x}^{(m)}, t_1), \quad m = 0, 1, \ldots \tag{4.5}$$

  with $\mathbf{x}^{(0)} = \tilde{\mathbf{x}}(t_1)$. When the iteration fails to converge, the prediction step will be repeat with $\delta \leftarrow \frac{\delta}{2}$. Eventually, an approximate value of $\mathbf{x}(t_1)$ can be determined until certain stopping criteria are met

5. Complex linear homotopies

Let $P(\mathbf{x}) = (p_1(\mathbf{x}), \ldots, p_n(\mathbf{x})) = \mathbf{0}$ be a system of n polynomial equations in n unknowns $\mathbf{x} = (x_1, \ldots, x_n)$, we want to find all isolated solutions of

$$\begin{aligned} p_1(x_1, \ldots, x_n) &= 0, \\ &\vdots \\ p_n(x_1, \ldots, x_n) &= 0 \end{aligned} \tag{5.1}$$

in $\mathbb{C}^n$. In the early stage, the homotopy continuation method for solving (5.1) is to define a trivial system $Q(\mathbf{x}) = (q_1(\mathbf{x}), \ldots, q_n(\mathbf{x})) = \mathbf{0}$ and then follow the curves in the real variable t which make up the solution set of

$$H(\mathbf{x}, t) = (1 - t)Q(\mathbf{x}) + tP(\mathbf{x}) = \mathbf{0}. \tag{5.2}$$

More precisely, if $Q(\mathbf{x}) = \mathbf{0}$ is chosen correctly, the following three properties hold:

Property 0 (Triviality): The solutions of $Q(\mathbf{x}) = \mathbf{0}$ are known.

Property 1 (Smoothness): The solution set of $H(\mathbf{x}, t) = \mathbf{0}$ for $0 \leq t < 1$ consists of a finite number of smooth paths, each parametrized by t in $[0, 1)$.

Property 2 (Accessibility): Every isolated solution of $H(\mathbf{x}, 1) = P(\mathbf{x}) = \mathbf{0}$ can be reached by some path originating at $t = 0$. It follows that this path starts at a solution of $H(\mathbf{x}, 0) = Q(\mathbf{x}) = \mathbf{0}$.

When the three properties hold, the solution paths can be traced from the initial points (known because of property 0) at $t = 0$ to all solutions of the original problem $P(\mathbf{x}) = \mathbf{0}$ at $t = 1$ using standard numerical techniques (Prediction-Correction steps given in the last section for instance).

Several authors have suggested choices of $Q(\mathbf{x})$ that satisfy the three properties (See [21, 57, 79, 115, 117] for a partial list). A typical suggestion is

$$\begin{aligned} q_1(\mathbf{x}) &= a_1 x_1^{d_1} - b_1, \\ &\vdots \\ q_n(\mathbf{x}) &= a_n x_n^{d_n} - b_n, \end{aligned} \tag{5.3}$$

where $d_1, \ldots, d_n$ are the degrees of $p_1(\mathbf{x}), \ldots, p_n(\mathbf{x})$ respectively and a_j, b_j are random complex numbers (and therefore nonzero with probability one).

So in one sense, the original problem we posed is solved. All isolated solutions of $P(\mathbf{x}) = \mathbf{0}$ are found at the end of the $d_1 \times \cdots \times d_n$ paths that make up the solution set of $H(\mathbf{x}, t) = \mathbf{0}$, $0 \leq t < 1$. The number $d_1 \times \cdots \times d_n$ is commonly known as the *total degree* or the *Bézout number* of the system. Sometimes it is regarded as the *expected* number of isolated solutions.

The book by A. Morgan [80] detailed many aspects of the above approach. A major part of this section will focus on the development afterwards that makes this method more convenient to apply.

The reason the problem is not satisfactorily solved by the above construction is the existence of *extraneous paths*. Although the above method produces $d := d_1 \times \cdots \times d_n$ paths, the system $P(\mathbf{x}) = \mathbf{0}$ may have fewer than d isolated solutions (even counting multiplicity). We call such a system *deficient*. In this case, some of the paths produced by the above method will be extraneous paths.

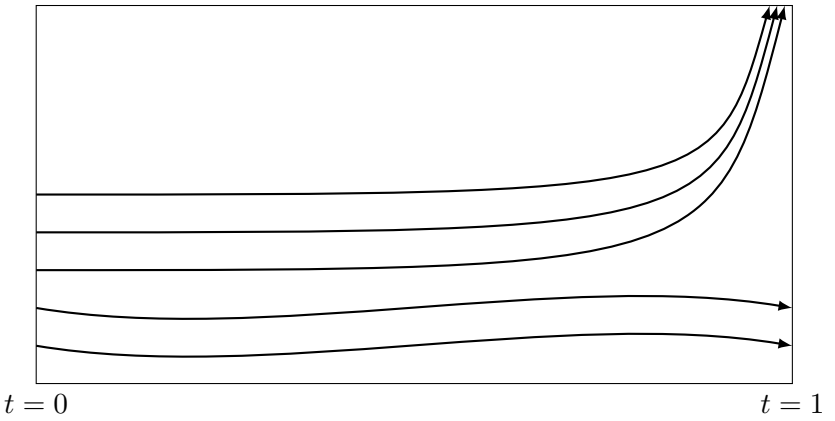


Figure 11. Extraneous homotopy paths may diverge to infinity

More precisely, even though Properties 0-2 imply that each isolated solution of $P(\mathbf{x}) = \mathbf{0}$ will lie at the end of a solution path, it is also consistent with those properties that some of the paths may diverge to infinity as the parameter t approaches 1 (the smoothness property rules this out for $t \to t_0 < 1$). In other words, it is quite possible for $Q(\mathbf{x}) = \mathbf{0}$ to have more isolated solutions than $P(\mathbf{x}) = \mathbf{0}$. In this case, some of the paths leading from roots of $Q(\mathbf{x}) = \mathbf{0}$ are extraneous, and diverge to infinity when $t \to 1$ (see Figure 11). Empirically, we find that most systems arising in applications are deficient. A great majority of the systems have fewer than, and in some cases only a small fraction of, the "expected number " of solutions.

For a typical example of this sort, let's look at the following famous Cassou-Noguès system [65]

$$
\begin{aligned}
p_1 &= 15b^4cd^2 + 6b^4c^3 + 21b^4c^2d - 144b^2c - 8b^2c^2e - 28b^2cde \\
&\quad - 648b^2d + 36b^2d^2e + 9b^4d^3 - 120, \\
p_2 &= 30b^4c^3d - 32cde^2 - 720b^2cd - 24b^2c^3e - 432b^2c^2 + 576ce \\
&\quad - 576de + 16b^2cd^2e + 16d^2e^2 + 16c^2e^2 + 9b^4c^4 + 39b^4c^2d^2 \\
&\quad + 18b^4cd^3 - 432b^2d^2 + 24b^2d^3e - 16b^2c^2de - 240c + 5184, \\
p_3 &= 216b^2cd - 162b^2d^2 - 81b^2c^2 + 1008ce - 1008de + 15b^2c^2de \\
&\quad - 15b^2c^3e - 80cde^2 + 40d^2e^2 + 40c^2e^2 + 5184, \\
p_4 &= 4b^2cd - 3b^2d^2 - 4b^2c^2 + 22ce - 22de + 261.
\end{aligned}
\tag{5.4}
$$

Since $d_1 = 7, d_2 = 8, d_3 = 6$ and $d_4 = 4$ for this system, the system $Q(\mathbf{x})$ in (5.3) will produce $d_1 \times d_2 \times d_3 \times d_4 = 7 \times 8 \times 6 \times 4 = 1344$ paths for the homotopy in (5.2). However, the system (5.4) has only 16 isolated zeros. Consequently, a major fraction of the paths are extraneous. Sending out 1344 paths in search of 16 solutions is a highly wasteful computation.

The choice of $Q(\mathbf{x})$ in (5.3) to solve the system $P(\mathbf{x}) = \mathbf{0}$ requires an amount of computational effort proportional to $d_1 \times \cdots \times d_n$ and roughly, proportional to the size of the system. We would like to derive methods for solving deficient systems for which the computational effort is instead proportional to the actual number of isolated solutions.

For deficient systems, there are some partial results that use algebraic geometry to reduce the number of extraneous paths with various degrees of success.

5.1. Random product homotopy

For a specific example that is quite simple, consider the algebraic eigenvalue problem,

$$A\mathbf{x} = \lambda\mathbf{x}$$

where

$$A = \begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & & \\ a_{n1} & \cdots & a_{nn} \end{bmatrix}$$

is an $n \times n$ matrix. This problem is actually an n polynomial equations in the $n+1$ variables $\lambda, x_1, \ldots, x_n$:

$$\begin{aligned} p_1 &= \lambda x_1 - (a_{11}x_1 + \cdots + a_{1n}x_n) = 0 \\ &\vdots \\ p_n &= \lambda x_n - (a_{n1}x_1 + \cdots + a_{nn}x_n) = 0. \end{aligned} \tag{5.5}$$

Augmenting the system with a linear equation

$$p_{n+1} = c_1x_1 + \cdots + c_nx_n + c_{n+1} = 0$$

where $c_1, \ldots, c_{n+1}$ are chosen at random, we have a polynomial system of $n+1$ equations in $n+1$ variables. Write $\mathbf{y} = (\lambda, \mathbf{x}) = (\lambda, x_1, \ldots, x_n)$. This system has total degree 2^n. Thus the "expected number of solutions" is 2^n, and the classical homotopy continuation method using the start system $Q(\mathbf{y}) = \mathbf{0}$ in (5.3) sends out 2^n paths from 2^n trivial starting points. However, the system $P(\mathbf{y}) = (p_1(\mathbf{y}), \ldots, p_{n+1}(\mathbf{y})) = \mathbf{0}$ has only n isolated solutions (even fewer for special choices of coefficients). This is a deficient system, at least $2^n - n$ paths will be extraneous. It is never known from the start which of the paths will end up to be extraneous, so they must all be followed to the end, representing wasted computation.

The random product homotopy was developed in [61, 62] to alleviate this problem. According to that technique, a more efficient choice for the trivial system $Q(\mathbf{y}) = \mathbf{0}$ is

$$\begin{aligned} q_1 &= (\lambda + e_{11})(x_1 + e_{12}) \\ q_2 &= (\lambda + e_{21})(x_2 + e_{22}) \\ &\vdots \\ q_n &= (\lambda + e_{n1})(x_n + e_{n2}) \\ q_{n+1} &= c_1x_1 + \cdots + c_nx_n + c_{n+1} \end{aligned} \tag{5.6}$$

where complex numbers e_{ij} for $i = 1, \ldots, n$, $j = 1, 2$ and c_k, $k = 1, \ldots, n+1$ are generically chosen. Set

$$H(\mathbf{y}, t) = (1-t)\, c\, Q(\mathbf{y}) + tP(\mathbf{y}) \quad \text{for generic } c \in \mathbb{C}.$$

It is clear by inspection that $Q(\mathbf{y}) = \mathbf{0}$ has exactly n roots. Thus there are only n paths starting from n starting points for this choice of homotopy. Moreover, it is proved in [62] that Properties 0-2 hold for almost all complex

numbers e_{ij} and c. Thus all solutions of $P(\mathbf{y}) = \mathbf{0}$ are found at the end of the n paths. The result of [62] is then a mathematical result (that there can be at most n solutions to (5.5)) and the basis of a numerical procedure for approximating the solutions.

The reason this works is quite simple. The solution paths of (5.2) which do not proceed to a solution of $P(\mathbf{y}) = \mathbf{0}$ in $\mathbb{C}^{n+1}$ diverge to infinity. If the system (5.2) is viewed in projective space

$$\mathbb{CP}^n = \{(x_0, \ldots, x_n) \in \mathbb{C}^{n+1} \backslash (0, \ldots, 0)\} / \sim$$

where the equivalent relation "$\sim$" is given by $\mathbf{x} \sim \mathbf{y}$ if $\mathbf{x} = c\,\mathbf{y}$ for some nonzero $c \in \mathbb{C}$, the diverging paths simply converge to a "point at infinity" in $\mathbb{CP}^n$.

For a polynomial $f(x_1, \ldots, x_n)$ of degree d, denote its associated homogeneous polynomial by

$$\widetilde{f}(x_0, x_1, \ldots, x_n) = x_0^d\, f\left(\frac{x_1}{x_0}, \ldots, \frac{x_n}{x_0}\right).$$

The solutions of $f(\mathbf{x}) = 0$ "at infinity" are those zeros of $\widetilde{f}$ in $\mathbb{CP}^n$ with $x_0 = 0$ and the remaining zeros of $\widetilde{f}$ with $x_0 \neq 0$ are the solutions of $f(\mathbf{x}) = 0$ in $\mathbb{C}^n$ when x_0 is set to be 1.

Viewed in projective space $\mathbb{CP}^{n+1}$ the system $P(\mathbf{y}) = \mathbf{0}$ in (5.5) has some roots at infinity. The roots at infinity make up a nonsingular variety, specifically the linear space $\mathbb{CP}^{n-2}$ defined by $x_0 = \lambda = 0$ and $c_1 x_1 + \cdots + c_n x_n = 0$. A Chern class formula from intersection theory ([28], 9.1.1, 9.1.2) shows that the contribution of a linear variety of solutions of dimension e to the "total degree" $d_1 \times \cdots \times d_n$, or the total expected number of solutions, of the system is at least s, where s is the coefficient of t^e in the Maclaurin series expansion of

$$(1+t)^{e-n} \prod_{j=1}^{n} (1 + d_j t).$$

In our case, $d_1 = \cdots = d_n = 2$, $d_{n+1} = 1$, and $e = n - 2$, hence,

$$\begin{aligned} \frac{(1+2t)^n (1+t)}{(1+t)^3} = \frac{(1+t+t)^n}{(1+t)^2} &= \frac{\sum_{j=0}^{n} (1+t)^{n-j} t^j \binom{n}{j}}{(1+t)^2} \\ &= \sum_{j=0}^{n} (1+t)^{n-j-2} t^j \binom{n}{j} \end{aligned}$$

and $s = \sum_{j=0}^{n-2} \binom{n}{j}$, meaning there are at least $\sum_{j=0}^{n-2} \binom{n}{j}$ solutions of $P(\mathbf{y}) = \mathbf{0}$ at infinity. In addition, there is an isolated solution $x_0 = x_1 = \cdots = x_n = 0,\ \lambda = 1$ at infinity. Thus there are at most

$$2^n - s - 1 = (1+1)^n - \sum_{j=0}^{n-2} \binom{n}{j} - 1 = n$$

solutions of $P(\lambda, x_1, \ldots, x_n) = \mathbf{0}$ in $\mathbb{C}^{n+1}$. The system $Q(\lambda, x_1, \ldots, x_n) = \mathbf{0}$ is chosen to have the same nonsingular variety at infinity, and this variety stays at infinity as the homotopy progresses from $t = 0$ to $t = 1$. As a result, the infinity solutions stay infinite, the finite solution paths stay finite, and no extraneous paths exist.

This turns out to be a fairly typical situation. Even though the system $P(\mathbf{x}) = \mathbf{0}$ to be solved has isolated solutions, when viewed in projective space there may be a large number of roots at infinity and quite often high-dimensional manifolds of roots at infinity. Extraneous paths are those that are drawn to the manifolds lying at infinity. If $Q(\mathbf{x}) = \mathbf{0}$ can be chosen correctly, extraneous paths can be eliminated. To be more precise, we state the main random product homotopy result, Theorem 2.2 of [62]. Let $V_\infty(Q)$ and $V_\infty(P)$ denote the zeros "at infinity" of $Q(\mathbf{x}) = \mathbf{0}$ and $P(\mathbf{x}) = \mathbf{0}$ respectively.

Theorem 5.1. *If $V_\infty(Q)$ is nonsingular and contained in $V_\infty(P)$, then Properties 1 and 2 hold.*

Of course, Properties 1 and 2 are not enough. Without starting points, the path-tracing method cannot get started. Thus $Q(\mathbf{x}) = \mathbf{0}$ should also be chosen to be of random product forms, as in (5.6), which are trivial to solve because of their form.

This result was superseded by the result in [60]. The complex numbers e_{ij} are chosen at random in [62] to ensure Properties 1 and 2. In [60], it was proved that e_{ij} can be any fixed numbers, as long as the complex number c is chosen at random, Properties 1 and 2 still hold. In fact, the result in [60] implies that the start system $Q(\mathbf{x}) = \mathbf{0}$ in the above theorem need not be in product form. It can be any chosen polynomial system as long as its zeros in $\mathbb{C}^n$ are known or easy to obtain and its variety of roots at infinity $V_\infty(Q)$ is nonsingular and contained in $V_\infty(P)$.

Theorem 2.1 in [68] goes one step further. Even when the set $V_\infty(Q)$ of roots at infinity of $Q(\mathbf{x}) = \mathbf{0}$ has singularities, if the set is contained in $V_\infty(P)$ counting multiplicities, that is, containment in the sense of *scheme* theory of algebraic geometry, then Properties 1 and 2 still hold. More precisely, let

$I = \langle \widetilde{q}_1, \ldots, \widetilde{q}_n \rangle$ and $J = \langle \widetilde{p}_1, \ldots, \widetilde{p}_n \rangle$ be the homogeneous ideals spanned by homogenizations of q_i's and p_i's respectively. For a point p at infinity, if the *local rings* I_p and J_p satisfy

$$I_p \subset J_p$$

then Properties 1 and 2 hold. However, this hypothesis can be much more difficult to verify than whether the set is nonsingular. This limits the usefulness of this approach for practical examples.

5.2. *m*-Homogeneous structure

In [81], another interesting approach to reduce the number of extraneous paths is developed, using the concept of multi-homogeneous, or m-homogeneous for short, structures.

The complex n-space $\mathbb{C}^n$ can be naturally embedded in the projective space $\mathbb{CP}^n$. Similarly, the space $\mathbb{C}^{k_1} \times \cdots \times \mathbb{C}^{k_m}$ can be naturally embedded in $\mathbb{CP}^{k_1} \times \cdots \times \mathbb{CP}^{k_m}$. A point $(\mathbf{y}_1, \ldots, \mathbf{y}_m)$ in $\mathbb{C}^{k_1} \times \cdots \times \mathbb{C}^{k_m}$ with $\mathbf{y}_j = (y_1^{(j)}, \ldots, y_{k_j}^{(j)})$, for $j = 1, \ldots, m$, corresponds to a point $(\mathbf{z}_1, \ldots, \mathbf{z}_m)$ in $\mathbb{CP}^{k_1} \times \cdots \times \mathbb{CP}^{k_m}$ with $\mathbf{z}_j = (z_0^{(j)}, \ldots, z_{k_j}^{(j)})$ and $z_0^{(j)} = 1$ for all $j = 1, \ldots, m$. The set of such points in $\mathbb{CP}^{k_1} \times \cdots \times \mathbb{CP}^{k_m}$ is usually called the *affine space* in this setting. The points in $\mathbb{CP}^{k_1} \times \cdots \times \mathbb{CP}^{k_m}$ with at least one $z_0^{(j)} = 0$ are called the *points at infinity*.

Let f be a polynomial in the n variables $x_1, \ldots, x_n$. If we partition the variables into m groups $\mathbf{y}_1 = (x_1^{(1)}, \ldots, x_{k_1}^{(1)})$, $\mathbf{y}_2 = (x_1^{(2)}, \ldots, x_{k_2}^{(2)})$, $\ldots$, $\mathbf{y}_m = (x_1^{(m)}, \ldots, x_{k_m}^{(m)})$ with $k_1 + \cdots + k_m = n$ and let d_i be the degree of f with respect to $\mathbf{y}_i$ (more precisely, to the variables in $\mathbf{y}_i$), then we can define its *m-homogenization* as

$$\widetilde{f}(z_1, \ldots, z_m) = (z_0^{(1)})^{d_1} \times \cdots \times (z_0^{(m)})^{d_m} f(\mathbf{y}_1/z_0^{(1)}, \ldots, \mathbf{y}_m/z_0^{(m)}).$$

This polynomial is homogeneous with respect to each group of variables $z_j = (z_0^{(j)}, \ldots, z_{k_j}^{(j)})$, for $j = 1, \ldots, m$. Here $z_i^{(j)} = x_i^{(j)}$, for $i \neq 0$. Such a polynomial is said to be m-homogeneous with respect to the partition of the variables, and the tuple $(d_1, \ldots, d_m)$ is called the *m-homogeneous degree* of f. To illustrate this definition, let us look at the polynomial system

$$\begin{aligned} p_1(\mathbf{x}) &= x_1(a_{11}x_1 + \cdots + a_{1n}x_n) + b_{11}x_1 + \cdots + b_{1n}x_n + c_1 = 0 \\ &\vdots \\ p_n(\mathbf{x}) &= x_1(a_{n1}x_1 + \cdots + a_{nn}x_n) + b_{n1}x_1 + \cdots + b_{nn}x_n + c_n = 0. \end{aligned} \tag{5.7}$$

This system has total degree $d = d_1 \cdots d_n = 2^n$. Thus the "expected number of solutions" is 2^n, and the classical homotopy continuation method using the start system $Q(\mathbf{x}) = \mathbf{0}$ in (5.3) sends out 2^n paths from 2^n trivial starting points. However, the system $P(\mathbf{x}) = \mathbf{0}$ has at most $n+1$ isolated solutions. This is a deficient system, at least $2^n - n - 1$ paths will be extraneous.

Now, if we consider the partition of variables $\mathbf{y}_1 = (x_1)$, $\mathbf{y}_2 = (x_2, \ldots, x_n)$ and $\mathbf{z}_1 = (x_0^{(1)}, x_1)$, $\mathbf{z}_2 = (x_0^{(2)}, x_2, \ldots, x_n)$, then for $j = 1, \ldots, n$, the degree of

$$\begin{aligned} p_j(\mathbf{x}) &= x_1(a_{j1}x_1 + \cdots + a_{jn}x_n) + b_{j1}x_1 + \cdots + b_{jn}x_n + c_j \\ &= a_{j1}x_1^2 + x_1(a_{j2}x_2 + \cdots + a_{jn}x_n + b_{j1}) \\ &\quad + b_{j2}x_2 + \cdots + b_{jn}x_n + c_j. \end{aligned}$$

is 2 with respect to $\mathbf{y}_1$ and is 1 with respect to $\mathbf{y}_2$. Hence, its 2-homogenization is

$$\begin{aligned} \widetilde{p}_j(\mathbf{z}_1, \mathbf{z}_2) = a_{j1}x_1^2x_0^{(2)} &+ x_1x_0^{(1)}(a_{j2}x_2 + \cdots + a_{jn}x_n + b_{j1}x_0^{(2)}) \\ &+ (x_0^{(1)})^2(b_{j2}x_2 + \cdots + b_{jn}x_n + c_jx_0^{(2)}), \end{aligned}$$

which is homogeneous with respect to both $\mathbf{z}_1 = (x_0^{(1)}, x_1)$ and $\mathbf{z}_2 = (x_0^{(2)}, x_2, \ldots, x_n)$. When the system (5.7) is viewed in $\mathbb{CP}^n$ with the homogenization

$$\begin{aligned} \widetilde{p}_1(x_0, x_1, \ldots, x_n) &= x_1(a_{11}x_1 + \cdots + a_{1n}x_n) \\ &\quad + (b_{11}x_1 + \cdots + b_{1n}x_n)x_0 + c_1x_0^2 \\ &= 0, \\ &\vdots \\ \widetilde{p}_n(x_0, x_1, \ldots, x_n) &= x_1(a_{n1}x_1 + \cdots + a_{nn}x_n) \\ &\quad + (b_{n1}x_1 + \cdots + b_{nn}x_n)x_0 + c_nx_0^2 \\ &= 0, \end{aligned}$$

its total degree, or the Bézout number, is $d = d_1 \times \cdots \times d_n = 2^n$. However, when (5.7) is viewed in $\mathbb{CP}^1 \times \mathbb{CP}^{n-1} = \{(\mathbf{z}_1, \mathbf{z}_2) = ((x_0^{(1)}, x_1), (x_0^{(2)}, x_2, \ldots, x_n))$ where $\mathbf{z}_1 = (x_0^{(1)}, x_1) \in \mathbb{CP}^1$ and $\mathbf{z}_2 = (x_0^{(2)}, x_2, \ldots, x_n) \in \mathbb{CP}^{n-1}\}$ with

2-homogenization

$$
\begin{aligned}
\widetilde{p}_1(\mathbf{z}_1,\mathbf{z}_2) &= a_{11}x_1^2x_0^{(2)} + x_1x_0^{(1)}(a_{12}x_2+\cdots+a_{1n}x_n+b_{11}x_0^{(2)}) \\
&\quad + (x_0^{(1)})^2(b_{12}x_2+\cdots+b_{1n}x_n+c_1x_0^{(2)}), \\
&\vdots \\
\widetilde{p}_n(\mathbf{z}_1,\mathbf{z}_2) &= a_{n1}x_1^2x_0^{(2)} + x_1x_0^{(1)}(a_{n2}x_2+\cdots+a_{nn}x_n+b_{n1}x_0^{(2)}) \\
&\quad + (x_0^{(1)})^2(b_{n2}x_2+\cdots+b_{nn}x_n+c_nx_0^{(2)}),
\end{aligned}
\tag{5.8}
$$

the Bézout number will be different. It is defined to be the coefficient of $\alpha_1^1\alpha_2^{n-1}$ in the product $(2\alpha_1+\alpha_2)^n$, which is equal to $2n$.

In general, for an m-homogeneous system

$$
\begin{aligned}
\widetilde{p}_1(\mathbf{z}_1,\ldots,\mathbf{z}_m) &= 0, \\
&\vdots \\
\widetilde{p}_n(\mathbf{z}_1,\ldots,\mathbf{z}_m) &= 0,
\end{aligned}
\tag{5.9}
$$

in $\mathbb{CP}^{k_1}\times\cdots\times\mathbb{CP}^{k_m}$ where $\widetilde{p}_j$ has m-homogeneous degree $(d_1^{(j)},\ldots,d_m^{(j)})$ with respect to $(\mathbf{z}_1,\ldots,\mathbf{z}_m)$ for $j=1,\ldots,n$, then the *m-homogeneous Bézout number* [95] d of the system with respect to $(\mathbf{z}_1,\ldots,\mathbf{z}_m)$ is the coefficient of $\alpha_1^{k_1}\times\cdots\times\alpha_m^{k_m}$ in the product

$$
\begin{aligned}
&(d_1^{(1)}\alpha_1+\cdots+d_m^{(1)}\alpha_m)(d_1^{(2)}\alpha_1+\cdots+d_m^{(2)}\alpha_m) \\
&\cdots(d_1^{(n)}\alpha_1+\cdots+d_m^{(n)}\alpha_m).
\end{aligned}
\tag{5.10}
$$

The classical Bézout Theorem says the system (5.9) has no more than d isolated solutions, counting multiplicities, in $\mathbb{CP}^{k_1}\times\cdots\times\mathbb{CP}^{k_m}$. Applying this to our example in (5.8), the upper bound on the number of isolated solutions, in affine space and at infinity, is $2n$. When solving the original system in (5.7), we may choose the start system $Q(\mathbf{x})=\mathbf{0}$ in the homotopy

$$
H(\mathbf{x},t) = (1-t)\,c\,Q(\mathbf{x}) + tP(\mathbf{x})
$$

in random product form to respect the 2-homogeneous structure of $P(\mathbf{x})$. For instance, we may choose $Q(\mathbf{x}) = (q_1(\mathbf{x}), \ldots, q_n(\mathbf{x}))$ to be

$$\begin{aligned} q_1(\mathbf{x}) &= (x_1 + e_{11})(x_1 + e_{12})(x_2 + \cdots + x_n + e_{13}), \\ q_2(\mathbf{x}) &= (x_1 + e_{21})(x_1 + e_{22})(x_2 + e_{23}), \\ &\vdots \\ q_n(\mathbf{x}) &= (x_1 + e_{n1})(x_1 + e_{n2})(x_n + e_{n3}), \end{aligned} \tag{5.11}$$

which has the same 2-homogeneous structure as $P(\mathbf{x})$ with respect to the partition $\mathbf{y}_1 = (x_1)$ and $\mathbf{y}_2 = (x_2, \ldots, x_n)$. Namely, for each j, $q_j(\mathbf{x})$ has degree 2 with respect to $\mathbf{y}_1$ and degree one with respect to $\mathbf{y}_2$. It is easy to see by inspection that for randomly chosen complex numbers e_{ij}, $Q(\mathbf{x}) = \mathbf{0}$ has $2n$ solutions in $\mathbb{C}^n = \mathbb{C}^1 \times \mathbb{C}^{n-1}$ (thus, no solutions at infinity when viewed in $\mathbb{CP}^1 \times \mathbb{CP}^{n-1}$). Hence there are $2n$ paths starting from $2n$ starting points for this choice of the homotopy. It was shown in [81] that Properties 1 and 2 hold for all complex number c except those lying on a finite number of rays starting at the origin. Thus, all solutions of $P(\mathbf{x}) = \mathbf{0}$ are found at the end of $n + 1$ paths. The number of extraneous paths, $2n - (n + 1) = n - 1$, is far less than the number of extraneous paths, $2^n - n - 1$, by using the classical homotopy with $Q(\mathbf{x}) = \mathbf{0}$ in (5.3).

More precisely, we state the main theorem in [81].

Theorem 5.2. *Let $Q(\mathbf{x})$ be a system of polynomials chosen to have the same m-homogeneous form as $P(\mathbf{x})$ with respect to certain partition of the variables $(x_1, \ldots, x_n)$. Assume $Q(\mathbf{x}) = \mathbf{0}$ has exactly the Bézout number of nonsingular solutions with respect to this partition, and let*

$$H(\mathbf{x}, t) = (1 - t)\, c\, Q(\mathbf{x}) + t P(\mathbf{x})$$

where $t \in [0, 1]$ and $c \in \mathbb{C}^* = \mathbb{C} \setminus \{0\}$. If $c = re^{i\theta}$ for some positive $r \in \mathbb{R}$, then for all but finitely many $\theta \in [0, 2\pi]$, Properties 1 and 2 hold.

Notice that when the number of nonsingular isolated zeros of $Q(\mathbf{x})$, having the same m-homogeneous structure of $P(\mathbf{x})$ with respect to a given partition of variables $(x_1, \ldots, x_n)$, reaches the corresponding Bézout number, then no other solutions of $Q(\mathbf{x}) = \mathbf{0}$ exist in affine space.

In general, if $\mathbf{x} = (x_1, \ldots, x_n)$ is partitioned into $\mathbf{x} = (\mathbf{y}_1, \ldots, \mathbf{y}_m)$ where

$$\mathbf{y}_1 = (x_1^{(1)}, \ldots, x_{k_1}^{(1)}),\ \mathbf{y}_2 = (x_1^{(2)}, \ldots, x_{k_2}^{(2)}),\ \ldots,\ \mathbf{y}_m = (x_1^{(m)}, \ldots, x_{k_m}^{(m)})$$

with $k_1+\cdots+k_m=n$, and for polynomial system $P(\mathbf{x})=(p_1(\mathbf{x}),\ldots,p_n(\mathbf{x}))$ where $p_j(\mathbf{x})$ has degree $(d_1^{(j)},\ldots,d_m^{(j)})$ with respect to $(\mathbf{y}_1,\ldots,\mathbf{y}_m)$ for $j=1,\ldots,n$, we may choose the start system $Q(\mathbf{x})=(q_1(\mathbf{x}),\ldots,q_n(\mathbf{x}))$ where

$$q_j(\mathbf{x})=\prod_{i=1}^{m}\prod_{l=1}^{d_i^{(j)}}(c_{l1}^{(i)}x_1^{(i)}+\cdots+c_{lk_i}^{(i)}x_{k_i}^{(i)}+c_{l0}^{(i)}),\quad j=1,\ldots,n. \tag{5.12}$$

Clearly, for each j, $q_j(\mathbf{x})$ has degree $(d_1^{(j)},\ldots,d_m^{(j)})$ with respect to $(\mathbf{y}_1,\ldots,\mathbf{y}_m)$, the same degree structure of $p_j(\mathbf{x})$. Furthermore, it is not hard to see that, for generic coefficients $Q(\mathbf{x})$ has exactly m-homogeneous Bézout number, with respect to this particular partition $\mathbf{x}=(\mathbf{y}_1,\ldots,\mathbf{y}_m)$, of nonsingular isolated zeros in $\mathbb{C}^n$. They are easy to obtain. In fact, the system $Q(\mathbf{x})$ in (5.11) is constructed according to this principle. In [113], the product in (5.12) is modified along the same line to be more efficient to evaluate.

As mentioned earlier, solving system in (5.7) with the start system $Q(\mathbf{x})$ in (5.11), there are still $n-1$ extraneous paths for the homotopy. This is because, even when viewed in $\mathbb{CP}^1\times\mathbb{CP}^{n-1}$, $P(\mathbf{x})$ has zeros at infinity. One can see in (5.8) that

$$S=\{((x_0^{(1)},x_1),(x_0^{(2)},x_2,\ldots,x_n))\in\mathbb{CP}^1\times\mathbb{CP}^{n-1}\mid x_0^{(1)}=0,\,x_0^{(2)}=0\}$$

is a set of zeros of $P(\mathbf{x})$ at infinity. So, to lower the number of those extraneous paths further, we may choose the start system $Q(\mathbf{x})$ to have the same nonsingular variety of zeros at infinity S as $P(\mathbf{x})$ does, in addition to sharing the same 2-homogeneous structure of $P(\mathbf{x})$. For instance, the system $Q(\mathbf{x})=(q_1(\mathbf{x}),\ldots,q_n(\mathbf{x}))$ where

$$\begin{aligned}
q_1(\mathbf{x})&=(x_1+e_{11})(x_1+x_2+\cdots+x_n+e_{12}),\\
q_2(\mathbf{x})&=(x_1+e_{21})(x_1+x_2+e_{22}),\\
&\vdots\\
q_n(\mathbf{x})&=(x_1+e_{n1})(x_1+x_n+e_{n2})
\end{aligned}$$

shares the same 2-homogeneous structure of $P(\mathbf{x})$ with $\mathbf{y}_1=(x_1)$ and $\mathbf{y}_2=(x_2,\ldots,x_n)$, namely, for each j, $q_j(\mathbf{x})$ has degree two with respect to $\mathbf{y}_1$ and degree one with respect to $\mathbf{y}_2$. On the other hand, when viewed in $(\mathbf{z}_1,\mathbf{z}_2)\in\mathbb{CP}^1\times\mathbb{CP}^{n-1}$ with $\mathbf{z}_1=(x_0^{(1)},x_1)$ and $\mathbf{z}_2=(x_0^{(2)},x_2,\ldots,x_n)$, this system has the same nonsingular variety S of zeros at infinity as $P(\mathbf{x})$. The system $Q(\mathbf{x})=\mathbf{0}$ also has $n+1$ solutions in $\mathbb{C}^n$ for generic e_{ji}'s, and there

will be no extraneous paths. It can be shown [68, 81] that if $Q(\mathbf{x}) = \mathbf{0}$ in

$$H(\mathbf{x}, t) = (1-t)cQ(\mathbf{x}) + tP(\mathbf{x})$$

is chosen to have the same m-homogeneous form as $P(\mathbf{x})$ and the set of zeros at infinity $V_\infty(Q)$ of $Q(\mathbf{x})$ is nonsingular and contained in $V_\infty(P)$, the set of zeros at infinity of $P(\mathbf{x})$, then for $c = re^{i\theta}$ for some positive $r \in \mathbb{R}$ and for all but finitely many θ, Properties 1 and 2 hold.

Most often the zeros at infinity of an m-homogeneous polynomial system $\widetilde{P}(\mathbf{z}_1, \ldots, \mathbf{z}_m)$ in $\mathbb{CP}^{k_1} \times \cdots \times \mathbb{CP}^{k_m}$ is hard to identify. Nevertheless, the choice of $Q(\mathbf{x}) = \mathbf{0}$ in (5.12), having no zeros at infinity regardless of the structure of the zeros at infinity of $P(\mathbf{x})$, can still reduce the number of extraneous paths dramatically by simply sharing the same m-homogeneous structure of $P(\mathbf{x})$ only.

Let us look at the system

$$\begin{aligned} p_1(\mathbf{x}) &= x_1(a_{11}x_1 + \cdots + a_{1n}x_n) + b_{11}x_1 + \cdots + b_{1n}x_n + c_1 = 0, \\ &\vdots \\ p_n(\mathbf{x}) &= x_1(a_{n1}x_1 + \cdots + a_{nn}x_n) + b_{n1}x_1 + \cdots + b_{nn}x_n + c_n = 0, \end{aligned}$$

in (5.7) again. This time we partition the variables $x_1, \ldots, x_n$ into $\mathbf{y}_1 = (x_1, x_2)$ and $\mathbf{y}_2 = (x_3, \ldots, x_n)$. For this partition, the 2-homogeneous degree structure of $p_j(\mathbf{x})$ stays the same, namely, the degree of $p_j(\mathbf{x})$ is two with respect to $\mathbf{y}_1$ and is one with respect to $\mathbf{y}_2$. However, the Bézout number with respect to this partition becomes the coefficient of $\alpha_1^2\alpha_2^{n-2}$ in the product $(2\alpha_1 + \alpha_2)^n$ according to (5.10). This number is

$$\begin{pmatrix} n \\ 2 \end{pmatrix} \times 2^2 = 2n(n-1),$$

which is greater than the original Bézout number $2n$ with respect to the partition $\mathbf{y}_1 = (x_1)$ and $\mathbf{y}_2 = (x_2, \ldots, x_n)$ when $n > 2$. If the start system $Q(\mathbf{x})$ is chosen to have the same m-homogeneous structure of $P(\mathbf{x})$ with respect to this partition, then, assuming $Q(\mathbf{x})$ has no zeros at infinity, we need to follow $2n(n-1)$ paths to find all $n+1$ isolated zeros of $P(\mathbf{x})$. This represents a much bigger amount of extraneous paths.

Apparently, the m-homogeneous Bézout number is highly sensitive to the chosen partition: different ways of partitioning the variables produce different Bézout numbers. By using Theorem 5.2, we usually trace the Bézout number (with respect to the chosen partition of variables) of paths to obtain

all the isolated zeros of $P(\mathbf{x})$. In order to minimize the number of paths need to be traced and hence avoid more extraneous paths, it's critically important to find a partition which provides the lowest Bézout number possible. In [112], an algorithm for this purpose was given. By using this algorithm one can determine, for example, the partition $\mathcal{P} = \{(b), (c, d, e)\}$ which gives the lowest possible Bézout number 368 for the Cassou-Nogues system in (5.4). Consequently, we may construct a random product start system $Q(\mathbf{x})$, as in (5.12) for instance, to respect the degree structure of the Cassou-Nogues system with respect to this partition. The start system $Q(\mathbf{x})$ will have 368 isolated zeros in $\mathbb{C}^n$, and, according to Theorem 2.2, only 368 homotopy paths need to be followed to obtain all 16 isolated zeros of the system, in contrast to following 1344 paths if we choose the start system $Q(\mathbf{x})$ as in (5.3).

In the remainder of this section, we shall elaborate the algorithm given in [112] designed to search for a partition of variables which provides the lowest corresponding m-homogeneous Bézout number of a polynomial system.

First of all, we need a systematic listing of all the possible partitions of the variables $\{x_1, \ldots, x_n\}$. This can be obtained via considering the reformulated problem: how many different ways are there to partition n distinct items into m identical boxes for $m = 1, \ldots, n$? Denote those numbers by $g(n, m)$, $m = 1, \ldots, n$. Clearly, we have $g(n, n) = 1$ and $g(n, 1) = 1$. Furthermore, the recursive relation

$$g(n, m) = m \times g(n - 1, m) + g(n - 1, m - 1)$$

holds, because for each of the $g(n - 1, m)$ partitions of $n - 1$ items, we may add the nth item to any one of m boxes, plus for each of the $g(n - 1, m - 1)$ partitions of $n - 1$ items into $m - 1$ boxes we can only put the nth item in the mth box by itself. The numbers $g(n, m)$ are known as *Stirling numbers of the second kind* [93].

For a given partition $\mathbf{y}_1 = (x_1^{(1)}, \ldots, x_{k_1}^{(1)}), \ldots, \mathbf{y}_m = (x_1^{(m)}, \ldots x_{k_m}^{(m)})$ of the variables $\{x_1, \ldots, x_n\}$ of a polynomial system $P(\mathbf{x}) = (p_1(\mathbf{x}), \ldots, p_n(\mathbf{x}))$ where $k_1 + \cdots + k_m = n$ and

$$d_{ij} = \text{degree of } p_i \text{ with respect to } \mathbf{y}_j,$$

straightforward application of the definition given in (5.10) to compute the Bézout number, (namely, expanding the product and finding the appropriate coefficient), does not lead to an efficient algorithm except in the most simplest cases. A simpler approach is given below.

First of all, it's easy to see that the definition of the Bézout number given in (5.10) can be restated as: the Bézout number is the sum of all products of the form

$$d_{1\ell_1} \times d_{2\ell_2} \times \cdots \times d_{n\ell_n}$$

where among $\ell_1, \ldots, \ell_n$ each integer $j = 1, \ldots, m$ appears exactly k_j times. That is, in the *degree matrix*

$$D = \begin{bmatrix} d_{11} & \cdots & d_{1m} \\ \vdots & \ddots & \vdots \\ d_{n1} & \cdots & d_{nm} \end{bmatrix} \tag{5.13}$$

we sum degree products over all possible ways to choose each row once while choosing k_j entries from each column j. Thus, to calculate the Bézout number, we may enumerate the permissible combinations, form the corresponding degree products, and add them up. Since many of the degree products contain common factors, we may reduce the number of multiples by a method resembling the evaluation of a determinant via expansion by minors, either down the column or across the rows. The row expansion is generally more efficient, so we shall present only this alternative.

For *partition vector* $K = [k_1, \ldots, k_m]$, consider forming degree products in degree matrix D in (5.13) as follows. First, in row 1 of D, suppose we choose element d_{1j}. Then to complete the degree product we must choose one element from each of the remaining rows while only $k_j - 1$ elements from the jth column are included. So, a *minor* corresponding to d_{1j} is derived by deleting row 1 of D and decrementing k_j by 1. This *minor* has the corresponding Bézout number in its own right, with respect to the partition vector $K' = [k_1, \ldots, k_{j-1}, k_j - 1, k_{j+1}, \ldots, k_m]$. The *row expansion algorithm* for the Bézout number of degree matrix D with respect to the partition vector $K = [k_1, \ldots, k_m]$ is to compute the sum along the first row of each $d_{1j} (k_j > 0)$ times the Bézout number of the corresponding minor. The Bézout number of each minor is then computed recursively by the same row expansion procedure.

More precisely, let $b(D, \bar{K}, i)$ be the Bézout number of the degree matrix

$$D_i = \begin{bmatrix} d_{i1} & \cdots & d_{im} \\ \vdots & \ddots & \vdots \\ d_{n1} & \cdots & d_{nm} \end{bmatrix}$$

consisted of the last $n-i+1$ rows of D in (5.13), with respect to the partition vector $\bar{K} = [\bar{k}_1, \ldots, \bar{k}_m]$. Here, of course, $\bar{k}_1 + \cdots + \bar{k}_m = n-i+1$. Let $M(\bar{K}, j)$ be the partition vector derived by decrementing $\bar{k}_j$ in $\bar{K}$ by 1, namely,

$$M(\bar{K}, j) = [\bar{k}_1, \ldots, \bar{k}_{j-1}, \bar{k}_j - 1, \bar{k}_{j+1}, \ldots, \bar{k}_m].$$

With the convention $b(D, \bar{K}, n+1) := 1$ the row expansion algorithm may be written as

$$b(D, \bar{K}, i) = \sum_{\substack{j=1 \\ k_j \neq 0}}^{m} d_{ij} b(D, M(\bar{K}, j), i+1)$$

and the Bézout number of the original degree matrix D with respect to the partition vector $K = [k_1, \ldots, k_m]$ is simply $B = b(D, K, 1)$.

Note that if the degree matrix D is sparse, we may skip over computations where $d_{ij} = 0$ and avoid expanding the recursion below that branch.

Example 5.3 ([112]). . For the polynomial system $P(\mathbf{x}) = (p_1(\mathbf{x}), \ldots, p_4(\mathbf{x}))$, with $\mathbf{x} = (x_1, x_2, x_3, x_4)$, let $\mathbf{y}_1 = (x_1, x_2)$ and $\mathbf{y}_2 = (x_3, x_4)$. So, the partition vector $K = [2, 2]$. Let

$$D = \begin{bmatrix} d_{11} & d_{12} \\ d_{21} & d_{22} \\ d_{31} & d_{32} \\ d_{41} & d_{34} \end{bmatrix}$$

be the degree matrix. Then, by the row expansion algorithm, the Bézout number B of D with respect to K is,

$$\begin{aligned}
B &= d_{11} b(D, [1,2], 2) + d_{12} b(D, [2,1], 2) \\
&= d_{11}\big[d_{21} \cdot b(D, [0,2], 3) + d_{22} \cdot b(D, [1,1], 3)\big] \\
&\quad + d_{12}\big[d_{21} \cdot b(D, [1,1], 3) + d_{22} \cdot b(D, [2,0], 3)\big] \\
&= d_{11}\big[d_{21} d_{32} \cdot b(D, [0,1], 4) + d_{22}(d_{31} \cdot b(D, [0,1], 4) + d_{32} \cdot b(D, [1,0], 4))\big] \\
&\quad + d_{12}\big[d_{21}(d_{31} \cdot b(D, [0,1], 4) + d_{32} \cdot b(D, [1,0], 4)) \\
&\quad + d_{22} \cdot (d_{31} \cdot b(D, [1,0], 4)\big] \\
&= d_{11}(d_{21} d_{32} d_{42} + d_{22}(d_{31} d_{42} + d_{32} d_{41})) \\
&\quad + d_{12}(d_{21}(d_{31} d_{42} + d_{32} d_{41}) + d_{22} d_{31} d_{41}).
\end{aligned}$$

Example 5.4 ([112]). Consider the system

$$\begin{aligned} x_1^2 + x_2 + 1 &= 0 \\ x_1 x_3 + x_2 + 2 &= 0 \\ x_2 x_3 + x_3 + 3 &= 0. \end{aligned}$$

There are five ways to partition the variables $\{x_1, x_2, x_3\}$. We list the degree matrices and Bézout numbers calculated by the row expansion algorithm for all five partition schemes as follows:

1) $\{x_1, x_2, x_3\}$

$$K = [3], \qquad D = \begin{bmatrix} 2 \\ 2 \\ 2 \end{bmatrix}, \qquad \text{Bézout number} = 8$$

2) $\{x_1, x_2\}\,\{x_3\}$

$$K = [2, 1], \qquad D = \begin{bmatrix} 2 & 0 \\ 1 & 1 \\ 1 & 1 \end{bmatrix} \qquad \text{Bézout number} = 4.$$

3) $\{x_1, x_3\}\,\{x_2\}$

$$K = [2, 1], \qquad D = \begin{bmatrix} 2 & 1 \\ 2 & 1 \\ 1 & 1 \end{bmatrix} \qquad \text{Bézout number} = 8$$

4) $\{x_1\}\,\{x_2, x_3\}$

$$K = [1, 2], \qquad D = \begin{bmatrix} 2 & 1 \\ 1 & 1 \\ 0 & 2 \end{bmatrix} \qquad \text{Bézout number} = 6$$

5) $\{x_1\}, \{x_2\}, \{x_3\}$

$$K = [1, 1, 1], \qquad D = \begin{bmatrix} 2 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 1 \end{bmatrix} \qquad \text{Bézout number} = 5$$

Thus, the grouping $(\{x_1, x_2\}, \{x_3\})$ has the lowest Bézout number$(= 4)$ and would lead to the most efficient homotopy continuation.

When exhaustively searching for the partitioning of the variables which provides the minimal Bézout number of the system, there are several ways to speed up the process. For instance, as we sequentially test partitioning in search for minimal Bézout numbers, we can use the smallest one found so far to cut short unfavorable partitioning. Since the degrees are all nonnegative, the Bézout number is a sum of nonnegative degree products. If at any time the running subtotal exceeds the current minimal Bézout number, the calculation can be aborted and testing of the next partitioning can proceed. This can save a substantial amount of computation during an exhaustive search. See [112] for more details.

While the number of partitioning to be tested grows rapidly with the number of variables, the exhaustive search can be easily parallelized by subdividing the tree of partitioning and distributing these branches to multiple processors for examination. Thus, continuing advances in both raw computer speed and in parallel machines will make progressively larger problems feasible.

In [110], a *generalized Bézout number*, or GB, is developed in the *GBQ-algorithm*, in which the partition of variables is permitted to vary among the $p_j(\mathbf{x})$'s. Even lower Bézout number may be achieved when a proper partition structure of the variables for each individual polynomial $p_j(\mathbf{x})$, $j = 1, \ldots, n$ is chosen. This strategy can take a great advantage on certain sparse systems where an appropriate partition of variables is evident.

5.3. Cheater's homotopy

To organize our discussion in this section, we will at times use a notation that makes the coefficients and variables in the polynomial system $P(\mathbf{x}) = \mathbf{0}$ explicit. Thus when the dependence on coefficients is important, we will consider the system $P(\mathbf{c}, \mathbf{x}) = \mathbf{0}$ of n polynomial equations in n unknowns, where $\mathbf{c} = (c_1, \ldots, c_M)$ are coefficients and $\mathbf{x} = (x_1, \ldots, x_n)$ are unknowns.

A method called the *cheater's homotopy* [63, 64] has been developed to deal with the problem when the system $P(\mathbf{c}, \mathbf{x}) = \mathbf{0}$ is asked to be solved for several different set of the coefficients $\mathbf{c}$ (a similar procedure can be found in [82]).

The idea of the method is to theoretically establish Properties 1 and 2 by deforming a sufficiently generic system (the precise sense will be given later) and then to "cheat" on Property 0 by using a preprocessing step. The amount of computation of preprocessing step may be large, but is amortized among the several solving characteristics of the problem.

We begin with an example. With $\mathbf{x} = (x_1, x_2)$, let $P(\mathbf{x})$ be the system

$$\begin{aligned} p_1(\mathbf{x}) &= x_1^3 x_2^2 + c_1 x_1^3 x_2 + x_2^2 + c_2 x_1 + c_3 = 0, \\ p_2(\mathbf{x}) &= c_4 x_1^4 x_2^2 - x_1^2 x_2 + x_2 + c_5 = 0. \end{aligned} \tag{5.14}$$

This is a system of two polynomial equations in two unknowns x_1 and x_2. We want to solve this system of equations several times for various specific choices of $\mathbf{c} = (c_1, \ldots, c_5)$.

It turns out that for any choice of coefficients $\mathbf{c}$, system (5.14) has no more than 10 isolated solutions. More precisely, there is an open dense subset S of $\mathbb{C}^5$ such that for all $\mathbf{c}$ belonging to S, (5.14) has exactly 10 solutions. Moreover, 10 is an upper bound for the number of isolated solutions for all $\mathbf{c}$ in $\mathbb{C}^5$. The total degree of the system is $6 \times 5 = 30$, meaning that if we had taken a generic system of two polynomials in two variables of degree 5 and 6, we expect there would be 30 solutions. Thus (5.14), with any choice of $\mathbf{c} = (c_1, \ldots, c_5)$, is a deficient system. The classical homotopy using the start system $Q(\mathbf{x}) = \mathbf{0}$ in (5.3) produces $d = 30$ paths, beginning at 30 trivial starting points. Thus there are (at least) 20 extraneous paths.

The cheater's homotopy continuation approach begins by solving (5.14) with *randomly-chosen* complex coefficients $\bar{\mathbf{c}} = (\bar{c}_1, \ldots, \bar{c}_5)$; let X^* be the set of 10 solutions. No work is saved there, since 30 paths need to be followed, and 20 paths are wasted. However, the 10 elements of the set X^* are the seeds for the remainder of the process. In the future, for each choice of coefficients $\mathbf{c} = (c_1, \ldots, c_5)$ for which the system (5.14) needs to be solved, we use the homotopy continuation method to follow a linear homotopy from the system with coefficient $\bar{\mathbf{c}}$ to the system with coefficient $\mathbf{c}$. We follow the 10 paths beginning at the 10 elements of X^*. Thus Property 0, that of having trivial-available starting points, is satisfied. The fact that Properties 1 and 2 are also satisfied is the content of Theorem 5.5 below. Thus for each fixed $\mathbf{c}$, all 10 (or fewer) isolated solutions of (5.14) lie at the end of 10 smooth homotopy paths beginning at the seeds in X^*. After the initial preprocessing step of finding the "seeds", the complexity of all further solving of (5.14) is proportional to the number of solutions 10, rather than the total degree 30.

Furthermore, this method requires no a priori analysis of the system. The first preprocessing step of finding the seeds establishes a sharp theoretical upper bound on the number of isolated solutions as a by-product of the computation; further solving of the system uses the optimal number of paths to be traced.

We earlier characterized a successful homotopy continuation method as having three properties: triviality, smoothness, and accessibility. Given an

arbitrary system of polynomial equations, such as (5.14), it is not too hard (through generic perturbations) to find a family of systems with the last two properties. The problem is that one member of the family must be trivial to solve, or the path-following cannot get started. The idea of the cheater's homotopy is simply to "cheat" on this part of the problem, and run a preprocessing step (the computation of the seeds X^*) which gives us the triviality property in a roundabout way. Thus the name, the "cheater's homotopy".

A statement of the theoretical result we need follows. Let

$$\begin{aligned} p_1(c_1,\ldots,c_M,x_1,\ldots,x_n) &= 0,\\ &\vdots\\ p_n(c_1,\ldots,c_M,x_1,\ldots,x_n) &= 0, \end{aligned} \tag{5.15}$$

be a system of polynomial equations in the variables $c_1,\ldots,c_M,x_1,\ldots,x_n$. Write $P(\mathbf{c},\mathbf{x}) = (p_1(\mathbf{c},\mathbf{x}),\ldots,p_n(\mathbf{c},\mathbf{x}))$. For each choice of $\mathbf{c} = (c_1,\ldots,c_M)$ in $\mathbb{C}^M$, this is a system of polynomial equations in the variables $x_1,\ldots,x_n$. Let d be the total degree of the system for a generic choice of $\mathbf{c}$.

Theorem 5.5. *Let c belong to $\mathbb{C}^M$. There exists an open dense full-measure subset U of $\mathbb{C}^{n+M}$ such that for $(b_1^*,\ldots,b_n^*,c_1^*,\ldots,c_M^*) \in U$, the following holds:*

(a) *The set X^* of solutions $\mathbf{x} = (x_1,\ldots,x_n)$ of*

$$\begin{aligned} q_1(x_1,\ldots,x_n) &= p_1(c_1^*,\ldots,c_M^*,x_1,\ldots,x_n) + b_1^* = 0\\ &\vdots\\ q_n(x_1,\ldots,x_n) &= p_n(c_1^*,\ldots,c_M^*,x_1,\ldots,x_n) + b_n^* = 0 \end{aligned}$$

consists of d_0 isolated points, for some $d_0 \le d$.

(b) *The smoothness and accessibility properties hold for the homotopy*

$$\begin{aligned} H(\mathbf{x},t) = P((1-t)c_1^* + tc_1,\ldots,(1-t)c_M^* + tc_M,x_1,\ldots,x_n)\\ + (1-t)b^* \end{aligned} \tag{5.16}$$

where $\mathbf{b}^* = (b_1^*,\ldots,b_n^*)$. It follows that every solution of $P(\mathbf{c},\mathbf{x}) = \mathbf{0}$ is reached by a path beginning at a point of X^*.

A proof of Theorem 5.5 can be found in [64]. The theorem is used as part of the following procedure. Let $P(\mathbf{c}, \mathbf{x}) = \mathbf{0}$ be as in (5.15) denote the system to be solved for various values of the coefficients $\mathbf{c}$.

Cheater's Homotopy Procedure:

(1) Choose complex numbers $(b_1^*, \ldots, b_n^*, c_1^*, \ldots, c_M^*)$ at random, and use the classical homotopy continuation method to solve $Q(\mathbf{x}) = \mathbf{0}$ in (5.16). Let d_0 denote the number of solutions found (This number is bounded above by the total degree d). Let X^* denote the set of d_0 solutions.

(2) For each new choice of coefficients $\mathbf{c} = (c_1, \ldots, c_M)$, follow the d_0 paths defined by $H(\mathbf{x}, t) = \mathbf{0}$ in (5.16), beginning at the points in X^*, to find all solutions of $P(\mathbf{c}, \mathbf{x}) = \mathbf{0}$.

In Step (1) above, for random complex numbers $(c_1^*, \ldots, c_M^*)$, using classical homotopy continuation methods to solve $Q(\mathbf{x}) = \mathbf{0}$ in (5.16) may itself be computationally expensive. It is desirable that those numbers do not have to be random. For illustration, consider the linear system

$$\begin{aligned} c_{11}x_1 + \cdots + c_{1n}x_n &= b_1, \\ &\vdots \\ c_{n1}x_1 + \cdots + c_{nn}x_n &= b_n, \end{aligned} \tag{5.17}$$

which may be considered as a polynomial system with degree one of each equation. For randomly chosen c_{ij}'s, (5.17) has a unique solution which is not available right away. However, if we choose $c_{ij} = \delta_{ij}$(the Kronecker delta; $= 1$ if $i = j$, $= 0$ if $i \neq j$), the solution is immediate.

For this purpose, an alternative is suggested in [69]. When a system $P(c, \mathbf{x}) = \mathbf{0}$ with a particular parameter c^0 is solved, this c^0 may be assigned specifically instead of being chosen randomly, then for any parameter $c \in \mathbb{C}^M$ consider the nonlinear homotopy

$$H(a, \mathbf{x}, t) = P((1 - [t - t(1-t)a])\mathbf{c}^0 + (t - t(1-t)a)\mathbf{c}, \mathbf{x}) = \mathbf{0}. \tag{5.18}$$

It was shown in [69] that for randomly chosen complex number a the solution paths of $H(a, \mathbf{x}, t) = \mathbf{0}$ in (5.18), emanating from the solutions of $P(\mathbf{c}^0, \mathbf{x}) = \mathbf{0}$ will reach the isolated solutions of $P(\mathbf{c}, \mathbf{x}) = \mathbf{0}$ under the natural assumption that for generic $\mathbf{c}$, $P(\mathbf{c}, \mathbf{x})$ has the same number of isolated zeros in $\mathbb{C}^n$.

The most important advantage of the homotopy in (5.18) is that the parameter $\mathbf{c}^0$ of the start system $P(\mathbf{c}^0, \mathbf{x}) = \mathbf{0}$ need not be chosen at random as long as it is chosen for which $P(\mathbf{c}^0, \mathbf{x}) = \mathbf{0}$ has the same number of

solutions as $P(\mathbf{c}, \mathbf{x}) = \mathbf{0}$ for generic $\mathbf{c}$. Therefore, in some situations, when the solutions of $P(\mathbf{c}, \mathbf{x}) = \mathbf{0}$ are easily available for certain particular parameter $\mathbf{c}^0$, the system $P(\mathbf{c}^0, \mathbf{x}) = \mathbf{0}$ may be used as the start system in (5.18) and the extra effort of solving $P(\mathbf{c}, \mathbf{x}) = \mathbf{0}$ for a randomly chosen $\mathbf{c}$ would be saved.

To finish, we give a more non-trivial example of the use of the procedure described in this section.

Consider the indirect position problem for revolute-joint kinematic manipulators. Each joint represents a one-dimensional choice of parameters, namely the angular position of the joint. If all angular positions are known, then of course the position and orientation of the end of the manipulator (the hand) are determined. The indirect position problem is the inverse problem: given the desired position and orientation of the hand, find a set of angular parameters for the (controllable) joints which will place the hand in the desired state.

The indirect position problem for six joints is reduced to a system of eight nonlinear equations in eight unknowns in [108]. The coefficients of the equations depend on the desired position and orientation, and a solution of the system (an eight-vector) represents the sines and cosines of the angular parameters. Whenever the manipulator's position is changed, the system needs to be resolved with new coefficients. The equations are too long to repeat here (see the appendix of [108]); suffice to say that it is a system of eight degree-two polynomial equations in eight unknowns which is quite deficient. The total degree of the system is $2^8 = 256$, but there are at most 32 isolated solutions.

The nonlinear homotopy (5.18) requires only 32 paths to solve the system with different set of parameters[67, 69]. The system contains 26 coefficients, and a specific set of coefficients is chosen for which the system has 32 solutions. For subsequent solving of the system, for any choice of the coefficients $c_1, \ldots, c_{26}$, all solutions can be found at the end of exactly 32 paths, by using the homotopy in (5.18) with randomly chosen complex number a.

6. Theorem of Bernshteín, mixed volume, and mixed cells

Almost all the homotopies we discussed above are in the form of

$$H(\mathbf{x}, t) = (1 - t)cQ(\mathbf{x}) + tP(\mathbf{x})$$

which is linear in t. Homotopies that are nonlinear in t was originally suggested by S. T. Yau [116]. In the middle of 90's, a major computational

advance has emerged in solving polynomial systems by the nonlinear homotopy method. The new method takes a great advantage of Bernsteín's theorem [9] which provides a much tighter bound, in general, than the classical Bézout number and its variants for the number of isolated zeros of a polynomial system in the algebraic tori $(\mathbb{C}^*)^n$ where $\mathbb{C}^* = \mathbb{C} \setminus \{0\}$. Based on this root count, a nonlinear homotopy, commonly known as the *polyhedral homotopy*, was introduced in [42] to find all isolated zeros of polynomial systems.

6.1. Theorem of Bernshteín

We begin with an example [42]. With $\mathbf{x} = (x_1, x_2)$, let $P(\mathbf{x})$ be the system

$$\begin{aligned} p_1(\mathbf{x}) &= c_{11}x_1x_2 + c_{12}x_1 + c_{13}x_2 + c_{14} = 0, \\ p_2(\mathbf{x}) &= c_{21}x_1x_2^2 + c_{22}x_1^2x_2 + c_{23} = 0. \end{aligned} \tag{6.1}$$

Here, $c_{ij} \in \mathbb{C}^* = \mathbb{C} \setminus \{0\}$. The formal expressions for the monomials $\{x_1x_2,\ x_1,\ x_2,\ 1\}$ in p_1 are $x_1x_2 = x_1^1x_2^1$, $x_1 = x_1^1x_2^0$, $x_2 = x_1^0x_2^1$ and $1 = x_1^0x_2^0$. The set of their exponents

$$S_1 = \{a = (0,0),\ b = (1,0),\ c = (1,1),\ d = (0,1)\}$$

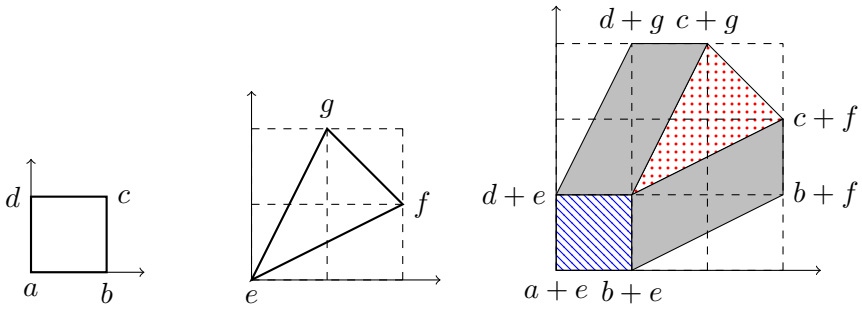
is called the *support* of p_1, and its convex hull $Q_1 = \mathrm{conv}(S_1)$ is called the *Newton polytope* of p_1. Similarly, p_2 has support $S_2 = \{e = (0,0),\ f = (2,1),\ g = (1,2)\}$ and Newton polytope $Q_2 = \mathrm{conv}(S_2)$. With the notation $\mathbf{x}^{\mathbf{q}} = x_1^{q_1}x_2^{q_2}$ for $\mathbf{q} = (q_1, q_2)$, we may rewrite (6.1) as

$$p_1(\mathbf{x}) = \sum_{\mathbf{q}\in S_1} c_{1,\mathbf{q}}\mathbf{x}^{\mathbf{q}} \quad \text{and} \quad p_2(\mathbf{x}) = \sum_{\mathbf{q}\in S_2} c_{2,\mathbf{q}}\mathbf{x}^{\mathbf{q}}.$$

For polytopes $R_1, \ldots, R_k$ in $\mathbb{R}^n$, their *Minkowski sum* [76] $R_1 + \cdots + R_k$ is defined by

$$R_1 + \cdots + R_k = \{\mathbf{r}_1 + \cdots + \mathbf{r}_k \mid \mathbf{r}_j \in R_j,\ j = 1, \ldots, k\}.$$

(polytopes Q_1, Q_2 and $Q_1 + Q_2$ for the system in (6.1) are shown in Figure 12). Now, consider the area of the convex polygon $\lambda_1Q_1 + \lambda_2Q_2$ with non-negative variables λ_1 and λ_2 for the system (6.1). First of all, the area



(a) Q_1 (b) Q_2 (c) The Minkowski sum $Q_1 + Q_2$

Figure 12. The Newton polytopes of p_1 and p_2 along with their Minkowski sum

of a triangle on the plane with vertices $\mathbf{u}$, $\mathbf{v}$ and $\mathbf{w}$ is known to be

$$\frac{1}{2}\left|\det\begin{bmatrix}\mathbf{u}-\mathbf{v}\\ \mathbf{w}-\mathbf{v}\end{bmatrix}\right|. \tag{6.2}$$

To compute the area $f(\lambda_1, \lambda_2)$ of $\lambda_1 Q_1 + \lambda_2 Q_2$, we may partition the polytope $\lambda_1 Q_1 + \lambda_2 Q_2$ into a collection of mutually disjoint triangles $A_1, A_2, \ldots, A_l$. If we choose those triangles in which none of their vertices is an interior point of the polytope $\lambda_1 Q_1 + \lambda_2 Q_2$, then all their vertices take the form $\lambda_1 \mathbf{r}_1 + \lambda_2 \mathbf{r}_2$ for certain $\mathbf{r}_1 \in Q_1$ and $\mathbf{r}_2 \in Q_2$. From (6.2), the area of each A_i is a second degree homogeneous polynomial in λ_1 and λ_2, and therefore, $f(\lambda_1, \lambda_2)$, as a sum of the areas of $A_1, \ldots, A_l$, is also a second degree homogeneous polynomial in λ_1 and λ_2. Writing

$$f(\lambda_1, \lambda_2) = a_1\lambda_1^2 + a_2\lambda_2^2 + a_{12}\lambda_1\lambda_2,$$

the coefficient a_{12} of $\lambda_1\lambda_2$ in f is called the *mixed volume* of the polytopes Q_1 and Q_2, denoted by $\mathcal{M}(Q_1, Q_2)$. Clearly,

$$\begin{aligned} a_{12} &= f(1,1) - f(1,0) - f(0,1) \\ &= \text{area of } (Q_1 + Q_2) - \text{ area of } (Q_1) - \text{ area of } (Q_2). \end{aligned}$$

The areas of $Q_1 + Q_2$, Q_1 and Q_2, as displayed in Figure 12, are 6.5, 1 and 3.5 respectively. Therefore, the mixed volume of the polytopes Q_1 and

Q_2 is

$$\mathcal{M}(Q_1, Q_2) = a_{12} = 6.5 - 1 - 1.5 = 4.$$

On the other hand, viewed in $\mathbb{CP}^2$, the system (6.1) has two zeros $(x_0, x_1, x_2) = (0, 0, 1)$ and $(0, 1, 0)$ at infinity; hence, it can have at most 4 isolated zeros in $(\mathbb{C}^*)^2$. This is the content of the **Bernshteín theorem**: The number of isolated zeros of (6.1) in $(\mathbb{C}^*)^2$, counting multiplicities, is bounded above by the mixed volume of its Newton polytopes. Furthermore, when c_{ij}'s in (6.1) are chosen generically, then these two numbers coincide.

To state the Bernshteín Theorem in general form, let the given polynomial system be $P(\mathbf{x}) = (p_1(\mathbf{x}), \ldots, p_n(\mathbf{x})) \in \mathbb{C}[\mathbf{x}]$, where $\mathbf{x} = (x_1, \ldots, x_n)$. With $\mathbf{a} = (a_1, \ldots, a_n)$ and $\mathbf{x}^{\mathbf{a}} = x_1^{a_1} \cdots x_n^{a_n}$, write

$$\begin{aligned} p_1(\mathbf{x}) &= \sum_{\mathbf{a} \in S_1} c^*_{1,\mathbf{a}} \mathbf{x}^{\mathbf{a}}, \\ &\vdots \\ p_n(\mathbf{x}) &= \sum_{\mathbf{a} \in S_n} c^*_{n,\mathbf{a}} \mathbf{x}^{\mathbf{a}}, \end{aligned} \tag{6.3}$$

where $S_1, \ldots, S_n$ are fixed subsets of $\mathbb{N}_0^n$ with cardinals $k_j = \#S_j$, and $c^*_{j,\mathbf{a}} \in \mathbb{C}^*$ for $\mathbf{a} \in S_j$, $j = 1, \ldots, n$. As before, S_j is the *support* of $p_j(\mathbf{x})$, and $S = (S_1, \ldots, S_n)$ is the *support* of $P(\mathbf{x})$. The convex hull $Q_j = \text{conv}(S_j)$ in $\mathbb{R}^n$ is the *Newton polytope* of p_j.

For nonnegative variables $\lambda_1, \ldots, \lambda_n$, let $\lambda_1 Q_1 + \cdots + \lambda_n Q_n$ be the *Minkowski sum* of $\lambda_1 Q_1, \ldots, \lambda_n Q_n$, that is,

$$\lambda_1 Q_1 + \cdots + \lambda_n Q_n = \{\lambda_1 r_1 + \cdots + \lambda_n r_n \,|\, r_j \in Q_j, j = 1, 2, \ldots, n\}.$$

Following similar reasonings for calculating $\text{Vol}_2(\lambda_1 Q_1 + \lambda_2 Q_2)$, the area of $\lambda_1 Q_1 + \lambda_2 Q_2$ of the system in (6.1), it can be shown that the n-dimensional volume, denoted by Vol_n, of the polytope $\lambda_1 Q_1 + \cdots + \lambda_n Q_n$ is a homogeneous polynomial of degree n in $\lambda_1, \ldots, \lambda_n$. The coefficient of the monomial $\lambda_1 \times \cdots \times \lambda_n$ in this homogeneous polynomial is called the *mixed volume* of the polytopes $Q_1, \ldots, Q_n$, denoted by $\mathcal{M}(Q_1, \ldots, Q_n)$, or the mixed volume of the supports $S_1, \ldots, S_n$ denoted by $\mathcal{M}(S_1, \ldots, S_n)$. When no ambiguities exist, it is called the mixed volume of $P(\mathbf{x})$ at times.

We now embed the system (6.3) in the systems $P(\mathbf{c},\mathbf{x}) = (p_1(\mathbf{c},\mathbf{x}),\ldots,p_n(\mathbf{c},\mathbf{x}))$ where

$$\begin{aligned} p_1(\mathbf{c},\mathbf{x}) &= \sum_{\mathbf{a}\in S_1} c_{1,\mathbf{a}}\mathbf{x}^{\mathbf{a}}, \\ &\vdots \\ p_n(\mathbf{c},\mathbf{x}) &= \sum_{\mathbf{a}\in S_n} c_{n,\mathbf{a}}\mathbf{x}^{\mathbf{a}}, \end{aligned} \tag{6.4}$$

and the coefficients $\mathbf{c} = (c_{j,\mathbf{a}})$ with $\mathbf{a} \in S_j$ for $j = 1,\ldots,n$ are taken to be a set of $m := k_1 + \cdots + k_n$ variables. That is, we regard $P(\mathbf{x}) = P(\mathbf{c}^*,\mathbf{x})$ for a set of specified values of coefficients $\mathbf{c}^* = (c^*_{j,\mathbf{a}})$ in (6.4).

In what follows, the total number of isolated zeros, counting multiplicities, of a polynomial system will be referred to as the *root count* of the system.

Lemma 6.1 ([42]). *For polynomial systems $P(\mathbf{c},\mathbf{x})$ in (6.4), there exists an open and dense set $U \subseteq \mathbb{C}^m$ such that for those coefficients $\mathbf{c}^* = (\mathbf{c}^*_{j,\mathbf{a}}) \in U$, the root count in $(\mathbb{C}^*)^n$ of the corresponding polynomial systems in (6.4) is a fixed number. Moreover, the root count in $(\mathbb{C}^*)^n$ the polynomial system $P(\mathbf{c},\mathbf{x})$ for any choice of $\mathbf{c}$ is bounded above by this number.*

Since the set U in the above lemma is open and dense (and hence has full measure), with probability one, the corresponding polynomial system for randomly chosen coefficients $\mathbf{c}^* = (c^*_{j,\mathbf{a}}) \in \mathbb{C}^m$ will have the same root count in $(\mathbb{C}^*)^n$. Such polynomial systems are said to be *in general position*. That is, a polynomial system $P(\mathbf{c},\mathbf{x})$ with the specific choice of coefficients $\mathbf{c} \in \mathbb{C}^m$ is in general position if its root count in $(\mathbb{C}^*)^n$ equals the upper bound provide by Lemma 6.1.

Remark 6.2. It is worth noting, however, the property of being in general position has a much stronger characterization: For the family $P(\mathbf{c},\mathbf{x})$ of polynomial systems given in (6.4) which is parametrized by the coefficients $\mathbf{c} = (c_{j,\mathbf{a}})$, there exists a polynomial $G(\mathbf{c})$ such that $P(\mathbf{c},\mathbf{x})$ is in general position whenever $G(\mathbf{c}) \neq 0$.

Example 6.3. A simple example that illustrates the assertions above is the following 2×2 linear systems:

$$\begin{aligned} c_{11}x_1 + c_{12}x_2 &= b_1, \\ c_{21}x_1 + c_{22}x_2 &= b_2. \end{aligned} \tag{6.5}$$

Here, $\mathbf{c} = (c_{11}, c_{12}, c_{21}, c_{22}, -b_1, -b_2)$. Let

$$G(\mathbf{c}) = \det \begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{pmatrix} \times \det \begin{pmatrix} c_{11} & b_1 \\ c_{21} & b_2 \end{pmatrix} \times \det \begin{pmatrix} b_1 & c_{12} \\ b_2 & c_{22} \end{pmatrix}.$$

Then, when the coefficients $\mathbf{c}^* = (c_{11}^*, c_{12}^*, c_{21}^*, c_{22}^*, -b_1^*, -b_2^*)$ satisfies $G(\mathbf{c}^*) = 0$, its corresponding linear system (6.5) has no isolated solution in $(\mathbb{C}^*)^2$; otherwise, the system has a unique solution in $(\mathbb{C}^*)^2$. Note that those $\mathbf{c} \in \mathbb{C}^6$ where $G(\mathbf{c}) \neq 0$ forms an open dense set in $\mathbb{C}^6$.

For $P(\mathbf{c}, \mathbf{x})$ in general position with support $(S_1, \ldots, S_n)$, let $L(S_1, \ldots, S_n)$ be the fixed number of its isolated zeros in $(\mathbb{C}^*)^n$. This number satisfies the following properties:

1: (*Symmetric*) $L(S_1, \ldots, S_n)$ remains invariant when S_i and S_j, $i \neq j$, exchange their positions along with their corresponding polynomials p_i and p_j.

2: (*Shift invariant*) $L(S_1, \ldots, \mathbf{a} + S_j, \ldots, S_n) = L(S_1, \ldots, S_n)$ for $\mathbf{a} \in \mathbb{N}_0^n$.
Replacing $p_j(c, \mathbf{x})$ in the system in (6.4) by $\mathbf{x}^{\mathbf{a}} p_j(c, \mathbf{x})$ results in a new system with support $(S_1, \ldots, \mathbf{a} + S_j, \ldots, S_n)$. Clearly, the number of its isolated zeros in $(\mathbb{C}^*)^n$ stays the same.

3: (*Multi-linear*) $L(S_1, \ldots, S_j + \bar{S}_j, \ldots, S_n) = L(S_1, \ldots, S_j, \ldots, S_n) + L(S_1, \ldots, \bar{S}_j, \ldots, S_n)$ for $\bar{S}_j \subset \mathbb{N}^n$.
Let $\bar{P}(c, \mathbf{x}) = (p_1(c, \mathbf{x}), \ldots, \bar{p}_j(c, \mathbf{x}), \ldots, p_j(c, \mathbf{x}))$ be a system in general position with support $(S_1, \ldots, \bar{S}_j, \ldots, S_n)$. Then replacing $p_j(c, \mathbf{x})$ in the system $P(c, \mathbf{x})$ by $p_j(c, \mathbf{x})\bar{p}_j(c, \mathbf{x})$ yields a system with support $(S_1, \ldots, S_j + \bar{S}_j, \ldots, S_n)$. It is clear that the number of isolated zeros of the resulting system in $(\mathbb{C}^*)^n$ is the sum of the numbers of those isolated zeros of $P(c, \mathbf{x})$ and $\bar{P}(c, \mathbf{x})$ in $(\mathbb{C}^*)^n$.

4: (*Automorphism invariant*) $L(S_1, \ldots, S_n) = L(US_1, \ldots, US_n)$ where U is an $n \times n$ integer matrix with $\det U = \pm 1$ and $US_j = \{U\mathbf{a} \mid \mathbf{a} \in S_j\}$ for $j = 1, \ldots, n$.
Note that in writing $\mathbf{x}^{\mathbf{a}} = x_1^{a_1} \cdots x_n^{a_n}$ we regard the vector $\mathbf{a} = (a_1, \ldots, a_n)$ as a column vector. Let U_j be the j-th column of $U = (u_{ij})$ and $\mathbf{x} = \mathbf{y}^U := (\mathbf{y}^{U_1}, \ldots, \mathbf{y}^{U_n})$, i.e.,

$$x_j = \mathbf{y}^{U_j} = y_1^{u_{1j}} \cdots y_n^{u_{nj}}, \quad j = 1, \ldots, n.$$

This coordinate transformation yields

$$\begin{aligned}\mathbf{x}^{\mathbf{a}} &= x_1^{a_1}\cdots x_n^{a_n}\\ &= (y^{U_1})^{a_1}\cdots(y^{U_n})^{a_n}\\ &= y_1^{u_{11}a_1+\cdots+u_{1n}a_n}\cdots y_n^{u_{n1}a_1+\cdots+u_{nn}a_n}\\ &= \mathbf{y}^{U\mathbf{a}},\end{aligned}\tag{6.6}$$

and transforms the system $P(\mathbf{x})$ with support $(S_1,\ldots,S_n)$ to $Q(\mathbf{y}) = P(\mathbf{y}^U)$ with support $(US_1,\ldots,US_n)$. For a given isolated zeros $\mathbf{y}_0$ of $Q(\mathbf{y})$ in $(\mathbb{C}^*)^n$, $\mathbf{x}_0 = \mathbf{y}_0^U$ is clearly an isolated zero of $P(\mathbf{x})$ in $(\mathbb{C}^*)^n$. Furthermore, since $\det U = \pm 1$, $V := U^{-1}$ is also an integer matrix, and

$$\mathbf{x}^V = (\mathbf{y}^U)^V = \mathbf{y}^{(UV)} = \mathbf{y}.$$

Therefore, for an isolated zero $\mathbf{x}_0$ of $P(\mathbf{x})$ in $(\mathbb{C}^*)^n$, $\mathbf{y}_0 = \mathbf{x}_0^V$ is an isolated zero of $Q(\mathbf{y})$ in $(\mathbb{C}^*)^n$. This one to one correspondence between isolated zeros of $Q(\mathbf{y})$ and $P(\mathbf{x})$ in $(\mathbb{C}^*)^n$ gives $L(S_1,\ldots,S_n) = L(US_1, \ldots, US_n)$.

Functions that take n finite subsets $S_1,\ldots,S_n$ of $\mathbb{N}_0^n$ and return a real number satisfying all the above properties are rarely available. The mixed volume $\mathcal{M}(S_1,\ldots,S_n)$, emerged in the early 20th century, happens to be one of them:

1: (*Symmetry*) This property is obvious for $\mathcal{M}(S_1,\ldots,S_n)$ by its definition.

2: (*Shift invariant*) For $\mathbf{a}\in\mathbb{N}_0^n$ and $Q_k = \mathrm{conv}(S_k)$, $k = 1,\ldots,n$,

$$\begin{aligned}&\mathrm{Vol}_n(l_1Q_1+\cdots+l_j(\mathbf{a}+Q_j)+\cdots+l_nQ_n)\\ =&\,\mathrm{Vol}_n(\ell_j\mathbf{a}+\ell_1Q_1+\cdots+\ell_jQ_j+\cdots+\ell_nQ_n)\\ =&\,\mathrm{Vol}_n(\ell_1Q_1+\cdots+\ell_nQ_n).\end{aligned}$$

Hence, $\mathcal{M}(S_1,\ldots,\mathbf{a}+S_j,\ldots S_n) = \mathcal{M}(S_1,\ldots,S_n)$.

3: (*Multi-linear*) We shall only prove this property for the first component of $\mathcal{M}(S_1,\ldots,S_n)$, namely, for $\bar{S}_1\subset\mathbb{N}_0^n$,

$$\mathcal{M}(S_1+\bar{S}_1,S_2,\ldots,S_n) = \mathcal{M}(S_1,\ldots,S_n)+\mathcal{M}(\bar{S}_1,\ldots S_n).$$

For positive α, β, $\ell_1, \ldots, \ell_n$ and $\bar{Q}_1 = \mathrm{conv}(\bar{S}_1)$,

$$\begin{aligned} &\mathrm{Vol}_n(\ell_1(\alpha Q_1 + \beta \bar{Q}_1) + \ell_2 Q_2 + \cdots + \ell_n Q_n) \\ &= \sum_{j_1+\cdots+j_n=n} a(\alpha, \beta, j_1, \ldots, j_n) \ell_1^{j_1} \cdots \ell_n^{j_n} \end{aligned} \tag{6.7}$$

where $a(\alpha, \beta, j_1, \ldots, j_n)$ denotes the coefficients of the above polynomial, and

$$\begin{aligned} &\mathrm{Vol}_n(\ell_1 \alpha Q_1 + \ell_1 \beta \bar{Q}_1 + \cdots + \ell_n Q_n) \\ &= \sum_{j_1+j_1'+\cdots+j_n=n} b(j_1, j_1', \ldots, j_n)(\ell_1\alpha)^{j_1}(\ell_1\beta)^{j_1'} \cdots \ell_n^{j_n} \end{aligned} \tag{6.8}$$

in which the coefficients are denoted by $b(j_1, j_1', \ldots, j_n)$. Comparing the coefficients of $\ell_1 \cdots \ell_n$ in (6.7) and (6.8) yields

$$a(\alpha, \beta, 1, \ldots, 1) = \alpha\, b(1, 0, 1, \ldots, 1) + \beta\, b(0, 1, \ldots, 1) \tag{6.9}$$

Letting (1) $\alpha = \beta = 1$, (2) $\alpha = 1, \beta = 0$, and (3) $\alpha = 0, \beta = 1$ in (6.9) respectively yields

$$\begin{aligned} \mathcal{M}(S_1 + \bar{S}_1, \ldots, S_n) &= a(1, \ldots, 1) = b(1, 0, 1, \ldots, 1) + b(0, 1, \ldots, 1) \\ &= a(1, 0, 1, \ldots, 1) + a(0, 1, \ldots, 1) \\ &= \mathcal{M}(S_1, \ldots, S_n) + \mathcal{M}(\bar{S}_1, \ldots, S_n). \end{aligned}$$

4: (*Automorphism invariant*) For linear transformation U,

$$\mathrm{Vol}_n(U(\ell_1 Q_1 + \cdots + \ell_n Q_n)) = |\det U| \,\mathrm{Vol}_n(\ell_1 Q_1 + \cdots + \ell_n Q_n)$$

Therefore, when $\det U = \pm 1$,

$$\begin{aligned} \mathrm{Vol}_n(\ell_1(UQ_1) + \cdots + \ell_n(UQ_n)) &= \mathrm{Vol}_n(U(\ell_1 Q_1 + \cdots + \ell_n Q_n)) \\ &= \mathrm{Vol}_n(\ell_1 Q_1 + \cdots + \ell_n Q_n), \end{aligned}$$

and consequently,

$$\mathcal{M}(US_1, \ldots, US_n) = \mathcal{M}(S_1, \ldots, S_n).$$

The above connection between $L(S_1, \ldots, S_n)$ and $\mathcal{M}(S_1, \ldots, S_n)$ suggested the following Bernshteín theorem:

Theorem 6.4 ([9, Theorem A]). *The number of isolated zeros, counting multiplicities, in* $(\mathbb{C}^*)^n$ *of a polynomial system* $P(\mathbf{x}) = (p_1(\mathbf{x}), \ldots, p_n(\mathbf{x}))$ *with support* $S = (S_1, \ldots, S_n)$ *is bound above by the mixed volume* $\mathcal{M}(S_1, \ldots, S_n)$. *When* $P(\mathbf{x})$ *is in general position, it has exactly* $\mathcal{M}(S_1, \ldots, S_n)$ *isolated zeros in* $(\mathbb{C}^*)^n$.

In [14], the root count in the above theorem was nicknamed the *BKK bound* after the works of Bernshtein [9], Kushnirenko [50] and Khovanskii [48]. In general, it provides a much tighter bound of the number of isolated zeros of a polynomial system compared to variant Bézout bounds discussed in the last section. However, this theorem does have an apparent limitation: it only counts the number of isolated zeros of a polynomial system in $(\mathbb{C}^*)^n$ rather than the number of all isolated zeros in affine space $\mathbb{C}^n$. For counting the number of all isolated zeros of a polynomial system in $\mathbb{C}^n$, a more general version of the theorem is strongly desirable. This problem was first attempted in [91], a bound for the root count in $\mathbb{C}^n$ was obtained via the notion of the *shadowed* sets. Later, a significantly tighter bound was discovered in the following theorem.

Theorem 6.5 ([70]). *The root count in* $\mathbb{C}^n$ *of a polynomial system* $P(\mathbf{x}) = (p_1(\mathbf{x}), \ldots, p_n(\mathbf{x}))$ *with supports* $S = (S_1, \ldots, S_n)$ *is bounded above by the mixed volume* $\mathcal{M}(S_1 \cup \{\mathbf{0}\}, \ldots, S_n \cup \{\mathbf{0}\})$.

In other words, the root count of a polynomial system $P(\mathbf{x}) = (p_1(\mathbf{x}), \ldots, p_n(\mathbf{x}))$ in $\mathbb{C}^n$ is bounded above by the root count in $(\mathbb{C}^*)^n$ of the polynomial system $\bar{P}(\mathbf{x})$ in general position obtained by augmenting constant terms to those p_j's in $P(\mathbf{x})$ which do not have constant terms. As a corollary, when $\mathbf{0} \in S_j$ for all $j = 1, \ldots, n$, namely, all $p_j(\mathbf{x})$'s in $P(\mathbf{x})$ have constant terms, then the mixed volume $\mathcal{M}(S_1, \ldots, S_n)$ of $P(\mathbf{x})$ is a bound for the root count of $P(\mathbf{x})$ in $\mathbb{C}^n$, more than just the root count of $P(\mathbf{x})$ in $(\mathbb{C}^*)^n$.

This theorem was further extended in several different ways (see [43, 92]). §10 will explore one such extension in detail.

6.2. Mixed volume and fine mixed subdivision

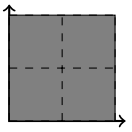
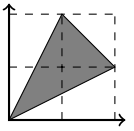
Let us take a look at the system of two polynomials

$$F(x_1, x_2) = \begin{cases} f_1(x_1, x_2) = c_1 x_1^2 x_2^2 + c_2 x_1^2 + c_3 x_2^2 + c_4 \\ f_2(x_1, x_2) = c_5 x_1^2 x_2 + c_6 x_1 x_2^2 + c_7. \end{cases} \tag{6.10}$$

The supports of these two polynomials are

$$S_1 = \{(2,2),(2,0),(0,2),(0,0)\}$$
$$S_2 = \{(2,1),(1,2),(0,0)\}$$

and the Newton polytopes of f_1 and f_2 are

$Q_1 = \operatorname{conv} S_1 =$  and $Q_2 = \operatorname{conv} S_2 =$ 

respectively. As before, the mixed volume of (Q_1, Q_2) can be computed by the formula

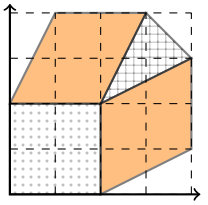
$$\mathcal{M}(Q_1, Q_2) = \text{area of } (Q_1 + Q_2) - \text{area of } (Q_1) - \text{area of } (Q_2).$$

However, when the polynomial system has more equations, this formula becomes

$$\mathcal{M}(Q_1, \ldots, Q_n) = (-1)^{n-1} \sum_{i=1}^{n} \operatorname{Vol}_n(Q_i) + (-1)^{n-2} \sum_{i<j} \operatorname{Vol}_n(Q_i + Q_j) + \cdots + \operatorname{Vol}_n(Q_1 + \cdots + Q_n).$$

Practically, it is difficult to use this somewhat complicated formula for mixed volume computations in general. To efficiently compute the mixed volume we must look for other formulations.

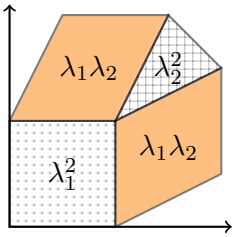
Let's look at the following subdivision of the Minkowski sum $Q_1 + Q_2$

$Q_1 + Q_2 \;=$ 

in which the dotted portion is an exact copy of Q_1 while the crossed portion is a translated copy of Q_2. The remaining (shaded) parts come from the Minkowski sum of "mixing" edges of Q_1 and Q_2.

The mixed volume of (Q_1, Q_2) is defined via the scaled versions $\lambda_1 Q_1$ and $\lambda_2 Q_2$ with scaling factors $\lambda_1, \lambda_2 \in \mathbb{R}^+$ and their Minkowski sum $\lambda_1 Q_1 + \lambda_2 Q_2$. Importantly, if we assume the original partition never *deteriorate* after

scaling, namely, the original partition of $Q_1 + Q_2$ stays as a partition of $\lambda_1 Q_1 + \lambda_2 Q_2$ after scaling, then the area of those individual parts in $\lambda_1 Q_1 + \lambda_2 Q_2$ are scaled by proper factors: The areas of the copies of Q_1 (dotted) and Q_2 (crossed) are scaled by λ_1^2 and λ_2^2 respectively, and the areas of the "mixed" parts are scaled by $\lambda_1\lambda_2$ as marked in the picture below:



So the area of the above polytope, under the mixed scaling, can be decomposed into

$$\mathrm{Vol}_2(\lambda_1 Q_1 + \lambda_2 Q_2) = \lambda_1^2 \; + \; \lambda_2^2 \;$$

$$+ \; \lambda_1\lambda_2 \; + \; \lambda_1\lambda_2 \;$$

which is a homogeneous polynomial of degree 2 in λ_1 and λ_2. Recall that the mixed volume $\mathcal{M}(Q_1, Q_2)$ is defined to be the coefficient of the term $\lambda_1\lambda_2$ in the above polynomial. Therefore,

$$\mathcal{M}(Q_1, Q_2) = \; + \; = 4 + 4 = 8,$$

and the BKK bound for the system (6.10) is then 8.

Assembling the mixed volume $\mathcal{M}(Q_1, Q_2)$ in this manner is independent of the scaling factors λ_1 and λ_2. It is valid for any such subdivisions of $Q_1 + Q_2$ as given above which never *deteriorate* after mixed scaling. That is, those original subdivisions stay as subdivisions under mixed scalings. They are known as the *mixed subdivisions*. To state a formal definition for such subdivisions with less notations, we shall omit those "+" and "conv" in most of the occasions. For instance, instead of formulating the subdivision for $Q_1 + \cdots + Q_n (= \mathrm{conv}(S_1) + \cdots + \mathrm{conv}(S_n))$ we shall deal with the n-tuple $(S_1, \ldots, S_n)$ for short.

Let $S = (S_1, \ldots, S_n)$ be a set of finite subsets of $\mathbb{N}_0^n$, whose union affinely spans $\mathbb{R}^n$. By a **cell** of $S = (S_1, \ldots, S_n)$ we mean a tuple $C = (C_1, \ldots, C_n)$ of nonempty subsets $C_i \subseteq S_i$, and each C_i is called a **component** of this cell. For brevity, the following notations will be used throughout this section:

$$\begin{aligned} \text{type } C &= (\dim(\text{conv } C_1), \ldots, \dim(\text{conv } C_n)) \\ \text{conv } C &= \text{conv } C_1 + \cdots + \text{conv } C_n \\ \text{Vol } C &= \text{Vol}(\text{conv } C). \end{aligned}$$

In the above, it is easy to verify that conv C is a convex polytope. Recall that a *polytope* $P \subset \mathbb{R}^n$ is the convex hull of finite many points in $\mathbb{R}^n$. A subset F of polytope P is called a **face** of P if there exists $\boldsymbol{\alpha} \in \mathbb{R}^n$ for which the linear functional $f(\mathbf{x}) = \langle \boldsymbol{\alpha}, \mathbf{x} \rangle$ for $\mathbf{x} \in \mathbb{R}^n$ attains its minimum over P at F, and the vector $\boldsymbol{\alpha}$ is called an *inner normal* of F. Here and after, $\langle\ ,\ \rangle$ stands for the usual inner product in Euclidean spaces. When P is a k-dimensional polytope in $\mathbb{R}^n$, its $(k-1)$-dimensional faces are called *facets* of P.

Definition 6.6. A **subdivision** of $S = (S_1, \ldots, S_n)$ is a collection $\mathcal{D}$ of cells $C = (C_1, \ldots, C_n)$ of $S = (S_1, \ldots, S_n)$ such that

(a): $\dim(\text{conv } C) = n$ for all $C \in \mathcal{D}$,

(b): For a distinct pair $A, B \in \mathcal{D}$, if conv $A \cap$ conv B is nonempty, then it is a common face of both,

(c): $\bigcup_{C \in \mathcal{D}} \text{conv } C = \text{conv } S$.

Note that the above conditions characterize precisely a subdivision of the single convex polytope $\tilde{Q} = Q_1 + \cdots + Q_n$ of dimension n in the familiar sense: a collection of convex polytopes of dimension n in $\tilde{Q}$ whose mutual intersections only appear on their common faces and whose union is the entire $\tilde{Q}$. While a proper subdivision of $\tilde{Q}$ is important in computing the volume of $\tilde{Q}$ as long as the volume of each sub-polytope is easy to attain, in the study of mixed volume, it is perhaps more important to find the expression of the volume of the Minkowski sum $\lambda_1 Q_1 + \cdots + \lambda_n Q_n$ in terms of the scaling factors $\lambda_1, \ldots, \lambda_n$. For this purpose, merely a subdivision as given above is insufficient.

For $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_n) \in (\mathbb{R}^+)^n$, we shall use the notation $\boldsymbol{\lambda} \circ S$ for the scaled version $(\lambda_1 S_1, \ldots, \lambda_n S_n)$, and $\boldsymbol{\lambda} \circ \tilde{Q}$ for $\lambda_1 \text{conv } S_1 + \cdots + \lambda_n \text{conv } S_n = \lambda_1 Q_1 + \cdots + \lambda_n Q_n$. This notation also applies to individual cells: $\boldsymbol{\lambda} \circ (C_1, \ldots, C_n) = (\lambda_1 C_1, \ldots, \lambda_n C_n)$ for any cell $C = (C_1, \ldots, C_n)$ of $S = (S_1, \ldots, S_n)$. Under this "mixed" scaling, a general subdivision of $S = (S_1, \ldots, S_n)$ may

not behave properly. That is, a subdivision of $S = (S_1, \ldots, S_n)$ may not stay, after mixed scaling, as a subdivision of $\boldsymbol{\lambda} \circ S = (\lambda_1 S_1, \ldots, \lambda_n S_n)$ correspondingly. A subdivision of $S = (S_1, \ldots, S_n)$ is called *scaling invariant* if after mixed scaling, the collection of scaled cells in the subdivision remains as a subdivision of $\boldsymbol{\lambda} \circ S = (\lambda_1 S_1, \ldots, \lambda_n S_n)$. To characterize such subdivisions, we add additional restrictions:

Definition 6.7. A subdivision $\mathcal{D}$ of $S = (S_1, \ldots, S_n)$ is called a **mixed subdivision** if, in addition, it satisfies

(d1): For each cell $C = (C_1, \ldots, C_n) \in \mathcal{D}$, $\sum_{j=1}^{n} \dim(\operatorname{conv} C_j) = n$

(d2): For any distinct pair of cells $A = (A_1, \ldots, A_n)$, $B = (B_1, \ldots, B_n) \in \mathcal{D}$,

$$(\operatorname{conv} A) \cap (\operatorname{conv} B) = (\operatorname{conv} A_1 \cap \operatorname{conv} B_1) + \cdots + (\operatorname{conv} A_n \cap \operatorname{conv} B_n).$$

Cells of a mixed subdivision are called **mixed cells**.

A mixed subdivision of $S = (S_1, \ldots, S_n)$ may be refined via further subdivision of individual components of each cell. It is computationally beneficial (as Equation (6.13) later in the section will show) to utilize the most refined mixed subdivisions:

Definition 6.8. A mixed subdivision $\mathcal{D}$ is called a **fine mixed subdivision** if it also satisfies the following condition:

(e): For each cell $C = (C_1, \ldots, C_n) \in D$, $\operatorname{conv} C_j$ is a simplex of dimension $\#C_j - 1$ for $j = 1, \ldots, n$.

The importance of mixed subdivisions (and fine mixed subdivisions) lies in their nice behavior under the "mixed" scaling by positive factors $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_n)$. They are "scaling invariant" as mentioned above. Moreover, under this condition, one can establish the tie between the cells in a subdivision and the expression of $\operatorname{Vol}_n(\lambda_1 Q_1 + \cdots + \lambda_n Q_n)$ by which the mixed volume is defined.

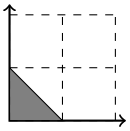
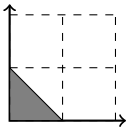
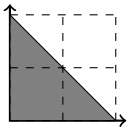
Proposition 6.9. *Let $\mathcal{D}$ be a mixed subdivision of $S = (S_1, \ldots, S_n)$. For any $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_n) \in (\mathbb{R}^+)^n$, the set*

$$\boldsymbol{\lambda} \circ \mathcal{D} := \{(\lambda_1 C_1, \ldots, \lambda_n C_n) \mid (C_1, \ldots, C_n) \in \mathcal{D}\}$$

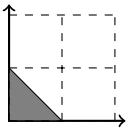
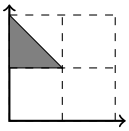
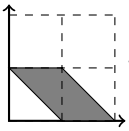
forms a mixed subdivision of $\boldsymbol{\lambda} \circ S = (\lambda_1 S, \ldots, \lambda_n S)$. *Furthermore, if* $\mathcal{D}$ *is a fine mixed subdivision of* S, *then* $\boldsymbol{\lambda} \circ \mathcal{D}$ *is a fine mixed subdivision of* $\boldsymbol{\lambda} \circ S = (\lambda_1 S, \ldots, \lambda_n S)$.

This proposition actually lays the ground for the version of the mixed volume computation algorithms to be discussed in Section 8. The complete proof is somewhat technical, please see [18]. Here we only want to emphasize that the condition **(d2)** plays a critical role in the proof and illustrate the subtlety via a counter example:

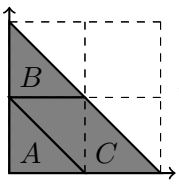
Example 6.10. For two supports $S_1 = \{(0,0),(1,0),(0,1)\}$ and $S_2 = \{(0,0),\ (1,0),(0,1)\}$ in $\mathbb{R}^2$ and their convex hulls $Q_1 = \operatorname{conv} S_1$ and $Q_2 = \operatorname{conv} S_2$, we have

$Q_1 =$  $Q_2 =$  and $Q_1 + Q_2 =$  .

Let $A = (\{(0,0),(1,0),(0,1)\},\{(0,0)\}), B = (\{(0,0),(1,0),(0,1)\},\{(0,1)\})$, and $C = (\{(1,0),(0,1)\},\{(0,0),(1,0)\})$ be the three cells of $\mathcal{D} = \{A, B, C\}$. That is,

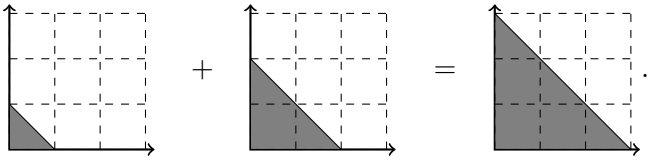
$\operatorname{conv} A =$  $\operatorname{conv} B =$  $\operatorname{conv} C =$  .

Clearly, $\dim(\operatorname{conv} A) = \dim(\operatorname{conv} B) = \dim(\operatorname{conv} C) = 2$ and the intersection of any two is their common face. Furthermore, the union of the three

$\operatorname{conv} A \ \cup\ \operatorname{conv} B \ \cup\ \operatorname{conv} C \ =$ 

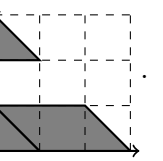
is indeed the entire $Q_1 + Q_2$. Therefore $\mathcal{D}$ satisfies the definition of a subdivision (conditions **(a),(b),(c)**). Moreover, since cells A, B, and C are of type $(2,0),(2,0)$, and $(1,1)$ respectively, $\mathcal{D}$ even satisfies the condition **(d1)**. However, it does not behave as expected under the scaling by $\boldsymbol{\lambda} = (\lambda_1, \lambda_2)$. The cells will both separate and overlap as one chooses different scaling factors.

For example, with factors $\boldsymbol{\lambda} = (\lambda_1, \lambda_2) = (1, 2)$, the scaled version $1 \cdot Q_1 + 2 \cdot Q_2$ is



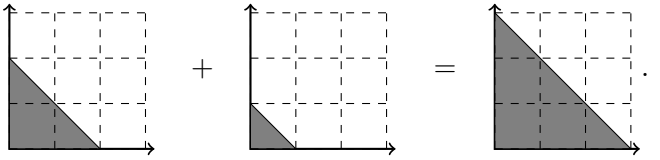
But under the same scaling factors the cells separate and the union becomes

$$\mathrm{conv}((1,2) \circ A) \ \cup \ \mathrm{conv}((1,2) \circ B) \ \cup \ \mathrm{conv}((1,2) \circ C) \ =$$

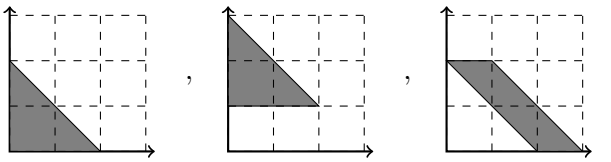


It no longer covers the entire $1 \cdot Q_1 + 2 \cdot Q_2$.

Alternatively, with the scaling factor $\boldsymbol{\lambda} = (2, 1)$, $2 \cdot Q_1 + 1 \cdot Q_2$ is



Under the same scaling, the cells $\mathrm{conv}((2,1) \circ A)$, $\mathrm{conv}((2,1) \circ B)$, $\mathrm{conv}((2,1) \circ C)$ are



respectively. Apparently there are overlaps among those three.

In both cases, with certain scalings $\boldsymbol{\lambda} \circ \mathcal{D}$ failed to form a subdivision of $\boldsymbol{\lambda} \circ Q = \lambda_1 Q_1 + \lambda_2 Q_2$. The main reason is the subdivision $\mathcal{D}$ of $S = (S_1, S_2)$ does not satisfy the condition **(d2)**.

Remark 6.11. The condition **(d2)** was absent when the "mixed subdivision" was originally defined in [42]. It first appeared in [22].

As shown in the above example, the condition **(d2)** is crucial to ensure a mixed subdivision transforms properly under mixed scaling in the sense of Proposition 6.9. The condition **(d1)**, on the other hand, relates the volume

$\mathrm{Vol}_n(\lambda_1 Q_1 + \cdots + \lambda_n Q_n)$ directly to the volumes of individual cells of a mixed subdivision:

Proposition 6.12. *If $\mathcal{D}$ is a mixed subdivision, then for a cell $C \in \mathcal{D}$ of type $(t_1, \ldots, t_n)$,*

$$\mathrm{Vol}_n(\mathrm{conv}(\boldsymbol{\lambda} \circ C)) = \lambda_1^{t_1} \cdots \lambda_n^{t_n}\ \mathrm{Vol}_n(\mathrm{conv}\, C). \tag{6.11}$$

(See [18] for a detailed proof.)

Combining Propositions 6.9 and 6.12 yields the important theorem which links the mixed volume to the mixed cells of type $(1, \ldots, 1)$.

Theorem 6.13. *Let $\mathcal{D}$ be a mixed subdivision of $S = (S_1, \ldots, S_n)$, then*

$$\mathcal{M}(Q_1, \ldots, Q_n) = \sum_{\substack{C \in \mathcal{D} \\ \text{type } C = (1,\ldots,1)}} \mathrm{Vol}_n(\mathrm{conv}\, C) \tag{6.12}$$

where $Q_j = \mathrm{conv}\, S_j$ for each $j = 1, \ldots, n$.

Proof. Note that for each cell $C \in \mathcal{D}$ of type $(t_1, \ldots, t_n)$, condition **(d1)** in the definition of a mixed subdivision requires $t_1 + \cdots + t_n$ to be exactly n. Therefore, by (6.11), each term in the polynomial $\mathrm{Vol}_n(\lambda_1 Q_1 + \cdots + \lambda_n Q_n)$ in the variables $\lambda_1, \ldots, \lambda_n$ has total degree n and hence this polynomial is homogeneous of degree n. By definition, the mixed volume of $Q = (Q_1, \ldots, Q_n)$ is the coefficient of the monomial $\lambda_1 \times \cdots \times \lambda_n$ in this polynomial, thus, the mixed volume is the sum of volumes of those type $(1, \ldots, 1)$ cells in $\mathcal{D}$ because of (6.11). □

In general, the volume of a cell may still be difficult to compute. However, if $\mathcal{D}$ is a *fine* mixed subdivision, then a cell $C \in \mathcal{D}$ of type $(1, \ldots, 1)$ is necessarily of the form $C = (\{\mathbf{a}_1, \mathbf{a}'_1\}, \ldots, \{\mathbf{a}_n, \mathbf{a}'_n\})$ with $\{\mathbf{a}_j, \mathbf{a}'_j\} \subseteq S_j$ for $j = 1, \ldots, n$ since each of its component must contain exactly $1 + 1 = 2$ points by condition **(e)**. The implication is $\mathrm{conv}\, C = \mathrm{conv}\{\mathbf{a}_1, \mathbf{a}'_1\} + \cdots + \mathrm{conv}\{\mathbf{a}_n, \mathbf{a}'_n\}$ becomes a Minkowski sum of n affinely independent line segments which is an n-dimensional parallelepiped. Its volume can be computed as follows:

$$\mathrm{Vol}_n(C) = \left| \det \begin{bmatrix} \mathbf{a}_1^\top - \mathbf{a}_1'^\top \\ \vdots \\ \mathbf{a}_n^\top - \mathbf{a}_n'^\top \end{bmatrix} \right|. \tag{6.13}$$

The above construction reveals a clear strategy for computing mixed volume: With the construction of a fine mixed subdivision of $S = (S_1, \ldots, S_n)$,

if one can systematically enumerate all the mixed cells of type $(1, \ldots, 1)$, then the sum of the volume of all these cells as given in (6.13) is precisely the mixed volume $\mathcal{M}(Q_1, \ldots, Q_n)$.

6.3. Mixed subdivisions induced by generic lifting

In this section we discuss a procedure, developed in [11, 42], with which a fine mixed subdivision of $S = (S_1, \ldots, S_n)$ can be constructed.

For each $j = 1, \ldots, n$, let $\omega_j : S_j \to \mathbb{R}$ be a function that assigns each point in S_j a real number. The function $\boldsymbol{\omega} := (\omega_1, \ldots, \omega_n)$ is known as a **lifting function** on $S = (S_1, \ldots, S_n)$. We say ω_j *lifts* S_j to its graph $\hat{S}_j = \{(\mathbf{a}, \omega_j(\mathbf{a})) : \mathbf{a} \in S_j\} \subset \mathbb{R}^{n+1}$ for each $j = 1, \ldots, n$,. This notation can be extended in the obvious way: $\hat{S} = (\hat{S}_1, \ldots, \hat{S}_n)$, $\hat{Q}_j = \text{conv}(\hat{S}_j)$, $\hat{Q} = \hat{Q}_1 + \cdots + \hat{Q}_n$, etc. The lifting function $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_n)$ is known as a *generic lifting* in the sense given below. Let $\pi : \mathbb{R}^{n+1} \to \mathbb{R}^n$ be the projection by erasing the last coordinate. So, $\pi(\hat{S}_j) = S_j$ for each $j = 1, \ldots, n$, and $\pi(\text{conv}\, \hat{S}) = \text{conv}\, S$.

Consider the polytope $\text{conv}\, \hat{S}$, now in $\mathbb{R}^{n+1}$. We are interested in its "lower hull" with respect to the projection π: A vector $\hat{\boldsymbol{\alpha}} \in \mathbb{R}^{n+1}$ is said to be *upward pointing* if its last coordinate is positive. Without loss, we assume the last coordinate of an upward pointing $\hat{\boldsymbol{\alpha}}$ is 1, that is, $\hat{\boldsymbol{\alpha}} = (\boldsymbol{\alpha}, 1) \in \mathbb{R}^{n+1}$. A face $\hat{F}$ of $\text{conv}\, \hat{S}$ is called a **lower face** if its inner normal is upward pointing, namely, there exists an $\hat{\boldsymbol{\alpha}} = (\boldsymbol{\alpha}, 1) \in \mathbb{R}^{n+1}$ such that

$$\hat{F} = (\text{conv}\, \hat{S})_{\hat{\boldsymbol{\alpha}}} := \left\{ \mathbf{x} \in \text{conv}\, \hat{S} \mid \langle \hat{\boldsymbol{\alpha}}, \mathbf{x} \rangle = \min_{\mathbf{y} \in \text{conv}\, \hat{S}} \langle \hat{\boldsymbol{\alpha}}, \mathbf{y} \rangle \right\}.$$

It is important to note that for a lower face $\hat{F}$ of $\text{conv}\, \hat{S}$, one can show that

$$\hat{F} = (\text{conv}\, \hat{S})_{\hat{\boldsymbol{\alpha}}} = (\text{conv}\, \hat{S}_1)_{\hat{\boldsymbol{\alpha}}} + \cdots + (\text{conv}\, \hat{S}_n)_{\hat{\boldsymbol{\alpha}}}$$

for some upward pointing inner normal $\hat{\boldsymbol{\alpha}}$. In other words, a lower face of $\text{conv}\, \hat{S}$ is necessarily a Minkowski sum of n faces of $\text{conv}\, \hat{S}_1, \ldots, \text{conv}\, \hat{S}_n$ respectively, they share a common inner normal of the form $\hat{\boldsymbol{\alpha}} = (\boldsymbol{\alpha}, 1)$. The **lower hull** of $\text{conv}\, \hat{S}$ is the collection of all its n-dimensional lower facets.

We shall impose a "genericity" condition on the lifting function. To facilitate the discussion, the following notation will be used: Fix any $j \in \{1, \ldots, n\}$ and a subset $X_j = \{\mathbf{x}_{j,1}, \ldots, \mathbf{x}_{j,m_j}\} \subseteq S_j$, containing m_j points

for some $m_j > 0$, define

$$V(X_j) = \begin{pmatrix} \mathbf{x}_{j,2}^\top - \mathbf{x}_{j,1}^\top \\ \mathbf{x}_{j,3}^\top - \mathbf{x}_{j,1}^\top \\ \vdots \\ \mathbf{x}_{j,m_j}^\top - \mathbf{x}_{j,1}^\top \end{pmatrix} \quad \text{and} \quad \Omega(X_j) = \begin{pmatrix} \omega_j(\mathbf{x}_{j,1}) - \omega_j(\mathbf{x}_{j,2}) \\ \omega_j(\mathbf{x}_{j,1}) - \omega_j(\mathbf{x}_{j,2}) \\ \vdots \\ \omega_j(\mathbf{x}_{j,1}) - \omega_j(\mathbf{x}_{j,m_j}) \end{pmatrix}. \tag{6.14}$$

Definition 6.14. A lifting function $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_n)$ for $S = (S_1, \ldots, S_n)$ is said to be **generic** if for any choice of n (possibly empty) subsets $X_j = \{\mathbf{x}_{j,1}, \ldots, \mathbf{x}_{j,m_j}\} \subseteq S_j$ for $j = 1, \ldots, n$ with $m_j \geq 0$ the linear system

$$\begin{pmatrix} V(X_1) \\ V(X_2) \\ \vdots \\ V(X_n) \end{pmatrix} \cdot \boldsymbol{\alpha} = \begin{pmatrix} \Omega(X_1) \\ \Omega(X_2) \\ \vdots \\ \Omega(X_n) \end{pmatrix} \tag{6.15}$$

in $\boldsymbol{\alpha}$ has a solution only when the rank of the matrix on the left equals the number of its rows. Note that if the subset X_j is empty, the blocks $V(X_j)$ and $\Omega(X_j)$ will not appear in the above equation.

Remark 6.15. By this definition, *almost all* liftings are generic, justifying the choice of the terminology. More precisely, for each $j = 1, \ldots, n$, we can identify $\omega_j : S_j \to \mathbb{R}$ with its images and regard ω_j as an element in $\mathbb{R}^{N_j}$ where $N_j = \#S_j$. Similarly, we may consider $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_n)$ as an element in $\mathbb{R}^N$ where $N = N_1 + \cdots + N_n$. If $\boldsymbol{\omega}$ is not generic, then there exists a choice of n (possibly empty) subsets $\{\mathbf{x}_{j,1}, \ldots, \mathbf{x}_{j,m_j}\} \subseteq S_j$ for $j = 1, \ldots, n$ with $m_j > 0$ for which the rank of the matrix on the left hand side of the linear system (6.15) is less than the number of its rows but the system has a solution. This condition forces $\boldsymbol{\omega}$ to be in an affine subspace of lower dimension. Since there are only finite many ways of choosing subsets of $S_1, \ldots, S_n$, the set of non-generic lifting is hence contained in a finite union of lower dimensional affine subspaces of $\mathbb{R}^N$ determined by points in S. This set is necessarily of measure zero. Indeed, it is closed and nowhere dense. This is of great practical importance: one can choose a lifting at random, which would be generic with probability one. Moreover, under this interpretation of the genericity, it is reasonable to choose rational lifting values only. Since the set of "non-generic" rational lifting is also contained in a finite union of lower dimensional affine subspace of $\mathbb{Q}^N$, with any reasonable probability distribution one imposes on $\mathbb{Q}^N$, the probability of picking a "non-generic"

rational lifting should also be zero. This fact will become important in the construction of polyhedral homotopy to be discussed in §7.

Proposition 6.16 (Induced fine mixed subdivision). *Let* $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_n)$ *be a generic lifting for* $S = (S_1, \ldots, S_n)$*, and let* $\hat{D}_{\boldsymbol{\omega}}$ *be the collection of all* $\hat{C} = (\hat{C}_1, \ldots, \hat{C}_n)$ *with* $\hat{C}_j \subseteq \hat{S}_j$ *for each* $j = 1, \ldots, n$ *such that*

1) $\operatorname{conv} \hat{C}_j$ *is a lower face of* $\operatorname{conv} \hat{S}_j$ *for each* $j = 1, \ldots, n$*;*
2) *Those* n *lower faces* $\operatorname{conv} \hat{C}_j$ *of* $\operatorname{conv} \hat{S}_j$ *for* $j = 1, \ldots, n$ *respectively share a common inner normal of the form* $\hat{\boldsymbol{\alpha}} = (\boldsymbol{\alpha}, 1)$ *where* $\boldsymbol{\alpha} \in \mathbb{R}^n$*; and*
3) $\dim(\operatorname{conv} \hat{C}_1) + \cdots + \dim(\operatorname{conv} \hat{C}_n) = n$.

Then the projections of all cells in $\hat{\mathcal{D}}_{\boldsymbol{\omega}}$

$$\mathcal{D}_{\boldsymbol{\omega}} = \{(\pi(\hat{C}_1), \ldots, \pi(\hat{C}_n)) \mid \hat{C} = (\hat{C}_1, \ldots, \hat{C}_n) \in \hat{\mathcal{D}}_{\boldsymbol{\omega}}\}$$

form a fine mixed subdivision of $S = (S_1, \ldots, S_n)$*. It is called the* ***subdivision induced by the lifting function*** $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_n)$.

(See [18] for a detailed proof.)

The subdivision I in Figure 12 for system (6.1) is, in fact, induced by the lifting $\omega = ((0, 1, 1, 1), (0, 0, 0))$, that is

$$\hat{S} = (\{(a, 0), (b, 1), (c, 1), (d, 1)\}, \{(e, 0), (f, 0), (g, 0)\}).$$

7. Polyhedral homotopy

In finding all isolated zeros of a given polynomial system $P(x_1, \ldots, x_n) = P(\mathbf{x}) = (p_1(\mathbf{x}), \ldots, p_n(\mathbf{x}))$ in $\mathbb{C}^n$, we wish to take advantage of generically much tighter bound of the root count, *mixed volume*, as discussed in previous sections.

In light of Theorems 5.5 and 6.5, to find all isolated zeros of a polynomial system $P(\mathbf{x}) = (p_1(\mathbf{x}), \ldots, p_n(\mathbf{x}))$ in $\mathbb{C}^n$ with support $S = (S_1, \ldots, S_n)$, we first augment the system by appending the monomial $\mathbf{x}^{\mathbf{0}} = x_1^0 \cdots x_n^0 = 1$ to those p_j's in $P(\mathbf{x})$ which do not have constant terms. Then generic coefficients are assigned for all the monomials. The resulting new system $Q(\mathbf{x})$ has supports $S_1', \ldots, S_n'$ with $S_j' = S_j \cup \{\mathbf{0}\}$ for $j = 1, \ldots, n$. We shall solve $Q(\mathbf{x}) = \mathbf{0}$ in the first place. After $Q(\mathbf{x}) = \mathbf{0}$ is solved, consider the linear

homotopy

$$H(\mathbf{x},t) = (1-t)\,c\,Q(\mathbf{x}) + t\,P(\mathbf{x}) = \mathbf{0} \quad \text{for generic } c \in \mathbb{C}^*. \tag{7.1}$$

By Theorem 5.5, Properties 1 and 2 (Smoothness and Accessibility) hold for this homotopy, and because all the isolated solutions of $Q(\mathbf{x}) = \mathbf{0}$ are known, Property 0 (triviality) also holds. Therefore, every isolated zero of $P(\mathbf{x})$ lies at the end of a homotopy path defined by $H(\mathbf{x},t) = \mathbf{0}$, emanating from an isolated solution of $Q(\mathbf{x}) = \mathbf{0}$.

To solve $Q(\mathbf{x}) = \mathbf{0}$, write

$$Q(\mathbf{x}) = \begin{cases} q_1(\mathbf{x}) = \sum_{\mathbf{a}\in S'_1} \bar{c}_{1,\mathbf{a}}\mathbf{x}^{\mathbf{a}}, \\ \qquad\vdots \\ q_n(\mathbf{x}) = \sum_{\mathbf{a}\in S'_n} \bar{c}_{n,\mathbf{a}}\mathbf{x}^{\mathbf{a}}. \end{cases} \tag{7.2}$$

From Remark 6.2, there exists a polynomial $G(\mathbf{c})$ in the variables $\mathbf{c} = (c_{j,\mathbf{a}})_{j=1,\ldots,n,\,\mathbf{a}\in S'_j}$ of the coefficients in (7.2) such that Q is in general position when $G(\mathbf{c}) \neq 0$. Since all those coefficients $\bar{c}_{j,\mathbf{a}}$ for $\mathbf{a} \in S'_j$ and $j = 1, \ldots, n$ are chosen randomly, with probability one, this system is *in general position*. Namely, the root count of $Q(\mathbf{x})$ is exactly the BKK bound.

Let t be a new complex variable and consider the polynomial system $\hat{Q}(\mathbf{x},t) = (\hat{q}_1(\mathbf{x},t), \ldots, \hat{q}_n(\mathbf{x},t))$ in the $n+1$ variables $(\mathbf{x},t)$ given by

$$\hat{Q}(\mathbf{x},t) = \begin{cases} \hat{q}_1(\mathbf{x},t) = \sum_{\mathbf{a}\in S'_1} \bar{c}_{1,\mathbf{a}}\mathbf{x}^{\mathbf{a}}\, t^{\omega_1(\mathbf{a})}, \\ \qquad\vdots \\ \hat{q}_n(\mathbf{x},t) = \sum_{\mathbf{a}\in S'_n} \bar{c}_{n,\mathbf{a}}\mathbf{x}^{\mathbf{a}}\, t^{\omega_n(\mathbf{a})}, \end{cases} \tag{7.3}$$

where each $\omega_j : S'_j \to \mathbb{Q}$ for $j = 1, \ldots, n$ is a function with generically chosen rational numbers as its images. For a fixed t_0, we rewrite the system in (7.3) as

$$\hat{Q}(\mathbf{x},t_0) = \begin{cases} \hat{q}_1(\mathbf{x},t_0) = \sum_{\mathbf{a}\in S'_1} (\bar{c}_{1,\mathbf{a}}\, t_0^{\omega_1(\mathbf{a})})\,\mathbf{x}^{\mathbf{a}}, \\ \qquad\vdots \\ \hat{q}_n(\mathbf{x},t_0) = \sum_{\mathbf{a}\in S'_n} (\bar{c}_{n,\mathbf{a}}\, t_0^{\omega_n(\mathbf{a})})\,\mathbf{x}^{\mathbf{a}}. \end{cases}$$

This system is in general position if for the polynomial $G(\mathbf{c})$ mentioned above

$$T(t_0) := G((\bar{c}_{j,\mathbf{a}} t_0^{\omega_j(\mathbf{a})})_{j=1,\ldots,n,\,\mathbf{a}\in S'_j}) \neq 0.$$

The equation $T(t) = 0$ can have at most finitely many solutions, since $T(t)$ is not identically zero because $T(1) = G(\bar{c}_{j,\mathbf{a}}) \neq 0$. Let

$$t_1 = r_1 e^{i\theta_1}, \quad \ldots, \quad t_k = r_k e^{i\theta_k}$$

be the solutions of $T(t) = 0$. Then, for any $\theta \neq \theta_l$ for $l = 1, \ldots, k$, the systems $\bar{Q}(\mathbf{x}, t) = (\bar{q}_1(\mathbf{x}, t), \ldots, \bar{q}_n(\mathbf{x}, t))$ given by

$$\bar{Q}(\mathbf{x}, t) = \begin{cases} \bar{q}_1(\mathbf{x}, t) = \sum\limits_{\mathbf{a}\in S'_1} (\bar{c}_{1,\mathbf{a}} e^{i\omega_1(\mathbf{a})\theta}) \mathbf{x}^{\mathbf{a}} t^{\omega_1(\mathbf{a})} \\ \qquad\qquad \vdots \\ \bar{q}_n(\mathbf{x}, t) = \sum\limits_{\mathbf{a}\in S'_n} (\bar{c}_{n,\mathbf{a}} e^{i\omega_n(\mathbf{a})\theta}) \mathbf{x}^{\mathbf{a}} t^{\omega_n(\mathbf{a})}, \end{cases}$$

are in general position for all $t > 0$ because

$$\bar{c}_{j,\mathbf{a}} e^{i\omega_j(\mathbf{a})\theta} t^{\omega_j(\mathbf{a})} = \bar{c}_{j,\mathbf{a}} (t e^{i\theta})^{\omega_j(\mathbf{a})}$$

and

$$G(\bar{c}_{j,\mathbf{a}} (t e^{i\theta})^{\omega_j(\mathbf{a})}) = T(t e^{i\theta}) \neq 0.$$

Therefore, without loss of generality, (by choosing an angle θ at random and change the coefficients $\bar{c}_{j,\mathbf{a}}$ to $\bar{c}_{j,\mathbf{a}} e^{i\omega_j(\mathbf{a})\theta}$ if necessary) we may assume the systems $\hat{Q}(\mathbf{x}, t)$ in (7.3) are in general position for all $t > 0$. By Lemma 6.1, the systems $\hat{Q}(\mathbf{x}, t)$ in (7.3) have the same number of isolated zeros in $(\mathbb{C}^*)^n$ for all $t > 0$ and this number equals the mixed volume $\mathcal{M}(S'_1, \ldots, S'_n) =: k$.

We now regard $\hat{Q}(\mathbf{x}, t) = \mathbf{0}$ as a homotopy, commonly known as the ***polyhedral homotopy***, defined on $(\mathbb{C}^*)^n \times [0, 1]$ with target system $\hat{Q}(\mathbf{x}, 1) = Q(\mathbf{x})$. The zero set of this homotopy is made up of k homotopy paths $\mathbf{x}^{(1)}(t), \ldots, \mathbf{x}^{(k)}(t)$. Since each $\hat{q}_j(\mathbf{x}, t)$ has nonzero constant term for all $j = 1, \ldots, n$, by a standard application of generalized Sard's Theorem , all those homotopy paths are smooth with no bifurcations. Therefore, both Property 1 (Smoothness) and Property 2 (Accessibility) given in §5 hold for this homotopy. However, at $t = 0$, $\hat{Q}(\mathbf{x}, 0) \equiv \mathbf{0}$, or undefined, see Figure 13. Consequently, those homotopy paths can not get started because their starting points $\mathbf{x}^{(1)}(0), \ldots, \mathbf{x}^{(k)}(0)$ can not be identified. This problem can be resolved by the following design.

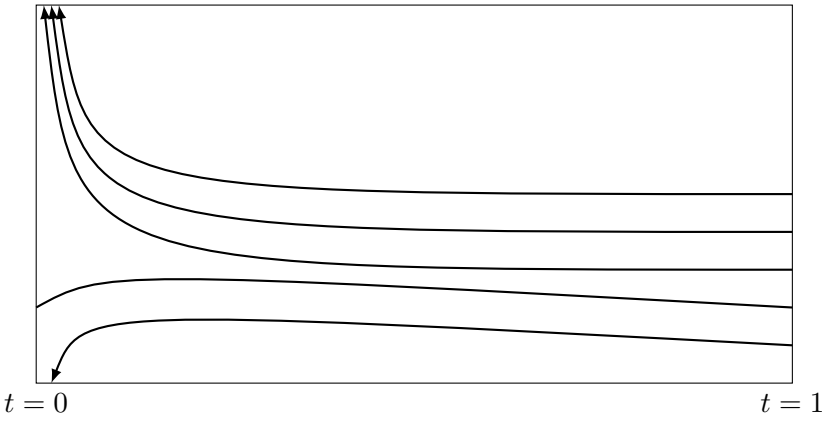


Figure 13. The starting point of the homotopy $\hat{Q}(\mathbf{x}, t)$ at $t = 0$ cannot be identified.

The function $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_n)$ with $\omega_j : S'_j \to \mathbb{Q}$, for $j = 1, \ldots, n$, may be considered as a *generic lifting* on the support $S' = (S'_1, \ldots, S'_n)$ of $Q(\mathbf{x})$ which lifts S'_j to its graph

$$\hat{S}'_j = \{\hat{\mathbf{a}} = (\mathbf{a}, \omega_j(\mathbf{a})) \mid \mathbf{a} \in S'_j\}, \quad j = 1, \ldots, n.$$

Let $\hat{\boldsymbol{\alpha}} = (\boldsymbol{\alpha}, 1)$ where $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n) \in \mathbb{Q}^n$ satisfies the following condition:

(A) *There exists a collection of pairs* $\{\mathbf{a}_1, \mathbf{a}'_1\} \subseteq S'_1, \ldots, \{\mathbf{a}_n, \mathbf{a}'_n\} \subseteq S'_n$, *such that* $\{\mathbf{a}_1 - \mathbf{a}'_1, \ldots, \mathbf{a}_n - \mathbf{a}'_n\}$ *is linearly independent and for* $j = 1, \ldots, n$,

$$\langle \hat{\mathbf{a}}_j, \hat{\boldsymbol{\alpha}} \rangle = \langle \hat{\mathbf{a}}'_j, \hat{\boldsymbol{\alpha}} \rangle$$
$$\langle \hat{\mathbf{a}}, \hat{\boldsymbol{\alpha}} \rangle > \langle \hat{\mathbf{a}}_j, \hat{\boldsymbol{\alpha}} \rangle \quad \textit{for} \quad \mathbf{a} \in S'_j \backslash \{\mathbf{a}_j, \mathbf{a}'_j\}.$$

For such $\hat{\boldsymbol{\alpha}} = (\boldsymbol{\alpha}, 1)$, let $\mathbf{y} = t^{-\boldsymbol{\alpha}}\mathbf{x}$ where $\mathbf{y} = (y_1, \ldots, y_n)$ and

$$\begin{aligned} y_1 &= t^{-\alpha_1} x_1, \\ &\vdots \\ y_n &= t^{-\alpha_n} x_n. \end{aligned} \tag{7.4}$$

With this transformation and $\mathbf{a} = (a_1, \ldots, a_n) \in \mathbb{N}_0^n$,

$$\begin{aligned} \mathbf{x}^{\mathbf{a}} &= x_1^{a_1} \cdots x_n^{a_n}, \\ &= (y_1 t^{\alpha_1})^{a_1} \cdots (y_n t^{\alpha_n})^{a_n} \\ &= y_1^{a_1} \cdots y_n^{a_n}\, t^{\alpha_1 a_1 + \cdots + \alpha_n a_n} \\ &= \mathbf{y}^{\mathbf{a}} t^{\langle \mathbf{a}, \boldsymbol{\alpha} \rangle}, \end{aligned} \tag{7.5}$$

and $\hat{q}_j(\mathbf{y}t^{\boldsymbol{\alpha}}, t)$ of $\hat{Q}(\mathbf{x}, t)$ in (7.3) becomes,

$$\begin{aligned} \hat{q}_j(\mathbf{y}t^{\boldsymbol{\alpha}}, t) &= \sum_{\mathbf{a} \in S'_j} \bar{c}_{j,\mathbf{a}} \mathbf{y}^{\mathbf{a}} t^{\langle \mathbf{a}, \boldsymbol{\alpha} \rangle} t^{\omega_j(\mathbf{a})} \\ &= \sum_{\mathbf{a} \in S'_j} \bar{c}_{j,\mathbf{a}} \mathbf{y}^{\mathbf{a}} t^{\langle (\mathbf{a}, w_j(\mathbf{a})), (\boldsymbol{\alpha}, 1)) \rangle} \\ &= \sum_{\mathbf{a} \in S'_j} \bar{c}_{j,\mathbf{a}} \mathbf{y}^{\mathbf{a}} t^{\langle \hat{\mathbf{a}}, \hat{\boldsymbol{\alpha}} \rangle}. \end{aligned} \tag{7.6}$$

Let

$$\beta_j = \min_{\mathbf{a} \in S'_j} \langle \hat{\mathbf{a}}, \hat{\boldsymbol{\alpha}} \rangle \quad \text{for } j = 1, \ldots, n, \tag{7.7}$$

and consider the homotopy

$$H^{\boldsymbol{\alpha}}(\mathbf{y}, t) = (h_1^{\boldsymbol{\alpha}}(\mathbf{y}, t), \ldots, h_n^{\boldsymbol{\alpha}}(\mathbf{y}, t)) = \mathbf{0} \tag{7.8}$$

on $(\mathbb{C}^*)^n \times [0, 1]$ where for $j = 1, \ldots, n$

$$\begin{aligned} h_j^{\boldsymbol{\alpha}}(\mathbf{y}, t) &= t^{-\beta_j} \hat{q}_j(\mathbf{y}t^{\boldsymbol{\alpha}}, t) \\ &= \sum_{\mathbf{a} \in S'_j} \bar{c}_{j,\mathbf{a}}\, \mathbf{y}^{\mathbf{a}}\, t^{\langle \hat{\mathbf{a}}, \hat{\boldsymbol{\alpha}} \rangle - \beta_j} \\ &= \sum_{\substack{\mathbf{a} \in S'_j \\ \langle \hat{\mathbf{a}}, \hat{\boldsymbol{\alpha}} \rangle = \beta_j}} \bar{c}_{j,\mathbf{a}} \mathbf{y}^{\mathbf{a}} + \sum_{\substack{\mathbf{a} \in S'_j \\ \langle \hat{\mathbf{a}}, \hat{\boldsymbol{\alpha}} \rangle > \beta_j}} \bar{c}_{j,\mathbf{a}} \mathbf{y}^{\mathbf{a}} t^{\langle \hat{\mathbf{a}}, \hat{\boldsymbol{\alpha}} \rangle - \beta_j}. \end{aligned} \tag{7.9}$$

This homotopy retains most of the properties of the homotopy $\hat{Q}(\mathbf{x}, t) = \mathbf{0}$; in particular, both Properties 1 (Smoothness) and 2 (Accessibility) remain

valid and

$$H^{\boldsymbol{\alpha}}(\mathbf{y},1)=\hat{Q}(\mathbf{y},1)=Q(\mathbf{y}). \tag{7.10}$$

From condition (A), for each $j=1,\dots,n$, $\langle\hat{\mathbf{a}}_j,\hat{\boldsymbol{\alpha}}\rangle=\left\langle\hat{\mathbf{a}}'_j,\hat{\boldsymbol{\alpha}}\right\rangle=\beta_j$ and $\langle\hat{\mathbf{a}},\hat{\boldsymbol{\alpha}}\rangle>\beta_j$ for $\mathbf{a}\in S'_j\backslash\{\mathbf{a}_j,\mathbf{a}'_j\}$, hence,

$$H^{\boldsymbol{\alpha}}(\mathbf{y},0)=\begin{cases} h_1^{\boldsymbol{\alpha}}(\mathbf{y},0)=\displaystyle\sum_{\substack{\mathbf{a}\in S'_1\\ \langle\hat{\mathbf{a}},\hat{\boldsymbol{\alpha}}\rangle=\beta_1}}\bar{c}_{1,\mathbf{a}}\mathbf{y}^{\mathbf{a}}=\bar{c}_{1,\mathbf{a}_1}\mathbf{y}^{\mathbf{a}_1}+c_{1,\mathbf{a}'_1}y^{\mathbf{a}'_1}=0,\\ \qquad\vdots\\ h_n^{\boldsymbol{\alpha}}(\mathbf{y},0)=\displaystyle\sum_{\substack{\mathbf{a}\in S'_n\\ \langle\hat{\mathbf{a}},\hat{\boldsymbol{\alpha}}\rangle=\beta_n}}\bar{c}_{n,\mathbf{a}}\mathbf{y}^{\mathbf{a}}=\bar{c}_{n,\mathbf{a}_n}\mathbf{y}^{\mathbf{a}_n}+c_{n,\mathbf{a}'_n}\mathbf{y}^{\mathbf{a}'_n}=0. \end{cases} \tag{7.11}$$

Such system is known as the *binomial system*, and its isolated solutions in $(\mathbb{C}^*)^n$ are constructively available as shown in the proof of the following

Proposition 7.1. *Under condition (A), the binomial system*

$$\begin{aligned} \bar{c}_{1,\mathbf{a}_1}\mathbf{y}^{\mathbf{a}_1}+\bar{c}_{1,\mathbf{a}'_1}\mathbf{y}^{\mathbf{a}'_1}&=0,\\ &\vdots\\ \bar{c}_{n,\mathbf{a}_n}\mathbf{y}^{\mathbf{a}_n}+\bar{c}_{n,\mathbf{a}'_n}\mathbf{y}^{\mathbf{a}'_n}&=0, \end{aligned} \tag{7.12}$$

has

$$k_{\boldsymbol{\alpha}}:=\left|\det\begin{bmatrix}\mathbf{a}_1^\top-\mathbf{a}_1'^\top\\ \vdots\\ \mathbf{a}_n^\top-\mathbf{a}_n'^\top\end{bmatrix}\right| \tag{7.13}$$

nonsingular isolated solutions in $(\mathbb{C}^*)^n$.

Proof. For $j=1,\dots,n$, let $\mathbf{v}_j=\mathbf{a}_j-\mathbf{a}'_j$. Since $\mathbf{y}\in(\mathbb{C}^*)^n$, we may rewrite the system (7.12) as

$$\begin{aligned} \mathbf{y}^{\mathbf{v}_1}&=b_1,\\ &\vdots\\ \mathbf{y}^{\mathbf{v}_n}&=b_n, \end{aligned} \tag{7.14}$$

where $b_j = -\frac{\bar{c}_{j,\mathbf{a}'_j}}{\bar{c}_{j,\mathbf{a}_j}}$ for $j = 1, \ldots, n$. Let

$$V = \left[\mathbf{v}_1 \;\middle|\; \mathbf{v}_2 \;\middle|\; \cdots \;\middle|\; \mathbf{v}_n \right] \tag{7.15}$$

and $\mathbf{b} = (b_1, \ldots, b_n)$. Then, (7.14) becomes

$$\mathbf{y}^V = \mathbf{b}. \tag{7.16}$$

Now, when the matrix V in (7.15) is an upper triangular matrix, known as the *Hermite Normal Form*,[1]

$$V = \begin{bmatrix} v_{11} & v_{12} & \cdots & v_{1n} \\ 0 & v_{22} & \cdots & v_{2n} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & v_{nn} \end{bmatrix},$$

then, the equations in (7.16) become

$$\begin{aligned} y_1^{v_{11}} &= b_1, \\ y_1^{v_{12}} y_2^{v_{22}} &= b_2, \\ &\vdots \\ y_1^{v_{1n}} y_2^{v_{2n}} \cdots y_n^{v_{nn}} &= b_n. \end{aligned} \tag{7.17}$$

By forward substitutions, all the solutions of the system (7.17) in $(\mathbb{C}^*)^n$ can be found, and the total number of solutions is $|v_{11}| \times \cdots \times |v_{nn}| = |\det V|$.

When V is a general matrix, we may upper triangularize it by the following process. Recall that the greatest common divisor d of two nonzero integers a and b, denoted by $\gcd(a, b)$, can be written as

$$d = \gcd(a, b) = ra + lb,$$

for certain nonzero integers r and l. Let

$$M = \begin{bmatrix} r & l \\ -\frac{b}{d} & \frac{a}{d} \end{bmatrix},$$

[1]Usually the definition of Hermite Normal Form requires the diagonal entries to be positive and any nondiagonal entries to have absolute values that are strictly smaller than the diagonal entry in the same column. This restriction, though beneficial from a computational point of view, is not enforced here.

then $\det(M) = 1$, and

$$M \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} r & l \\ -\frac{b}{d} & \frac{a}{d} \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} d \\ 0 \end{bmatrix}.$$

Similar to using Givens rotation to produce zeros in a matrix for its QR factorization, the matrix M can be used to upper triangularize V as follows. For $\mathbf{v} \in \mathbb{Z}^n$, let a and b be its i-th and j-th (nonzero) components where $i < j$, that is,

$$\mathbf{v} = \begin{bmatrix} \vdots \\ a \\ \vdots \\ b \\ \vdots \end{bmatrix} \begin{matrix} \\ \rightarrow i\text{-th} \\ \\ \rightarrow j\text{-th.} \\ \\ \end{matrix}$$

With $d = \gcd(a, b)$, let

$$U(i,j) = \begin{array}{c} \begin{matrix} & & & i^{\text{th}} & & & j^{\text{th}} & & \end{matrix} \\ \begin{bmatrix} 1 & & & & & & & & \\ & \ddots & & & & & & & \\ & & 1 & & & & & & \\ & & & r & & & l & & \\ & & & & 1 & & & & \\ & & & & & \ddots & & & \\ & & & & & & 1 & & \\ & & & -\frac{b}{d} & & & \frac{a}{d} & & \\ & & & & & & & 1 & \ddots \\ & & & & & & & & 1 \end{bmatrix} \begin{matrix} \\ \\ \\ i^{\text{th}} \\ \\ \\ \\ j^{\text{th}} \\ \\ \end{matrix} \end{array} \tag{7.18}$$

in which all empty entries are zero. Clearly, $U(i,j)$ is an integer matrix with $|\det(U(i,j))| = 1$ and

$$U(i,j)\mathbf{v} = \begin{bmatrix} \vdots \\ d \\ \vdots \\ 0 \\ \vdots \end{bmatrix} \begin{matrix} \\ i\text{-th} \\ \\ j\text{-th} \\ \end{matrix}.$$

Thus multiplication on the left by a series of matrices in the form of $U(i,j)$ in (7.18) can successively produce zeros in the lower triangular part of the matrix V, resulting in an upper triangular matrix. Let U be the product of all those $U(i,j)$'s. Then with $|\det U| = 1$, and since UV is upper triangular, we may solve the system

$$(\mathbf{z}^U)^V = \mathbf{z}^{UV} = \mathbf{b} \tag{7.19}$$

in $(\mathbb{C}^*)^n$ by forward substitutions. And the total number of solutions in $(\mathbb{C}^*)^n$ is

$$|\det(UV)| = |\det(U)| \cdot |\det(V)| = |\det(V)|.$$

By letting $\mathbf{y} = \mathbf{z}^U$ for each solution $\mathbf{z}$ of (7.19) in $(\mathbb{C}^*)^n$, we obtain $|\det(V)|$ number of solutions of the system (7.12) in $(\mathbb{C}^*)^n$. One can show that all those solutions are nonsingular. □

Now, by (7.10), following paths $\mathbf{y}(t)$ of the homotopy $H^{\boldsymbol{\alpha}}(\mathbf{y},t) = \mathbf{0}$ in (7.8) that emanate from $k_{\boldsymbol{\alpha}}$, as in (7.13), isolated zeros in $(\mathbb{C}^*)^n$ of the binomial start system $H^{\boldsymbol{\alpha}}(\mathbf{y},0) = \mathbf{0}$ in (7.11), yields $k_{\boldsymbol{\alpha}}$ isolated zeros of the system $Q(\mathbf{x})$ in (7.2) when $t = 1$. Moreover, a different $\hat{\boldsymbol{\alpha}} = (\boldsymbol{\alpha}, 1) \in \mathbb{Q}^{n+1}$ associated with its corresponding collection of pairs that satisfy condition (A) will induce a different homotopies $H^{\boldsymbol{\alpha}}(\mathbf{y},t) = \mathbf{0}$ in (7.9). Following corresponding solution paths of those different homotopy equations will reach different sets of isolated zeros of $Q(\mathbf{x})$. Those different sets of isolated zeros of $Q(\mathbf{x})$ are actually disjoint from each other, and they hence provide $\sum_{\boldsymbol{\alpha}} k_{\boldsymbol{\alpha}}$ isolated zeros of $Q(\mathbf{x})$ in total. To see they are disjoint, let paths $\mathbf{y}^{\boldsymbol{\alpha}^{(1)}}(t)$ of $H^{\boldsymbol{\alpha}^{(1)}}(\mathbf{y},t) = \mathbf{0}$ and $\mathbf{y}^{\boldsymbol{\alpha}^{(2)}}(t)$ of $H^{\boldsymbol{\alpha}^{(2)}}(\mathbf{y},t) = \mathbf{0}$ for $\boldsymbol{\alpha}^{(1)} = (\alpha_1^{(1)}, \ldots, \alpha_n^{(1)})$ and $\boldsymbol{\alpha}^{(2)} = (\alpha_1^{(2)}, \ldots, \alpha_n^{(2)}) \in \mathbb{Q}^n$ reach the same point at $t = 1$, then their corresponding homotopy paths $\mathbf{x}(t) = \mathbf{y}(t)t^{\boldsymbol{\alpha}}$ of $\hat{Q}(\mathbf{x},t) = \mathbf{0}$ are the same since zeros of the system $Q(\mathbf{x}) = \hat{Q}(\mathbf{x},1)$ are isolated and nonsingular. Thus,

$\mathbf{x}(t) = \mathbf{y}^{\boldsymbol{\alpha}^{(1)}}(t)t^{\boldsymbol{\alpha}^{(1)}} = \mathbf{y}^{\boldsymbol{\alpha}^{(2)}}(t)t^{\boldsymbol{\alpha}^{(2)}}$ implies

$$1 = \lim_{t\to 0} \frac{y_j^{\boldsymbol{\alpha}^{(1)}}(t)}{y_j^{\boldsymbol{\alpha}^{(2)}}(t)}\, t^{\alpha_j^{(1)}-\alpha_j^{(2)}}, \quad \text{for each } j = 1,\ldots,n, \tag{7.20}$$

and therefore, $\alpha_j^{(1)} = \alpha_j^{(2)}$ for all $j = 1,\ldots,n$.

On the other hand, when $\boldsymbol{\omega} = (\omega_1,\ldots,\omega_n)$ is a generic lifting, by Proposition 6.16, it induces a fine mixed subdivision $S'_{\boldsymbol{\omega}}$ of $S' = (S'_1,...,S'_n)$. It is easy to see that the collection of pairs $C^{\boldsymbol{\alpha}} = (\{\mathbf{a}_1,\mathbf{a}'_1\},\ldots,\{\mathbf{a}_n,\mathbf{a}'_n\})$ with $\{\mathbf{a}_j,\mathbf{a}'_j\} \subset S'_j$ for each $j = 1,\ldots,n$ satisfies condition (A) with $(\boldsymbol{\alpha},1) \in \mathbb{Q}^{n+1}$ if and only if it is a cell of type $(1,\ldots,1)$ in $S'_{\boldsymbol{\omega}}$, and, by (6.13),

$$\mathrm{Vol}_n(C^{\boldsymbol{\alpha}}) = \left|\det \begin{bmatrix} \mathbf{a}_1^\top - \mathbf{a}_1'^\top \\ \vdots \\ \mathbf{a}_n^\top - \mathbf{a}_n'^\top \end{bmatrix}\right| =: \kappa_{\boldsymbol{\alpha}}. \tag{7.21}$$

By Theorem 6.13, the mixed volume $\mathcal{M}(S'_1,\ldots,S'_n)$, the root count of $Q(\mathbf{x})$ in $(\mathbb{C}^*)^n$, is the sum of the volume of all cells $C^{\boldsymbol{\alpha}}$. That is,

$$\mathcal{M}(S'_1,\ldots,S'_n) = \sum_{\boldsymbol{\alpha}} k_{\boldsymbol{\alpha}}.$$

In other words, each isolated zero of $Q(\mathbf{x})$ lies at the end of certain homotopy path of the homotopy $H^{\boldsymbol{\alpha}}(\mathbf{y},t) = \mathbf{0}$ induced by certain $\hat{\boldsymbol{\alpha}} = (\boldsymbol{\alpha},1) \in \mathbb{Q}^{n+1}$ along with its corresponding collection of pairs that satisfy condition (A).

A key step in the procedure described above for solving system $Q(\mathbf{x})$ is the search for all those vectors $\hat{\boldsymbol{\alpha}} = (\boldsymbol{\alpha},1) \in \mathbb{Q}^{n+1}$ as well as their associated cells $C^{\boldsymbol{\alpha}} = (\{\mathbf{a}_1,\mathbf{a}'_1\},\ldots,\{\mathbf{a}_n,\mathbf{a}'_n\})$ that satisfy condition (A). This step turns out to be the main bottleneck in the polyhedral homotopy method for solving polynomial systems. We shall address this important issue in the next section.

In conclusion, we list the polyhedral homotopy procedure.

Polyhedral Homotopy Procedure

Given polynomial system $P(\mathbf{x}) = (p_1(\mathbf{x}),\ldots,p_n(\mathbf{x}))$ with support $S = (S_1,\ldots,S_n)$, let $S' = (S'_1,\ldots,S'_n)$ where $S'_j = S_j \cup \{\mathbf{0}\}$ for $j = 1,\ldots,n$.

Step 0. Initialization: Choose polynomial system $Q(\mathbf{x}) = (q_1(\mathbf{x}), \ldots, q_n(\mathbf{x}))$ having support $S' = (S'_1, \ldots, S'_n)$ and generically chosen coefficients. Write

$$q_j(\mathbf{x}) = \sum_{\mathbf{a} \in S'_j} c_{j,\mathbf{a}} x^{\mathbf{a}}, \quad j = 1, \ldots, n.$$

Step 1. Solve: $Q(\mathbf{x}) = \mathbf{0}$

Step 1.1: Construct the functions $\omega_j : S'_j \to \mathbb{Q}$, $j = 1, \ldots, n$, with generic images.

Step 1.2: Find all the cells $C^{\boldsymbol{\alpha}} = (\{\mathbf{a}_1, \mathbf{a}'_1\}, \ldots, \{\mathbf{a}_n, \mathbf{a}'_n\})$ of type $(1, \ldots, 1)$ with $\boldsymbol{\alpha} \in \mathbb{Q}^n$ and $\{\mathbf{a}_j, \mathbf{a}'_j\} \subset S'_j$, $j = 1, \ldots, n$, in the fine mixed subdivision $S_{\boldsymbol{\omega}}$ of $S' = (S'_1, \ldots, S'_n)$ induced by $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_n)$. (The algorithm for this step will be described in detail in §8)

Step 1.3: For each $\boldsymbol{\alpha} \in \mathbb{Q}^n$ and its associated cell $C^{\boldsymbol{\alpha}}$ obtained in Step 1.2.

Step 1.3.1: Solve the binomial system

$$c_{j,\mathbf{a}_j} \mathbf{y}^{\mathbf{a}_j} + c_{j,\mathbf{a}'_j} \mathbf{y}^{\mathbf{a}'_j} = 0, \quad j = 1, \ldots, n$$

in $(C^*)^n$. Let the solution set be $X^*_{\boldsymbol{\alpha}}$.

Step 1.3.2: Follow homotopy paths $\mathbf{y}(t)$ of the homotopy equation $H^{\boldsymbol{\alpha}}(\mathbf{y}, t) = (h_1^{\boldsymbol{\alpha}}(\mathbf{y}, t), \ldots, h_n^{\boldsymbol{\alpha}}(\mathbf{y}, t)) = \mathbf{0}$ with

$$h_j^{\boldsymbol{\alpha}}(\mathbf{y}, t) = \sum_{\mathbf{a} \in S'_j} c_{j,\mathbf{a}} \mathbf{y}^{\mathbf{a}} t^{\langle \hat{\mathbf{a}}, \hat{\boldsymbol{\alpha}} \rangle - \beta_j}, \quad j = 1, \ldots, n,$$

where $\beta_j = \langle \hat{\mathbf{a}}_j, \hat{\boldsymbol{\alpha}} \rangle$, starting from the solutions in $X^*_{\boldsymbol{\alpha}}$. Collect all the points of $\mathbf{y}(1)$ as a subset of isolated zeros of $Q(\mathbf{x})$.

Step 2: Solve $P(\mathbf{x}) = \mathbf{0}$

Follow homotopy paths of the homotopy

$$H(\mathbf{x}, t) = (1 - t)cQ(\mathbf{x}) + tP(\mathbf{x}) = \mathbf{0} \quad \text{for generic} \quad c \in \mathbb{C}^*$$

starting from the solutions of $Q(\mathbf{x}) = \mathbf{0}$ obtained in Step 1 to reach all isolated solutions of $P(\mathbf{x}) = \mathbf{0}$ at $t = 1$.

Remark 7.2. As we can see in the above procedure, in order to find all isolated zeros of $P(\mathbf{x})$ in $\mathbb{C}^n$, there are $k = \mathcal{M}(S'_1, \ldots, S'_n)$ homotopy paths need to be followed in both Step 1.3 and Step 2, hence $2k$ paths in total. This work may be reduced in half by the following strategy:

For

$$p_j(\mathbf{x}) = \sum_{\mathbf{a}\in S'_j} \bar{c}_{j,\mathbf{a}}\mathbf{x}^{\mathbf{a}}, \quad j = 1, \ldots, n,$$

we select the coefficients $c_{j,\mathbf{a}}$'s of $q_j(\mathbf{x})$, $j = 1, \ldots, n$ at **Step 0** to be $\bar{c}_{j,\mathbf{a}} + \epsilon_{j,\mathbf{a}}$, where $\epsilon_{j,\mathbf{a}}$'s are generically chosen small numbers to ensure each $q_j(\mathbf{x})$ is in general position. And at Step 1.3.2, we follow homotopy paths of the homotopy equation $\bar{H}^{\boldsymbol{\alpha}}(\mathbf{y}, t) = (\bar{h}_1^{\boldsymbol{\alpha}}(\mathbf{y}, t), \ldots, \bar{h}_n^{\boldsymbol{\alpha}}(\mathbf{y}, t)) = \mathbf{0}$, where

$$\bar{h}_j^{\boldsymbol{\alpha}}(\mathbf{y}, t) = \sum_{\mathbf{a}\in S'_j} [\bar{c}_{j,\mathbf{a}} + (1-t)\epsilon_{j,\mathbf{a}}]\mathbf{y}^{\mathbf{a}} t^{\langle \hat{\mathbf{a}}, \hat{\boldsymbol{\alpha}}\rangle - \beta_j}, \quad j = 1, \ldots, n. \tag{7.22}$$

It can be shown that the starting system $\bar{H}^{\boldsymbol{\alpha}}(\mathbf{y}, 0) = \mathbf{0}$ of this homotopy retain the same binomial system as before which was solved at Step 1.3.1 (with different coefficients of course). Most importantly, since $\bar{H}^{\boldsymbol{\alpha}}(\mathbf{y}, 1) = \bar{H}^{\boldsymbol{\alpha}}(\mathbf{x}, 1) = P(\mathbf{x})$, Step 2 in the above procedure is no longer necessary and we only need to follow k paths.

However, in following the solution paths of $\bar{H}^{\boldsymbol{\alpha}}(\mathbf{y}, t) = \mathbf{0}$ by the prediction-correction method, the first step of the predictor at $t = 0$ cannot be taken if a power of t in $\bar{H}^{\boldsymbol{\alpha}}(\mathbf{y}, t)$ is less than one, since $\bar{H}_t^{\boldsymbol{\alpha}}(\mathbf{y}, t)$ would then be undefined at $t = 0$. If the minimum power of t in (7.22) is, say, $t^{0.01}$ then changing variables with $T = t^{0.01}$ would solve the immediate problem. But it would replace numerical stability and computational efficiency if large powers of t, such as $t^{1,000}$, were also contained in $\bar{H}^{\boldsymbol{\alpha}}(\mathbf{y}, t)$. Then the tangent vector $\dot{\mathbf{y}} = (\bar{H}_{\mathbf{y}}^{\boldsymbol{\alpha}})^{-1} * \bar{H}_t^{\boldsymbol{\alpha}}$ would contain the terms in the order of $100,000 * t^{99,999}$ which, if evaluated at any $t \in [0, 1)$, would give 0. Close to 1, however, the tangent vector would become extremely steep, and step sizes for following the homotopy path must be correspondingly minuscule. Actually, while these sorts of problems already exist when "the polyhedral step" and "the linear step" are split as implemented earlier, they become multiply amplified when the combined polyhedral-linear homotopy is used. Ironically, notwithstanding the number of paths needed to be followed was cut in half by combining the polyhedral and linear steps, the difference between the computing times of the two approaches is almost negligible most of the time.

This problem was successfully addressed in [53] by applying the transformation $s = \ln t$ in (7.22), resulting in the homotopy $\bar{H}^{\boldsymbol{\alpha}}(\mathbf{y}, s) = (\bar{h}_1^{\boldsymbol{\alpha}}(\mathbf{y}, s), \ldots,$

$\bar{h}_n^{\boldsymbol{\alpha}}(\mathbf{y}, s)) = \mathbf{0}$, $s \in (-\infty, 0]$ where

$$\bar{h}_j^{\boldsymbol{\alpha}}(\mathbf{y}, s) = \sum_{\mathbf{a} \in S_j'} [\bar{c}_{j,\mathbf{a}} + (1 - e^s)\epsilon_{j,\mathbf{a}}]\mathbf{y}^{\mathbf{a}} e^{s*(\langle \hat{\mathbf{a}}, \hat{\boldsymbol{\alpha}} \rangle - \beta_j)}, \quad j = 1, \ldots, n. \tag{7.23}$$

We now need to follow the solution paths of $\bar{H}^{\boldsymbol{\alpha}}(\mathbf{y}, s) = \mathbf{0}$ from $s = -\infty$ to 0. For this purpose, a crucial observation here is, by a simple computation, one can show that for a solution path $\mathbf{y}(s)$ of $\bar{H}^{\boldsymbol{\alpha}}(\mathbf{y}, s) = \mathbf{0}$

$$\lim_{s \to -\infty} \frac{d\mathbf{y}}{ds} = \mathbf{0}.$$

Hence, the values of $\mathbf{y}(s)$ stay close to constant for negative s of large magnitude. Please see [53] for details.

Combining linear and nonlinear homotopies to reduce the number of solution paths needed to be followed in the polyhedral homotopy method by half was originally suggested back in [58]. However this idea was not successfully implemented earlier because of the involved numerical stability and efficiency problems. Addressing those difficulties by the transformation $s = \ln t$ and parameterizing the solution path by $s \in (-\infty, 0]$ [53], a substantial improvement in algorithmic efficiency and stability has been achieved as evidenced by the results of intensive numerical experiments. This combination strategy is particularly important when the polyhedral homotopies are used to solve large problems where mixed volumes of the systems are more than millions.

8. Mixed cell enumeration algorithm

Having discussed in the last section, a key step in the polyhedral homotopy for solving polynomial system $P(\mathbf{x}) = (p_1(\mathbf{x}), \ldots, p_n(\mathbf{x}))$ in $\mathbb{C}^n$ with support $S = (S_1, \ldots, S_n)$ is the identification of all the vectors $\hat{\boldsymbol{\alpha}} = (\boldsymbol{\alpha}, 1) \in \mathbb{Q}^{n+1}$ as well as their associate pairs $(\{\mathbf{a}_1, \mathbf{a}_1'\}, \ldots, \{\mathbf{a}_n, \mathbf{a}_n'\})$ that satisfy condition (A). For simplicity, we now assume all $p_j's$ have constant terms, namely $S_j = S_j \cup \{\mathbf{0}\}$ for $j = 1, \ldots, n$:

As mentioned before, those pairs $\{\mathbf{a}_1, \mathbf{a}_1'\}, \ldots, \{\mathbf{a}_n, \mathbf{a}_n'\}$ with $\hat{\boldsymbol{\alpha}} = (\boldsymbol{\alpha}, 1) \in \mathbb{Q}^{n+1}$ in condition (A), denoted by $C^{\boldsymbol{\alpha}} = (C_1, \ldots, C_n)$ with $C_j = \{\mathbf{a}_j, \mathbf{a}_j'\} \subset S_j$ for $j = 1, \ldots, n$, is actually a fine mixed cell of type $(1, \ldots, 1)$ in the subdivision $S_{\boldsymbol{\omega}}$ of $S = (S_1, \ldots, S_n)$ induced by the lifting $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_n)$. On the other hand, by (6.13), the volume of $C^{\boldsymbol{\alpha}}$ is

$$\mathrm{Vol}_n(C^{\boldsymbol{\alpha}}) = \left| \det \begin{bmatrix} \mathbf{a}_1'^{\top} - \mathbf{a}_1^{\top} \\ \vdots \\ \mathbf{a}_n'^{\top} - \mathbf{a}_n^{\top} \end{bmatrix} \right|.$$

Moreover, by Theorem 6.13, the mixed volume $\mathcal{M}(S_1, \ldots, S_n)$ of $S = (S_1, \ldots, S_n)$ is the sum of the volumes of all mixed cells $C^{\boldsymbol{\alpha}}$ of type $(1, \ldots, 1)$ in $S_{\boldsymbol{\omega}}$. Therefore, when all those mixed cells are available, the mixed volume $\mathcal{M}(S_1, \ldots, S_n)$ can be assembled with little extra computational effort. Here, let us reemphasize the critical role those pairs $\{\mathbf{a}_1, \mathbf{a}_1'\}, \ldots, \{\mathbf{a}_n, \mathbf{a}_n'\}$ play in the construction of the polyhedral homotopy as we have seen in the last section.

There is a substantial body of works devoted to the problem of enumerating all the mixed cells (e.g. [16, 18, 29, 30, 52, 59, 71, 77, 78, 106, 109]) In this section, we shall present an algorithm given in [59] for finding all those important mixed cells and their associated vectors $\hat{\boldsymbol{\alpha}} = (\boldsymbol{\alpha}, 1) \in \mathbb{Q}^{n+1}$.

In detail, a mixed cell of type $(1, \ldots, 1)$ in the subdivision $S_{\boldsymbol{\omega}}$ of $S = (S_1, \ldots, S_n)$ induced by the lifting $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_n)$ is an n-tuple $(\{\mathbf{a}_1^{(1)}, \mathbf{a}_2^{(1)}\}, \ldots, \{\mathbf{a}_1^{(n)}, \mathbf{a}_2^{(n)}\})$ with $\{\mathbf{a}_1^{(j)}, \mathbf{a}_2^{(j)}\} \subset S_j$, $j = 1, \ldots, n$, for which there exists a vector $\hat{\boldsymbol{\alpha}} = (\boldsymbol{\alpha}, 1) \in \mathbb{Q}^{n+1}$ such that

$$\text{(8.1)} \qquad \begin{cases} \langle \hat{\mathbf{a}}_1^{(1)}, \hat{\boldsymbol{\alpha}} \rangle = \langle \hat{\mathbf{a}}_2^{(1)}, \hat{\boldsymbol{\alpha}} \rangle < \langle \hat{\mathbf{a}}, \hat{\boldsymbol{\alpha}} \rangle & \text{for all } \mathbf{a} \in S_1 \backslash \{\mathbf{a}_1^{(1)}, \mathbf{a}_2^{(1)}\} \\ \qquad\qquad \vdots & \qquad\qquad \vdots \\ \langle \hat{\mathbf{a}}_1^{(n)}, \hat{\boldsymbol{\alpha}} \rangle = \langle \hat{\mathbf{a}}_2^{(n)}, \hat{\boldsymbol{\alpha}} \rangle < \langle \hat{\mathbf{a}}, \hat{\boldsymbol{\alpha}} \rangle & \text{for all } \mathbf{a} \in S_n \backslash \{\mathbf{a}_1^{(n)}, \mathbf{a}_2^{(n)}\} \end{cases}$$

where for each $\mathbf{a} \in S_j$, $\hat{\mathbf{a}} = (\mathbf{a}, \omega_j(\mathbf{a})) \in \hat{S}_j$ is its "lifted" version in $\mathbb{R}^{n+1}$.

One of the most efficient class of algorithms for enumerating mixed cells is developed from the idea of systematic "extension of subfaces". In the following, the concept of "subfaces" and their extensions will be defined first. A computational procedure, known as the "one-point test", for testing the possibilities of subface extensions will then be elaborated in detail in §8.2. Using one-point test as the basic building block, a mixed cell enumeration algorithm can then be constructed. At this stage, the algorithm is quite straightforward, and is far from computational efficiency. Four crucially important techniques provided in §8.3, §8.4, §8.5, §8.6 substantially accelerate the process. §9 generalizes the algorithm to handle the "semi-mixed" cases.

8.1. Enumeration via extensions of subfaces

Instead of finding n-tuples $(\{\mathbf{a}_1^{(1)}, \mathbf{a}_2^{(1)}\}, \ldots, \{\mathbf{a}_1^{(n)}, \mathbf{a}_2^{(n)}\})$ with $\{\mathbf{a}_1^{(j)}, \mathbf{a}_2^{(j)}\} \subset S_j,\ j = 1, \ldots, n$, to satisfy (8.1) directly, the scheme of our enumeration constructs the mixed cells by adding one point at a time. The idea is, one first focuses on the first lifted support $\hat{S}_1$ and locates all its lower vertices, that is, collecting every $\mathbf{a}_1^{(1)} \in S_1$ for which there exists an $\boldsymbol{\alpha} \in \mathbb{Q}^n$ such that

$$\langle \hat{\mathbf{a}}_1^{(1)}, \hat{\boldsymbol{\alpha}} \rangle \leq \langle \hat{\mathbf{a}}, \hat{\boldsymbol{\alpha}} \rangle \quad \text{for all } \ \mathbf{a} \in S_1$$

where $\hat{\boldsymbol{\alpha}} = (\boldsymbol{\alpha}, 1)$. Then for each of these points, one continues to search for $\mathbf{a}_2^{(1)} \in S_1 \setminus \{\mathbf{a}_1^{(1)}\}$ for which there exists an $\boldsymbol{\alpha} \in \mathbb{Q}^n$ such that

$$\langle \hat{\mathbf{a}}_1^{(1)}, \hat{\boldsymbol{\alpha}} \rangle = \langle \hat{\mathbf{a}}_2^{(1)}, \hat{\boldsymbol{\alpha}} \rangle < \langle \hat{\mathbf{a}}, \hat{\boldsymbol{\alpha}} \rangle \quad \text{for all } \ \mathbf{a} \in S_1 \setminus \{\mathbf{a}_1^{(1)}, \mathbf{a}_2^{(1)}\} \quad .$$

This "extends" a lower vertex of $\hat{S}_1$ to a lower edge of $\hat{S}_1$. To proceed, each of these possibilities of lower edges of $\hat{S}_1$, in turn, allows for the search of an additional point $\mathbf{a}_1^{(2)} \in S_2$ such that

$$\begin{aligned} &\langle \hat{\mathbf{a}}_1^{(1)}, \hat{\boldsymbol{\alpha}} \rangle = \langle \hat{\mathbf{a}}_2^{(1)}, \hat{\boldsymbol{\alpha}} \rangle < \langle \hat{\mathbf{a}}, \hat{\boldsymbol{\alpha}} \rangle \quad \text{for all } \ \mathbf{a} \in S_1 \setminus \{\mathbf{a}_1^{(1)}, \mathbf{a}_2^{(1)}\} \\ &\langle \hat{\mathbf{a}}_1^{(2)}, \hat{\boldsymbol{\alpha}} \rangle \leq \langle \hat{\mathbf{a}}, \hat{\boldsymbol{\alpha}} \rangle \qquad\qquad\quad \text{for all } \ \mathbf{a} \in S_2. \end{aligned} \quad .$$

This search, again, corresponds to the geometric action of pairing a lower edge of $\hat{S}_1$ and a lower vertex of $\hat{S}_2$ so that the two may share a common upward pointing inner normal vector in $\mathbb{R}^{n+1}$. Similarly, for each of the resulting positive possibilities further search attempts will be carried out to extend them to a pair of lower edges of $\hat{S}_1$ and $\hat{S}_2$, and they share a common upward pointing inner normal vector in $\mathbb{R}^{n+1}$. This self-sustaining process can proceed until one reaches all the possible n-tuples of lower edges of $\hat{S}_1, \ldots, \hat{S}_n$ that share a common upward pointing inner normal vector in $\mathbb{R}^{n+1}$ which are exactly the lifted set of all mixed cells of type $(1, \ldots, 1)$.

To formalize the above description, we first define, for $k \geq 0$, a k-dimensional lower face of conv $\hat{S}_1$, or simply a lower k-face, to be an affinely independent set of $k+1$ points $\{\hat{\mathbf{a}}_0^{(1)}, \ldots, \hat{\mathbf{a}}_k^{(1)}\}$ in $\hat{S}_1$ for which there exists an $\hat{\boldsymbol{\alpha}} = (\boldsymbol{\alpha}, 1) \in \mathbb{Q}^{n+1}$ such that

$$\begin{cases} \langle \hat{\mathbf{a}}_0^{(1)}, \hat{\boldsymbol{\alpha}} \rangle = \langle \hat{\mathbf{a}}_j^{(1)}, \hat{\boldsymbol{\alpha}} \rangle & \text{for } j = 1, \ldots, k \\ \langle \hat{\mathbf{a}}_0^{(1)}, \hat{\boldsymbol{\alpha}} \rangle \leq \langle \hat{\mathbf{a}}, \hat{\boldsymbol{\alpha}} \rangle & \text{for all } \ \mathbf{a} \in S_1 \setminus \{\mathbf{a}_0^{(1)}, \ldots, \mathbf{a}_k^{(1)}\}. \end{cases}$$

Extending this notion to two supports, we define a (k_1, k_2)-subface of $(\hat{S}_1, \hat{S}_2)$ (a special case of the more general Definition 8.1 given below) to be a pair of affinely independent subsets $(\{\hat{\mathbf{a}}_0^{(1)}, \ldots, \hat{\mathbf{a}}_{k_1}^{(1)}\}, \{\hat{\mathbf{a}}_0^{(2)}, \ldots, \hat{\mathbf{a}}_{k_2}^{(2)}\})$ of $\hat{S}_1$ and $\hat{S}_2$ respectively for which there exists an $\hat{\boldsymbol{\alpha}} = (\boldsymbol{\alpha}, 1) \in \mathbb{Q}^{n+1}$ such that

$$\begin{cases} \langle \hat{\mathbf{a}}_0^{(1)}, \hat{\boldsymbol{\alpha}} \rangle = \langle \hat{\mathbf{a}}_j^{(1)}, \hat{\boldsymbol{\alpha}} \rangle & \text{for } j = 1, \ldots, k_1 \\ \langle \hat{\mathbf{a}}_0^{(1)}, \hat{\boldsymbol{\alpha}} \rangle \leq \langle \hat{\mathbf{a}}, \hat{\boldsymbol{\alpha}} \rangle & \text{for all } \mathbf{a} \in S_1 \setminus \{\mathbf{a}_0^{(1)}, \ldots, \mathbf{a}_{k_1}^{(1)}\}. \\ \langle \hat{\mathbf{a}}_0^{(2)}, \hat{\boldsymbol{\alpha}} \rangle = \langle \hat{\mathbf{a}}_j^{(2)}, \hat{\boldsymbol{\alpha}} \rangle & \text{for } j = 1, \ldots, k_2 \\ \langle \hat{\mathbf{a}}_0^{(2)}, \hat{\boldsymbol{\alpha}} \rangle \leq \langle \hat{\mathbf{a}}, \hat{\boldsymbol{\alpha}} \rangle & \text{for all } \mathbf{a} \in S_2 \setminus \{\mathbf{a}_0^{(2)}, \ldots, \mathbf{a}_{k_2}^{(2)}\}. \end{cases}$$

Such a (k_1, k_2)-subface actually identifies a pair of k_1-dimension and k_2-dimension lower faces of conv $\hat{S}_1$ and conv $\hat{S}_2$ respectively that share a common upward pointing inner normal in $\mathbb{R}^{n+1}$.

More generally, we may define a $(k_1, \ldots, k_r)$-subface of $(\hat{S}_1, \ldots, \hat{S}_r)$, for some $r \leq n$, in a similar fashion:

Definition 8.1. A $(k_1, \ldots, k_r)$-subface of $(\hat{S}_1, \ldots, \hat{S}_r)$ is a r-tuple of affinely independent sets of the form

$$\left(\left\{ \hat{\mathbf{a}}_0^{(1)}, \ldots, \hat{\mathbf{a}}_{k_1}^{(1)} \right\} , \ \ldots \ , \left\{ \hat{\mathbf{a}}_0^{(r)}, \ldots, \hat{\mathbf{a}}_{k_r}^{(r)} \right\} \right)$$

with each $\hat{\mathbf{a}}_j^{(i)} \in \hat{S}_i$ for which there exists an $\hat{\boldsymbol{\alpha}} = (\boldsymbol{\alpha}, 1) \in \mathbb{Q}^{n+1}$ such that for each $i = 1, \ldots, r$,

$$\begin{cases} \langle \hat{\mathbf{a}}_0^{(i)}, \hat{\boldsymbol{\alpha}} \rangle = \langle \hat{\mathbf{a}}_j^{(i)}, \hat{\boldsymbol{\alpha}} \rangle & \text{for } j = 1, \ldots, k_i \\ \langle \hat{\mathbf{a}}_0^{(i)}, \hat{\boldsymbol{\alpha}} \rangle \leq \langle \hat{\mathbf{a}}, \hat{\boldsymbol{\alpha}} \rangle & \text{for all } \mathbf{a} \in S_i \setminus \{\mathbf{a}_0^{(i)}, \ldots, \mathbf{a}_{k_i}^{(i)}\} \end{cases} \tag{8.2}$$

(where the equality only appears if $k_i \geq 1$). Furthermore, we say a subface **extends** another subface if the former one can be obtained by joining the latter with a single point.

Remark 8.2. In this section, we focus on searching for mixed cells of type $(1, \ldots, 1)$. Thus, $k_i = 1$ for all $i = 1, \ldots, n$ in the above definition. However, for later use, $k_i's$ are allowed to be 0 or bigger then 1.

Interpreted geometrically, a $(k_1, \ldots, k_r)$-subface of $(\hat{S}_1, \ldots, \hat{S}_r)$ consists of r lower faces of conv $\hat{S}_1, \ldots,$ conv $\hat{S}_r$ having dimensions $k_1, \ldots, k_r$ respectively which share a common upward pointing inner normal. In particular, a (k)-subface of $\hat{S}_1$ defines a lower k-face of conv $\hat{S}_1$.

An observation of great importance is that since each point in a subface corresponds to an additional equality constraint in (8.2) algebraically, a subset of a subface must also be a subface. After all, a solution $\hat{\boldsymbol{\alpha}}$ of (8.2) would still be a solution if certain equality constraints were removed. We state the essence of this observation as a proposition.

Proposition 8.3. *Given a subface $(F_1, \ldots, F_r)$ of $(\hat{S}_1, \ldots, \hat{S}_r)$, any k-tuple $(F'_1, \ldots, F'_k)$ of sets such that $k \leq r$ and $F'_1 \subseteq F_1, \ldots, F'_k \subseteq F_k$ is a subface of $(\hat{S}_1, \ldots, \hat{S}_k)$.*

With the notion of subfaces, the basic scheme for searching mixed cells can then be stated as the systematic extension of (0)-subfaces to (1)-subfaces and then to $(1, 0)$-subfaces, etc. The process terminates when one reaches all the $(1, \ldots, 1)$-subfaces of $(\hat{S}_1, \ldots, \hat{S}_n)$ which are precisely the mixed cells of type $(1, \ldots, 1)$ in lifted form.

8.2. Extension of subfaces via one point test

In this subsection, we will elaborate a technique, known as the "one-point test", for testing the possibilities of subface extensions. It is the basic tool for our mixed cell enumeration algorithm. Originally developed in [59], this simple procedure has since been adopted by most software packages for mixed cell enumerations.

First, a simple transformation of (8.2) will greatly simplify the following discussions: Consider the system of inequalities

$$\begin{aligned} \langle \hat{\boldsymbol{a}}_0, \hat{\boldsymbol{\alpha}} \rangle &= \langle \hat{\boldsymbol{a}}_1, \hat{\boldsymbol{\alpha}} \rangle \\ \langle \hat{\boldsymbol{a}}_0, \hat{\boldsymbol{\alpha}} \rangle &\leq \langle \hat{\boldsymbol{a}}_2, \hat{\boldsymbol{\alpha}} \rangle . \end{aligned}$$

By introducing a new variable h to represent the common values of $\langle \hat{\boldsymbol{a}}_0, \hat{\boldsymbol{\alpha}} \rangle$ and $\langle \hat{\boldsymbol{a}}_1, \hat{\boldsymbol{\alpha}} \rangle$, then the above system becomes

$$\begin{array}{lll} h = \langle \hat{\boldsymbol{a}}_0, \hat{\boldsymbol{\alpha}} \rangle = \langle \mathbf{a}_0, \boldsymbol{\alpha} \rangle + \omega(\mathbf{a}_0) & & \langle -\mathbf{a}_0, \boldsymbol{\alpha} \rangle + h = \omega(\mathbf{a}_0) \\ h = \langle \hat{\boldsymbol{a}}_1, \hat{\boldsymbol{\alpha}} \rangle = \langle \mathbf{a}_1, \boldsymbol{\alpha} \rangle + \omega(\mathbf{a}_1) & \text{or} & \langle -\mathbf{a}_1, \boldsymbol{\alpha} \rangle + h = \omega(\mathbf{a}_1) \\ h \leq \langle \hat{\boldsymbol{a}}_2, \hat{\boldsymbol{\alpha}} \rangle = \langle \mathbf{a}_2, \boldsymbol{\alpha} \rangle + \omega(\mathbf{a}_2) & & \langle -\mathbf{a}_2, \boldsymbol{\alpha} \rangle + h \leq \omega(\mathbf{a}_2) \end{array}$$

in which the new unknowns $(\boldsymbol{\alpha}, h) = (\alpha_1, \ldots, \alpha_n, h)$ appear only on the left-hand-side. This is a more preferred form in discussing systems of inequalities. With a similar transformation, (8.2) can also be put into such forms: For affinely independent sets $F_1 = \{\hat{\mathbf{a}}_0^{(1)}, \ldots, \hat{\mathbf{a}}_{k_1}^{(1)}\}, \ldots, F_r = \{\hat{\mathbf{a}}_0^{(r)}, \ldots, \hat{\mathbf{a}}_{k_r}^{(r)}\}$

where $\hat{\mathbf{a}}_j^{(i)} \in \hat{S}_i$ for each $j = 0, \ldots, k_i$, we associate a system of inequalities in the variables $(\alpha_1, \ldots, \alpha_n, h_1, \ldots, h_r) \in \mathbb{Q}^{n+r}$:

$$
(8.3) \qquad I(F_1, \ldots, F_r) \; : \; \begin{cases} \left\langle -\mathbf{a}_j^{(1)}, \boldsymbol{\alpha} \right\rangle + h_1 = \omega_1(\mathbf{a}_j^{(1)}) & \text{for } j = 0, \ldots, k_1 \\ \left\langle -\mathbf{a}^{(1)}, \boldsymbol{\alpha} \right\rangle + h_1 \le \omega_1(\mathbf{a}^{(1)}) & \text{for all } \mathbf{a}^{(1)} \in S_1 \\ \qquad\qquad \vdots & \qquad\qquad \vdots \\ \left\langle -\mathbf{a}_j^{(r)}, \boldsymbol{\alpha} \right\rangle + h_r = \omega_r(\mathbf{a}_j^{(r)}) & \text{for } j = 0, \ldots, k_r \\ \left\langle -\mathbf{a}^{(r)}, \boldsymbol{\alpha} \right\rangle + h_r \le \omega_r(\mathbf{a}^{(r)}) & \text{for all } \mathbf{a}^{(r)} \in S_r. \end{cases}
$$

With $\hat{\boldsymbol{\alpha}} = (\boldsymbol{\alpha}, 1) = (\alpha_1, \ldots, \alpha_n, 1) \in \mathbb{R}^{n+1}$, this system is equivalent to (8.2), and therefore, $(F_1, \ldots, F_r)$ is a subface if and only if $I(F_1, \ldots, F_r)$ has a solution. Namely, $(F_1, \ldots, F_r)$ is a $(k_1, \ldots, k_r)$-subface of $(\hat{S}_1, \ldots, \hat{S}_r)$ is equivalent to the feasibility of the system $I(F_1, \ldots, F_r)$. Notice that this formulation makes it possible to consider cases where $F_j = \{\,\}$, an empty set, for some j, indicating there is no equality constraints in the corresponding blocks in (8.3).

With this setup, the possibility of extending a lower subface by a single point can then be determined by solving a linear programming problem known as a "one-point test".

Definition 8.4 (One-point test). Given affinely independent sets $F_1, \ldots, F_r$ with each $F_i = \{\hat{\mathbf{a}}_0^{(i)}, \ldots, \hat{\mathbf{a}}_{k_i}^{(i)}\} \subseteq \hat{S}_i$ (possibly empty) for which $I(F_1, \ldots, F_r)$ is feasible, and a point $\hat{\mathbf{b}} \in \hat{S}_r \setminus F_r$, the **one-point test** of $\mathbf{b}$ with respect to $(F_1, \ldots, F_r)$ is the *linear programming problem*

$$
(8.4) \qquad \begin{aligned} &\text{Maximize } \langle -\mathbf{b}, \boldsymbol{\alpha} \rangle + h_r \\ &\text{subject to } \begin{cases} \langle -\mathbf{a}_j^{(1)}, \boldsymbol{\alpha} \rangle + h_1 = \omega_1(\mathbf{a}_j^{(1)}) & \text{for } j = 0, \ldots, k_1 \\ \langle -\mathbf{a}^{(1)}, \boldsymbol{\alpha} \rangle + h_1 \le \omega_1(\mathbf{a}^{(1)}) & \text{for all } \mathbf{a}^{(1)} \in S_1 \\ \qquad\qquad \vdots & \qquad\qquad \vdots \\ \langle -\mathbf{a}_j^{(r)}, \boldsymbol{\alpha} \rangle + h_r = \omega_r(\mathbf{a}_j^{(r)}) & \text{for } j = 0, \ldots, k_r \\ \langle -\mathbf{a}^{(r)}, \boldsymbol{\alpha} \rangle + h_r \le \omega_r(\mathbf{a}^{(r)}) & \text{for all } \mathbf{a}^{(r)} \in S_r \end{cases} \end{aligned}
$$

which will be denoted by $LP(F_1, \ldots, F_r\,; \hat{\mathbf{b}})$.

Clearly, if the maximum of this problem agrees with $\omega_r(\mathbf{b})$, then the equality

$$\langle -\mathbf{b}, \boldsymbol{\alpha} \rangle + h_r = \omega_r(\mathbf{b})$$

is valid in addition to the above constraints. Accordingly, the $(k_1, \ldots, k_r)$-subface $(F_1, \ldots, F_r)$ can be extended to $(k_1, \ldots, k_r+1)$-subface $(F_1, \ldots, F_r \cup \{\hat{\mathbf{b}}\})$. On the other hand, if the maximum is strictly less than $\omega_r(\mathbf{b})$, then $I(F_1, \ldots, F_r \cup \{\hat{\mathbf{b}}\})$ is infeasible, hence $(F_1, \ldots, F_r \cup \{\hat{\mathbf{b}}\})$ fails to extend $(F_1, \ldots, F_r)$.

Remark 8.5. For the efficiency in actual implementations, the auxiliary variables $h_1, \ldots, h_r$ can be eliminated by substitution. For example, using the equality $\langle -\mathbf{a}_j^{(1)}, \boldsymbol{\alpha} \rangle + h_1 = \omega_1(\mathbf{a}_j^{(1)})$, or $h_1 = \omega_1(\mathbf{a}_j^{(1)}) + \langle \mathbf{a}_j^{(1)} \boldsymbol{\alpha} \rangle$, all appearances of the auxiliary variable h_1 in the rest of the system can be replaced by $\omega_1(\mathbf{a}_j^{(1)}) + \langle \mathbf{a}_j^{(1)} \boldsymbol{\alpha} \rangle$, and hence being eliminated.

To demonstrate the subface extension using one-point tests, let $(\{\hat{\mathbf{a}}_1, \hat{\mathbf{a}}_2\}, \{\hat{\mathbf{b}}_1\})$ be a subface of $(\hat{S}_1, \hat{S}_2)$ and $\hat{\mathbf{b}}_2 \in \hat{S}_2$, the one-point test of $\hat{\mathbf{b}}_2$ with respect to this subface is the linear programming (**LP**) problem $LP(\{\hat{\mathbf{a}}_1, \hat{\mathbf{a}}_2\}, \{\hat{\mathbf{b}}_1\}; \hat{\mathbf{b}}_2)$ in the unknowns $(\alpha_1, \ldots, \alpha_n, h_1, h_2)$ given by

$$\text{Maximize } \langle -\mathbf{b}_2, \boldsymbol{\alpha} \rangle + h_2$$

$$\text{subject to } \begin{cases} \langle -\mathbf{a}_1, \boldsymbol{\alpha} \rangle + h_1 = \omega_1(\mathbf{a}_1) \\ \langle -\mathbf{a}_2, \boldsymbol{\alpha} \rangle + h_1 = \omega_1(\mathbf{a}_2) \\ \langle -\mathbf{a}, \boldsymbol{\alpha} \rangle + h_1 \le \omega_1(\mathbf{a}) \quad \text{for all } \mathbf{a} \in S_1 \\ \langle -\mathbf{b}_1, \boldsymbol{\alpha} \rangle + h_2 = \omega_2(\mathbf{b}_1) \\ \langle -\mathbf{b}, \boldsymbol{\alpha} \rangle + h_2 \le \omega_2(\mathbf{b}) \quad \text{for all } \mathbf{b} \in S_2. \end{cases}$$

Note that the constraints of this LP problem restrict the objective function $\langle -\mathbf{b}_2, \boldsymbol{\alpha} \rangle + h_2$ to be bounded above by $\omega_2(\mathbf{b}_2)$. When the maximum reaches this upper bound, then $I(\{\hat{\mathbf{a}}_1, \hat{\mathbf{a}}_2\}, \{\hat{\mathbf{b}}_1, \hat{\mathbf{b}}_2\})$ is also feasible, and the subface $(\{\hat{\mathbf{a}}_1, \hat{\mathbf{a}}_2\}, \{\hat{\mathbf{b}}_1\})$ can be extended to $(\{\hat{\mathbf{a}}_1, \hat{\mathbf{a}}_2\}, \{\hat{\mathbf{b}}_1, \hat{\mathbf{b}}_2\})$. Similarly, additional one point tests can also be carried out to extend this newly obtained subface to other subfaces. Using such one-point test as the basic building tool, the algorithm for mixed cell enumeration via systematic extension of subfaces of $\hat{S}_1, \ldots, \hat{S}_n$ can be constructed. We summarize this basic algorithm here:

Step 1: Starting from supports $S_1, \ldots, S_n$, one assigns lifting functions $\omega_k : S_k \to \mathbb{Q}$ for $k = 1, \ldots, n$ with random images.

Step 2: A (0)-subface is obtained by solving the so-called *phase-one* problem (will be discussed in §8.3.2). This can be achieved by solving an augmented LP problem (8.8).

Step 3: Continue from the 0-subfaces, one proceeds to enumerate (1)-subfaces by solving one-point test problems of the form (8.4). The resulting (1)-subfaces, if any, are saved in a pool.

Step 4a: For each $(\overbrace{1,\ldots,1}^{r-1},1,0)$-subface in the pool involving points in $S_1,\ldots,S_r$, further one-point tests are attempted using points in S_r to extend it to an $(\overbrace{1,\ldots,1,1}^{r})$-subface. If successful, the resulting subfaces are saved in the pool. If $r=n$, then the new subfaces are mixed cells.

Step 4b: For each $(\overbrace{1,\ldots,1}^{r})$-subface in the pool with $r<n$, further one-point tests are attempted using points from the next support to extend it to a $(\overbrace{1,\ldots,1}^{r},0)$-subface. If successful, the resulting subfaces are also saved in the pool.

Step 4a and **4b** are repeated until all subfaces in the pool have been extended, at this stage all mixed cells have been located.

However, algorithms employing one-point tests for subface extensions in a straightforward manner may become quite inefficient. At the first sight, there are so many one-point tests need to be carried out, especially when the system is large. Fortunately, by the development of more advanced techniques, not all one-point test problems need to be solved in the subface extension algorithms. Those will be the subjects of the following subsections. All these techniques have been proven to be indispensable in efficient implementation of our mixed cell enumeration algorithms.

8.3. Accelerated extension via simplex method

As mentioned above, for a subface $(F_1,\ldots,F_r)$ with $F_i=\{\hat{\mathbf{a}}_0^{(i)},\ldots,\hat{\mathbf{a}}_{k_i}^{(i)}\}\subseteq \hat{S}_i$ and $\hat{\mathbf{b}}\in\hat{S}_r\setminus F_r$ the one-point test of $\hat{\mathbf{b}}$ with respect to $(F_1,\ldots,F_r)$ is a linear programming problem $LP(F_1,\ldots,F_r\,;\,\hat{\mathbf{b}})$ (8.4) of the form

$$\text{Maximize } \langle -\mathbf{b}, \boldsymbol{\alpha} \rangle + h_r$$

$$\text{subject to } \begin{cases} \langle -\mathbf{a}_j^{(1)}, \boldsymbol{\alpha} \rangle + h_1 = \omega_1(\mathbf{a}_j^{(1)}) & \text{for } j = 0, \ldots, k_1 \\ \langle -\mathbf{a}^{(1)}, \boldsymbol{\alpha} \rangle + h_1 \le \omega_1(\mathbf{a}^{(1)}) & \text{for all } \mathbf{a}^{(1)} \in S_1 \\ \qquad\qquad \vdots & \qquad\qquad \vdots \\ \langle -\mathbf{a}_j^{(r)}, \boldsymbol{\alpha} \rangle + h_r = \omega_r(\mathbf{a}_j^{(r)}) & \text{for } j = 0, \ldots, k_r \\ \langle -\mathbf{a}^{(r)}, \boldsymbol{\alpha} \rangle + h_r \le \omega_r(\mathbf{a}^{(r)}) & \text{for all } \mathbf{a}^{(r)} \in S_r. \end{cases}$$

While such linear programming problems can be solved by various methods, in our cases the classical *simplex method* offers a great advantage because the rich information generated by "pivoting process" in the simplex method enables the discovery of solutions to many other related one-point test problems without actually solving them.

8.3.1. Simplex method. The simplex method in the context of one-point test is briefly reviewed below, and the details can be found in standard texts, e.g., [10]. For simplicity, for some fixed k, we consider the linear programming problems of the form

$$\text{Maximize } \langle f, \mathbf{x} \rangle + f_0$$

$$\text{subject to } \begin{cases} \langle \mathbf{c}_j, \mathbf{x} \rangle \le \alpha_j, & \text{for } j = 1, \ldots, m \\ \langle \mathbf{q}_j, \mathbf{x} \rangle = \gamma_j, & \text{for } j = 1, \ldots, \ell \end{cases} \tag{8.5}$$

in the variables $\mathbf{x} = (x_1, \ldots, x_k) \in \mathbb{Q}^k$ where $f \in \mathbb{Q}^k$ defines the *objective function* and $\{\mathbf{c}_j\}, \{\mathbf{q}_j\} \subset \mathbb{Q}^k$ are the *constraints*. Here we allow ℓ to be zero in which case there are no equality constraints.

In the context of one-point tests (8.4), the equality constraints $\mathbf{q}_1, \ldots, \mathbf{q}_\ell$ correspond to the points in the subfaces while the inequality constraints $\mathbf{c}_1, \ldots, \mathbf{c}_m$ correspond to the remaining points in the supports involved. The objective function corresponds to the point being tested. Note that since the objective function also appears as one of the inequality constraints, this problem, if feasible, must be bounded.

When the simplex method is used to solve the linear programming problem (8.5), it is customary, for historical as well as practical reasons, to convert

the problem into the "standard form"

$$\text{Minimize } \langle \mathbf{d}, \mathbf{y} \rangle$$
$$\text{subject to } \begin{cases} A\mathbf{y} = \mathbf{b} \\ \mathbf{y} \geq 0 \end{cases}$$

for some vector $\mathbf{d}$ and some $n \times (m + \ell)$ matrix A with $n < m + \ell$. However, for our needs, it is critically important to solve the problem in the form given in (8.5) directly as described in §8.3.2, §8.3.3 and §8.3.4 which utilize special features of this formulation.

The set R of all points $\mathbf{x} \in \mathbb{R}^k$ satisfying the constraints of (8.5) is called the *feasible region* of the LP problem, and the problem is said to be *infeasible* if its feasible region is empty. For a feasible LP problem, the simplex method is a particular way to move within the feasible region that will maximize the objective function: It jumps from one vertex to another vertex along edges in the direction that increases the objective function until the optimum is reached. See the illustration in Figure 14.

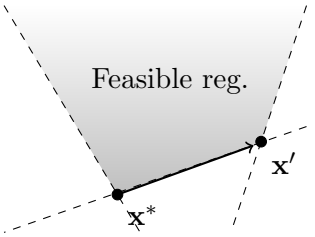


Figure 14. The simplex method jumps from one basic feasible solution to another basic feasible solution on the boundary of the feasible region.

We now translate this geometric procedure back to algebraic terms: at a feasible point $\mathbf{x} \in R$, a constraint in (8.5) is said to be *active* if the equality holds. A *basic feasible solution* of R is a point in R where exactly k constraints are active and these constraints are linearly independent. They are the vertices of R. The *basic matrix* at a basic feasible solution is the $k \times k$ matrix whose rows are the active constraints, therefore the basic matrix must be nonsingular. A basic feasible solution, i.e., a vertex, is a point where at least k edges meet. We shall see the directions of these k edges are given by the columns of the inverse of the basic matrix at this point.

Remark 8.6. In general, it is certainly possible for more than k edges to meet at a vertex. However, in the current context, it is easy to check

under the assumption that the lifting values are generic in the sense of Definition 6.14, exactly k edges meet at each vertex.

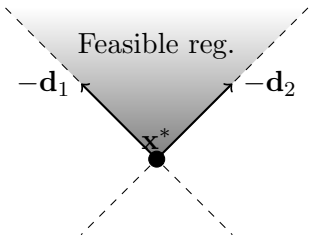


Figure 15. Columns $-\mathbf{d}_1$ and $-\mathbf{d}_2$ in $-D = -B^{-1}$ point to the direction of the edges leaving $\mathbf{x}^*$ while staying inside R.

For a basic feasible solution $\mathbf{x}^*$ of (8.5) let

$$B = \begin{bmatrix} — & \mathbf{b}_1 & — \\ & \vdots & \\ — & \mathbf{b}_k & — \end{bmatrix}$$

be the basic matrix at $\mathbf{x}^*$ where $\{\mathbf{b}_j\}_{j=1}^k \subset \{\mathbf{c}_j\}_{j=1}^m \cup \{\mathbf{q}_j\}_{j=1}^\ell$ are the active constraints (equalities in the system). Also, let β_j be the entry in $\{\alpha_j\}_{j=1}^m \cup \{\gamma_j\}_{j=1}^\ell$ corresponding to $\mathbf{b}_j$ (the right-hand-side of the corresponding constraint in (8.5)). Then $\mathbf{x}^*$ is the unique solution of the nonsingular linear system

$$\begin{aligned} \langle \mathbf{b}_1, \mathbf{x}^* \rangle &= \beta_1 \\ &\vdots \\ \langle \mathbf{b}_k, \mathbf{x}^* \rangle &= \beta_k. \end{aligned}$$

Let $D = [\mathbf{d}_1, \ldots, \mathbf{d}_k] = B^{-1}$, then by definition $\langle \mathbf{b}_i, \mathbf{d}_j \rangle = 0$ for any $i \neq j$. Fix any j in $\{1, \ldots, k\}$, and consider the ray $\mathbf{x}^* - t\mathbf{d}_j$ for $t > 0$ that originate at $\mathbf{x}^*$ and points to the direction of $-\mathbf{d}_j$. Clearly, for any $i \neq j$,

$$\langle \mathbf{b}_i \, , \, \mathbf{x}^* - t\mathbf{d}_j \rangle = \langle \mathbf{b}_i \, , \, \mathbf{x}^* \rangle - t \, \langle \mathbf{b}_i \, , \, \mathbf{d}_j \rangle = \langle \mathbf{b}_i \, , \, \mathbf{x}^* \rangle - t \cdot 0 = \beta_i.$$

That is, any point on this ray still satisfies all but the j-th equation in the above linear system. Moreover,

$$\langle \mathbf{b}_j \, , \, \mathbf{x}^* - t\mathbf{d}_j \rangle = \langle \mathbf{b}_j \, , \, \mathbf{x}^* \rangle - t \, \langle \mathbf{b}_j \, , \, \mathbf{d}_j \rangle = \beta_j - t.$$

That is, the j-th active constraint becomes inactive (but is still valid). Geometrically, this ray pointing to the $-\mathbf{d}_j$ direction leaves the j-th active constraint at $\mathbf{x}^*$ while maintaining the rest of the active constraints. See, for example, the two-dimensional illustration in Figure 15.

Clearly, to stay inside the feasible region, movement is only allowed in the direction $\mathbf{d}_j$ corresponding to an inequality constraint among $\mathbf{c}_1, \ldots, \mathbf{c}_m$. Along each of these directions $\mathbf{d}_j$, the objective function becomes

$$\langle f, \mathbf{x}^* - t\mathbf{d}_j \rangle + f_0 = \langle f, \mathbf{x}^* \rangle - t \langle f, \mathbf{d}_j \rangle + f_0$$

which we intend to maximize. It increases for $t > 0$ when $\langle f, \mathbf{d}_j \rangle < 0$. If $\langle f, \mathbf{d}_j \rangle \geq 0$ for all $\mathbf{d}_j$ corresponding to the inequality constraints among $\mathbf{c}_1, \ldots, \mathbf{c}_m$, then $\mathbf{x}^*$ is an optimal solution of (8.5). Otherwise, the simplex method moves along one of these directions which increase the objective function to an adjacent basic feasible solution. This maneuver is commonly known as *pivoting*. While having multiple directions along which the objective function could increase is possible, it is sufficient to choose the direction with the largest per unit increment with respect to the increment of t.

To reach the adjacent basic feasible solution, the step size along the ray $\mathbf{x}^* - t\mathbf{d}_j$ is chosen to be the minimum $t > 0$ for which another previously inactive constraint becomes active, namely,

$$\Delta t = \min \left\{ \frac{\langle \mathbf{c}_i, \mathbf{x}^* \rangle - \alpha_i}{\langle \mathbf{c}_i, -\mathbf{d}_j \rangle} \;\middle|\; \mathbf{c}_i \text{ is inactive and } \langle \mathbf{c}_i, \mathbf{d}_j \rangle < 0 \right\}. \tag{8.6}$$

This minimum step size always exists in the context of one-point tests (certainly not in general linear programming problems), since the objective functions in problems of the form (8.4) are bounded above. With this chosen step size, a new basic feasible solution $\mathbf{x}' := \mathbf{x}_j(\Delta t) = \mathbf{x}^* - \Delta t \mathbf{d}_j$ is produced with an increased value of the objective function, and this process can be repeated until an optimal solution is reached.

8.3.2. Solving the phase-one problem. In the above, we described the simplex method for solving linear programming problem of the form (8.4) by jumping from one basic feasible solution to another basic feasible solution until an optimum solution is reached. This part is known as the "phase two" of the simplex method. To bootstrap this process, one must determine the feasibility of the problem in the first place, and if it is indeed feasible, one must locate a basic feasible solution to start the phase two process. This bootstrapping step is generally referred to as the "phase one" of the simplex method. While, in general, solving the phase one problem can be

very costly [10], the phase one problems for one-point test problems are somewhat straightforward due to their special structure.

Consider first a one-point test of the form $LP(\{\}; \hat{\mathbf{b}})$ which is the starting point of the mixed cell enumeration process via extensions of subfaces. It is a linear programming problem

$$\begin{aligned} &\text{Maximize } \langle -\mathbf{b}, \boldsymbol{\alpha} \rangle + h_1 \\ &\text{subject to } \langle -\mathbf{a}, \boldsymbol{\alpha} \rangle + h_1 \leq \omega_1(\mathbf{a}) \quad \text{for all } \mathbf{a} \in S_1 \end{aligned} \tag{8.7}$$

for some $\mathbf{b} \in S_1$.

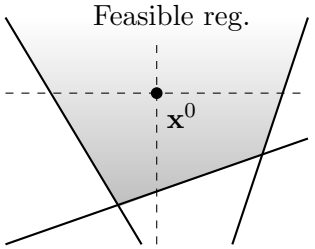


Figure 16. The initial solution with dashed lines representing artificial constraints.

This problem is always feasible, since by choosing an arbitrary $M \in \mathbb{Q}$ with $M < \omega_1(\mathbf{a})$ for all $\mathbf{a} \in S_1$, $(\boldsymbol{\alpha}, h_1) = (\mathbf{0}, M)$ is obviously a feasible point of this linear programming problem.

For simplicity, we assume the feasible region to be full-dimensional, since the method discussed here can be trivially extended to cases where the feasible region is contained in a affine space of lower dimension by simply restricting all the pivoting procedures to within that affine space.

To find a basic feasible solution to start the pivoting process, notice that the feasible solution $\mathbf{x}^0 = (\boldsymbol{\alpha}, h_1) = (\mathbf{0}, M)$ is not a basic feasible solution since none of the equalities hold. However, we may look at the following problem instead

$$\begin{aligned} &\text{Max } \langle -\mathbf{b}, \boldsymbol{\alpha} \rangle + h_1 \\ &\text{subject to } \begin{cases} \langle -\mathbf{a}, \boldsymbol{\alpha} \rangle + h_1 \leq \omega_1(\mathbf{a}) \text{ for } \mathbf{a} \in S_1 \\ \alpha_1 = 0 \\ \alpha_2 = 0 \\ \vdots \\ \alpha_n = 0 \\ h_1 = M. \end{cases} \end{aligned} \tag{8.8}$$

where the last $n+1$ constraints are artificially imposed. Clearly, $\mathbf{x}^0$ is a basic feasible solution of this augmented problem since exactly $n+1$ equalities hold. The pivoting process can now be used on (8.8) to remove the artificially imposed constraints one at a time until a basic feasible solution of (8.7) is reached. Therefore, the phase one problem can be solved by exactly $n+1$ pivots.

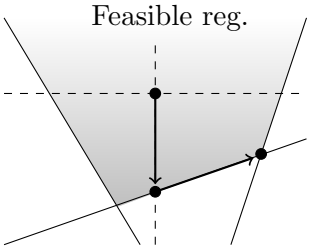


Figure 17. Solving the phase one problem by leaving the artificial constraints one at a time.

From the geometric standpoint, this process first moves the initial solution $\mathbf{x}^0$ from the interior of the feasible region to an n-dimensional (co-dimension 1) face on the boundary of the feasible region by leaving an artificial constraint, say $\alpha_1 = 0$, and then to an $(n-1)$-dimensional (co-dimension 2) face on the boundary by leaving another artificial constraint, say $\alpha_2 = 0$, etc. Since we have assumed the feasible region to be full-dimensional, eventually this process will reach a vertex on the boundary at which all $n+1$ artificially imposed constraints are removed. Figure 17 is a 2-dimensional illustration of this process.

For general one-point tests, when the maximum for $LP(F_1, \ldots, F_r;\ \hat{\mathbf{b}})$ agrees with $\omega_r(\mathbf{b})$, then the basic feasible solution that reaches the optimum of this problem is also a basic feasible solution of $LP(F_1, \ldots, F_r \cup \{\mathbf{b}\};\ \mathbf{b}')$ for any $\mathbf{b}' \in S_r$. In simple terms, the optimum solution of the one-point test that produces a new subface successfully is a basic feasible solution of all one-point tests with respect to this subface involving the same set of supports.

For one-point tests involving different supports, their phase one problem is still easy to solve. Let $(\boldsymbol{\alpha}^*, h_1^*, \ldots, h_r^*)$ be a basic feasible solution of $I(F_1, \ldots, F_r)$ resulted from other one-point test. There are exactly $n+r$ equalities hold in $I(F_1, \ldots, F_r)$ at this point. Let

$$h_{r+1}^* = \min \{\omega_{r+1}(\mathbf{b}) - \langle -\mathbf{b}, \boldsymbol{\alpha} \rangle \mid \mathbf{b} \in S_{r+1}\}$$

then one equality holds among

$$\langle -\mathbf{b}, \boldsymbol{\alpha}^* \rangle + h_{r+1}^* \leq \omega_{r+1}(\mathbf{b})$$

for all $\mathbf{b} \in S_{r+1}$. Therefore, the point $\mathbf{x}' = (\boldsymbol{\alpha}^*, h_1^*, \ldots, h_r^*, h_{r+1}^*)$ is a basic feasible solution of $LP(F_1, \ldots, F_r\ ;\ \hat{\mathbf{b}}_{r+1})$ for any $\mathbf{b}_{r+1} \in S_{r+1}$ since exactly $n + r + 1$ equalities hold at this point, matching the number of unknowns.

The combined cascade effects ensure that through extensions, a basic feasible solution for any one-point test can always be obtained easily from an optimum solution of an "upstream" one-point test in the extension process.

8.3.3. Harvesting information generated in simplex method. A big advantage in adopting the above simplex method is when the one-point test problem (8.4) is solved by this method, rich information generated by the pivoting process is particularly fruitful.

To illustrate this, we begin with the simplest case: Let the one-point test $LP(\{\}\,;\,\hat{\mathbf{a}}')$ on $\mathbf{a}' \in S_1$ (the starting points of the mixed cell enumeration procedure) be given by

$$\begin{aligned} &\text{Maximize } \langle -\mathbf{a}', \boldsymbol{\alpha} \rangle + h_1 \\ &\text{subject to } \langle -\mathbf{a}, \boldsymbol{\alpha} \rangle + h_1 \leq \omega_1(\mathbf{a}) \quad \text{for all } \ \mathbf{a} \in S_1. \end{aligned}$$

Let $\mathbf{x}^* = (\boldsymbol{\alpha}^*, h_1^*) \in \mathbb{R}^{n+1}$ be a basic feasible solution (not necessarily optimal) visited during the pivoting process in the simplex method (e.g. a vertex in Figure 14). By definition, there are exactly $n + 1$ equalities hold. Suppose $\mathbf{a}_0, \ldots, \mathbf{a}_n$ are the points in S_1 correspond to those equalities. Then $\mathbf{x}^* = (\boldsymbol{\alpha}^*, h_1^*)$ satisfies

$$\begin{aligned} \langle -\mathbf{a}_0, \boldsymbol{\alpha}^* \rangle + h_1^* &= \omega_1(\mathbf{a}_0) \\ &\vdots \\ \langle -\mathbf{a}_n, \boldsymbol{\alpha}^* \rangle + h_1^* &= \omega_1(\mathbf{a}_n) \\ \langle -\mathbf{a}, \boldsymbol{\alpha}^* \rangle + h_1^* &\leq \omega_1(\mathbf{a}) \quad \text{for all } \ \mathbf{a} \in S_1. \end{aligned}$$

Evidently, it also satisfies $I(\{\hat{\mathbf{a}}_0\})$ given by

$$\begin{aligned} \langle -\mathbf{a}_0, \boldsymbol{\alpha}^* \rangle + h_1^* &= \omega_1(\mathbf{a}_0) \\ \langle -\mathbf{a}, \boldsymbol{\alpha}^* \rangle + h_1^* &\leq \omega_1(\mathbf{a}) \quad \text{for all } \ \mathbf{a} \in S_1 \end{aligned}$$

since the feasible region of this system contains the feasible region of the previous system. Thus, without any additional computation, $I(\{\hat{\mathbf{a}}_0\})$ is feasible.

Similarly, $I(\{\hat{\mathbf{a}}_1\}), \ldots, I(\{\hat{\mathbf{a}}_n\})$ are all feasible. Furthermore, $\mathbf{x}^* = (\boldsymbol{\alpha}^*, h_1^*)$ satisfies

$$\begin{aligned} \langle -\mathbf{a}_0, \boldsymbol{\alpha}^* \rangle + h_1^* &= \omega_1(\mathbf{a}_0) \\ \langle -\mathbf{a}_1, \boldsymbol{\alpha}^* \rangle + h_1^* &= \omega_1(\mathbf{a}_1) \\ \langle -\mathbf{a}, \boldsymbol{\alpha}^* \rangle + h_1^* &\le \omega_1(\mathbf{a}) \quad \text{for all} \ \ \mathbf{a} \in S_1. \end{aligned}$$

Therefore $I(\{\hat{\mathbf{a}}_0, \hat{\mathbf{a}}_1\})$ is also feasible. Indeed, by the same reasoning, for any subset $F \subseteq \{\hat{\mathbf{a}}_0, \ldots, \hat{\mathbf{a}}_n\}$, $I(F)$ is always feasible and hence produces the subface F of $\hat{S}_1$. Thus the need for performing corresponding one-point tests is completely eliminated. It is important to note that all these eliminations are due virtually to the information generated by a single vertex.

In general, when the above simplex method is used to solve the one-point test $LP(F_1, \ldots, F_r; \hat{\mathbf{b}})$ for some $\mathbf{b} \in S_r$, every basic feasible solution visited in the pivoting process could potentially reveal the existence of a large number of subfaces: Suppose $\mathbf{x}^* = (\boldsymbol{\alpha}^*, h_1^*, \ldots, h_r^*)$ is a basic feasible solution of $LP(F_1, \ldots, F_r; \hat{\mathbf{b}})$ and $\mathbf{b}_1, \ldots, \mathbf{b}_m \in S_r$ are the points corresponds to the equalities that hold at $\mathbf{x}^*$ within the block for S_r. Then for any $F' \subseteq \{\hat{\mathbf{b}}_1, \ldots, \hat{\mathbf{b}}_m\} \setminus F_r$, the system $I(F_1, \ldots, F_r \cup F')$ must be feasible, indicating the existence of the subface $(F_1, \ldots, F_r \cup F')$.

Since much of this section is focusing on the algorithms for finding mixed cells of type $(1, \ldots, 1)$, therefore only subfaces of the form $(F_1, \ldots, \{\hat{\mathbf{b}}_j, \hat{\mathbf{b}}_k\})$ where $\mathbf{b}_j, \mathbf{b}_k \in \{\mathbf{b}_1, \ldots, \mathbf{b}_m\}$ would be of interest. However, the great value of utilizing combinations of the equalities at a basic feasible solution would become apparent in §9 where general types of mixed cells are needed.

8.3.4. Removal of extraneous constraints. In solving one-point test problems via the simplex method described above, a significant portion of computation lies in the step (8.6) which goes through each constraint to find the appropriate step size for moving from one basic feasible solution to another. Indeed, when the number of points in the supports is large, this step generally dictates the overall cost of this approach. An important resolution is the possibility of removing a substantial amount of constraints in (8.4), without affecting the final result of course.

If it is known *a priori* that certain constraints will never become active during the pivoting process, then the solution remains the same when those constraints are removed. In the mixed cell enumeration process, if a one-point test $LP(F_1, \ldots, F_r\,;\, \hat{\mathbf{b}})$ fails, then there are no points in the feasible region of $LP(F_1, \ldots, F_r\,;\, \hat{\mathbf{b}})$ which satisfy $\langle -\mathbf{b}, \boldsymbol{\alpha} \rangle + h_r = \omega_r(\mathbf{b})$. This can only occur when the entire feasible region lies in the interior of $\{\boldsymbol{\alpha}:$

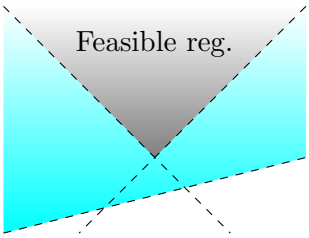


Figure 18. An extraneous constraint having no intersection with the feasible region.

$\langle -\mathbf{b}, \boldsymbol{\alpha} \rangle + h_r < \omega_r(\mathbf{b})\}$. See Figure 18 for a simple two-dimensional illustration. Consequently, the constraint $\langle -\mathbf{b}, \boldsymbol{\alpha} \rangle + h_r \leq \omega_r(\mathbf{b})$ cannot become active in any linear programming problems having the same feasible region and therefore can be removed from one-point tests $LP(F_1, \ldots, F_r\,;\, \hat{\mathbf{b}}')$ for any $\mathbf{b}'$. In other words, each failed one-point test $LP(F_1, \ldots, F_r\,;\hat{\mathbf{b}})$ (in the sense that it does not produce an extension) of $\hat{\mathbf{b}}$ with respect to the subface $(F_1, \ldots, F_r)$ can eliminate the constraint corresponding to $\mathbf{b}$ from all one-point tests with respect to the same subface. Furthermore, since the extension of subfaces essentially amounts to imposing additional constraints to the system of equalities (8.3) resulting in the restriction of their feasible regions, so the constraint $\langle -\mathbf{b}, \boldsymbol{\alpha} \rangle + h_r \leq \omega_r(\mathbf{b})$ can also be removed from one-point tests with respect to any subfaces containing $(F_1, \ldots, F_r)$.

The removal of extraneous constraints of this sort cumulatively yields a substantial reduction in the amount of computation. Other way of removing extraneous constraints is further exploited by the construction of the "relation table" to be discussed in §8.5.

Remark 8.7. Since the simplex method solves linear programming problems by visiting vertices of the feasible region, the removal of such extraneous constraints which can never be active has no effect on the important technique for harvesting information generated by the pivoting process described in §8.3.3.

8.4. Quick eliminations of extensions

As formulated in (8.3), in the systematic extensions of lower subfaces, the main event is the testing of the feasibilities of systems of inequalities of the

form

$$I\,(F_1,\ldots,F_r)\ :\ \begin{cases} \langle -\mathbf{a}_j^{(1)},\boldsymbol{\alpha}\rangle + h_1 = \omega_1(\mathbf{a}_j^{(1)}) & \text{for } j=1,\ldots,k_1\\ \langle -\mathbf{a}_j^{(1)},\boldsymbol{\alpha}\rangle + h_1 \le \omega_1(\mathbf{a}_j^{(1)}) & \text{for all } \mathbf{a}\in S_1\\ \qquad\vdots & \qquad\vdots\\ \langle -\mathbf{a}_j^{(r)},\boldsymbol{\alpha}\rangle + h_r = \omega_r(\mathbf{a}_j^{(r)}) & \text{for } j=1,\ldots,k_r\\ \langle -\mathbf{a}_j^{(r)},\boldsymbol{\alpha}\rangle + h_r \le \omega_r(\mathbf{a}_j^{(r)}) & \text{for all } \mathbf{a}\in S_r \end{cases}$$

in the unknowns $(\boldsymbol{\alpha}, h_1, h_2, \ldots, h_r) \in \mathbb{Q}^{n+r}$. Recall that after a (1)-subface F_1 of $(\hat{S}_1)$ is obtained, the extension process continues as systematic attempts of extending a solution of $I(F_1)$ to a solution of $I(F_1, \{\hat{\mathbf{b}}\})$ for some $\mathbf{b} \in S_2$. If it is successful, further attempts are made to extend the solution to $I(F_1, \{\hat{\mathbf{b}}, \hat{\mathbf{b}}'\})$. This self-sustaining process continues until all the mixed cells are obtained.

In the context of the above system of inequalities, each extension step is the inclusion of an additional group of constraints. An important observation is that certain extensions that will fail can sometimes be detected without executing the corresponding one-point tests, and hence avoiding a great deal of computations. This section gives a geometric interpretation of this observation. Originally introduced in [78], it has inspired several variations which are adopted by DECiMs[77], MixedVol-2.0[52], and MixedVol-3.0[16].

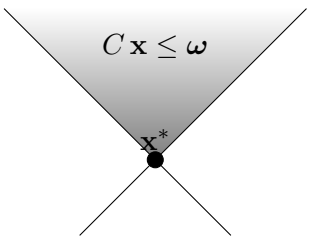
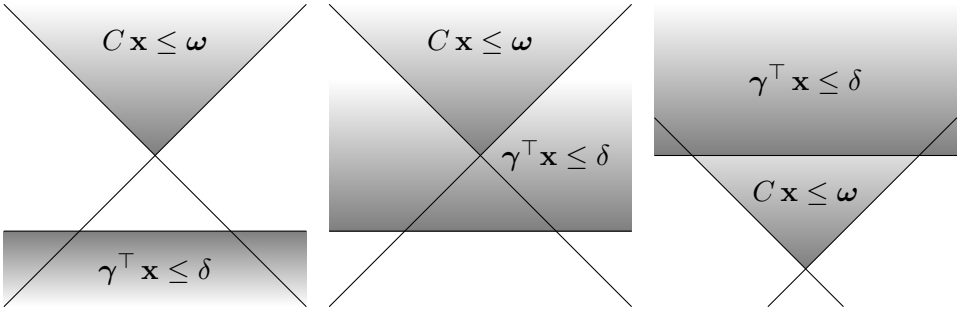


Figure 19. The feasible region of the original system of inequalities $C\,\mathbf{x} \le \mathbf{b}$ together with a feasible solution for which two equalities hold.

8.4.1. The effect of adding a single constraint. We first look at a simplified scenario where a single new constraint is added to a system of inequalities known to be feasible. That is, for a system of inequalities $C\,\mathbf{x} \le \boldsymbol{\omega}$ in the variables $\mathbf{x} = (x_1, \ldots, x_m)$ for some fixed integer $m > 1$ having a



(a) (*Infeasible*) With the new constraint, the system becomes infeasible.

(b) (*Feasible & inactive*) With the new constraint, the system is still feasible but the new constraint can never be active.

(c) (*Feasible & active*) With the new constraint, the system is still feasible, and the new constraint may become active.

Figure 20. Three possible ways an additional constraint can change the feasibility of a system of inequalities that is known to be feasible.

vertex $\mathbf{x}^*$ in its feasible region (solution set), we shall investigate the possible solution to the extended system

$$\begin{bmatrix} C \\ \boldsymbol{\gamma}^\top \end{bmatrix} \mathbf{x} \leq \begin{bmatrix} \boldsymbol{\omega} \\ \delta \end{bmatrix} \tag{8.9}$$

where $\boldsymbol{\gamma} \in \mathbb{R}^m$, $\delta \in \mathbb{R}$, and $\gamma^\top \mathbf{x} \leq \delta$ represents a single new constraint.

The addition of the new constraint can change the feasibility in three different ways:

Infeasible: With the new constraint added, the new system of inequalities may become infeasible if the feasible region of $C\,\mathbf{x} \leq \boldsymbol{\omega}$ has no intersection with that of $\boldsymbol{\gamma}^\top \mathbf{x} \leq \delta$. Figure 20a is a depiction of this case.

Feasible, but never active: If the entire feasible region of $C\,\mathbf{x} \leq \boldsymbol{\omega}$ is strictly contained in the feasible region of the new constraint $\boldsymbol{\gamma}^\top \mathbf{x} \leq \delta$ and does not intersect with the hyperplane defined by $\boldsymbol{\gamma}^\top \mathbf{x} = \delta$, then the new system is still feasible, but the new constraint can never become active. See Figure 20b for a two-dimensional illustration of the situation.

Feasible, and possibly active: Finally, if the feasible region of $C\,\mathbf{x} \leq \boldsymbol{\omega}$ intersects with the hyperplane defined by $\boldsymbol{\gamma}^\top \mathbf{x} = \delta$, then the new

system is still feasible and the new constraint may become active at some basic-feasible-solution of the new system.

Of the most importances are the first two cases (infeasible and feasible-but-never-active), since they signify that certain extensions are impossible. Therefore if those two cases can be simply detected, then one can immediately eliminate the need to carry out the actual extension attempts, avoiding the most computationally intensive part of the mixed cells enumeration algorithm.

8.4.2. Detecting the infeasible case. Consider the first case where inclusion of the new constraint renders the extended system (8.9) infeasible. Note that a vertex $\mathbf{x}^* = (x_1^*, \ldots, x_m^*)$ of the feasible region defined by $C\mathbf{x} \leq \boldsymbol{\omega}$ is a point in that region where the equality holds for at least m linearly independent constraints. The basic matrix at this point is then the m rows of C corresponding to those constraints. Let B be the basic matrix at the vertex $\mathbf{x}^*$, and let $D = [\mathbf{d}_1, \ldots, \mathbf{d}_m] = B^{-1}$. Then each of the vectors $-\mathbf{d}_1, \ldots, -\mathbf{d}_m$ points to a direction one can move within the feasible region and keep all but one equalities unchanged. With this understanding, if $\mathbf{x}^*$ is not in the feasible region of the new system of inequalities and each possible direction one may follow to leave $\mathbf{x}^*$ points away from the feasible region of the new constraint $\boldsymbol{\gamma}^\top \mathbf{x} \leq \delta$, then the new system must be infeasible, as illustrated by the example in Figure 21. Algebraically, if

1) $\boldsymbol{\gamma}^\top \mathbf{x}^* > \delta$, and
2) $\langle \boldsymbol{\gamma}, -\mathbf{d}_i \rangle > 0$,

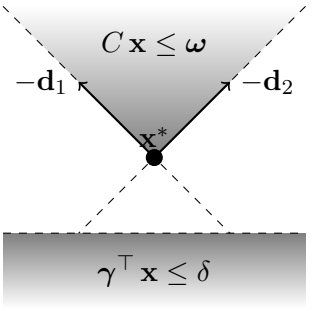


Figure 21. The feasible region of the original system of inequalities $C\,\mathbf{x} \leq \boldsymbol{\omega}$ together with a feasible solution for which two equalities hold.

then the new system (8.9) is infeasible. This can be verified very quickly and only local information at a basic-feasible-solution $\mathbf{x}^*$ of $C\mathbf{x} \leq \boldsymbol{\omega}$ is used.

8.4.3. Detecting the feasible-but-never-active case. Similarly, with B being the basic matrix of $C\mathbf{x} \leq \boldsymbol{\omega}$ at the basic feasible solution $\mathbf{x}^*$ and $D = [\mathbf{d}_1, \ldots, \mathbf{d}_m] = B^{-1}$, then each of the vectors $-\mathbf{d}_1, \ldots, -\mathbf{d}_m$ points to a direction one can move within the feasible region and keep all but one equalities unchanged. If $\mathbf{x}^*$ is already in the interior of the new constraint $\gamma^\top \mathbf{x} \leq \delta$, then the extended system (8.9) is feasible since $\mathbf{x}^*$ is already a feasible solution (actually a basic-feasible-solution).

Additionally, if each possible direction one is allowed to leave $\mathbf{x}^*$ points into the interior of the feasible region of the new constraint $\boldsymbol{\gamma}^\top \mathbf{x} \leq \delta$, then this new constraint can never become active, that is, the equality $\boldsymbol{\gamma}^\top \mathbf{x} = \delta$ can never hold within the feasible region of (8.9). Figure 22 shows a two dimensional illustration of this case.

Algebraically, if

1) $\boldsymbol{\gamma}^\top \mathbf{x}^* \leq \delta$, and

2) $\langle \boldsymbol{\gamma}, -\mathbf{d}_i \rangle \leq 0$ for each $i = 1, \ldots, m$,

then the new system (8.9) is feasible, but the new constraint $\gamma^\top \mathbf{x} \leq \delta$ will never become active.

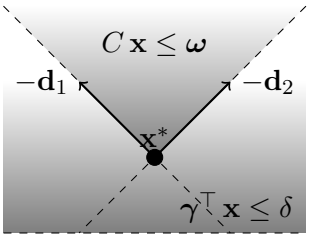


Figure 22. The feasible region of the original system of inequalities $C\,\mathbf{x} \leq \boldsymbol{\omega}$ together with a feasible solution for which two equalities hold.

8.5. Relation tables

A simple yet effective data structure that can reinforce the technique discussed in §8.3.3 on quickly eliminating lower subfaces that cannot exist from the extension process is known as the **relation table**.

In the construction of (8.3), each point imposes an additional constraint. Therefore, for two tuples $(F_1, \ldots, F_r)$ and $(F'_1, \ldots, F'_r)$ with $r \leq$

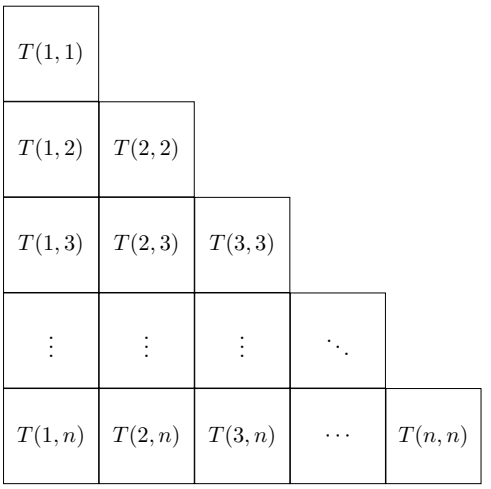
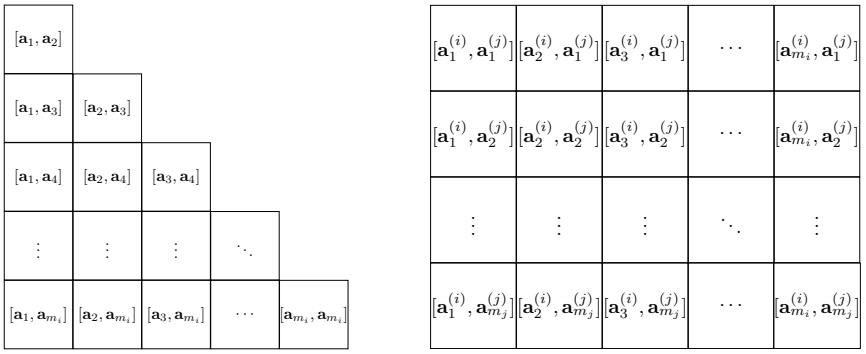


Figure 23. Relation table

n and $F_i, F_i' \subseteq \hat{S}_i$, if $F_i' \supseteq F_i$ for each i, then the system of inequalities $I(F_1', \ldots, F_r')$ is a restriction of $I(F_1, \ldots, F_r)$ in the sense that the feasible region of $I(F_1', \ldots, F_r')$, if nonempty, is contained in the feasible region of $I(F_1, \ldots, F_r)$. Consequently, if $I(F_1, \ldots, F_r)$ is not feasible, nor is $I(F_1', \ldots, F_r')$.

In particular, for a pair of points $\mathbf{a} \in S_i$ and $\mathbf{b} \in S_j$, if $I(\{\hat{\mathbf{a}}\}, \{\hat{\mathbf{b}}\})$ (or $I(\{\hat{\mathbf{a}}, \hat{\mathbf{b}}\})$ when $i = j$) is infeasible, then no lower subfaces containing both points can exist. Subsequently, this information directly eliminates a large number of candidates for one-point tests and greatly reduces the amount of computation. The *relation table* is the data structure that encodes this pairwise relation between points. It consists of pairwise relation subtables $T(i,j)$ between supports S_i and S_j for all $1 \le i \le j \le n$ as shown in Figure 23. Each subtable $T(i,j)$, in turn, encodes the pairwise relationships between points of S_i and S_j in the following sense:

Definition 8.8. Given two points $\mathbf{a}$ and $\mathbf{a}'$ in the same support S_i for some i, we say these two points are **related** if the system of inequalities $I(\{\hat{\mathbf{a}}, \hat{\mathbf{a}}'\})$, defined in (8.3), is feasible, namely, if $\{\hat{\mathbf{a}}, \hat{\mathbf{a}}'\}$ defines a lower edge of $\hat{S}_i$. Similarly, given points $\mathbf{a} \in S_i$ and $\mathbf{a}' \in S_j$ for $i \neq j$, we say these two points are **related** if the system of inequalities $I(\{\hat{\mathbf{a}}\}, \{\hat{\mathbf{a}}'\})$ is feasible, that is if $(\{\hat{\mathbf{a}}\}, \{\hat{\mathbf{a}}'\})$ is a lower $(0,0)$-subface of $(\hat{S}_i, \hat{S}_j)$.



(a) A example of diagonal subtable $T(i,i)$ of the relation table

(b) A example of non-diagonal subtable $T(i,j)$ of the relation table

Figure 24. Examples of subtables of the relation table

The subtable $T(i,j)$ has rows corresponding to points in S_i and columns corresponding to points in S_j. Denote the entry on table $T(i,j)$ located at the row containing $\mathbf{a}_l^{(i)}$ and column containing $\mathbf{a}_m^{(j)}$ by $[\mathbf{a}_l^{(i)}, \mathbf{a}_m^{(j)}]$. Set $[\mathbf{a}_l^{(i)}, \mathbf{a}_m^{(j)}] = 1$ if $\mathbf{a}_l^{(i)}$ and $\mathbf{a}_m^{(j)}$ are related and $[\mathbf{a}_l^{(i)}, \mathbf{a}_m^{(j)}] = 0$ otherwise.

As noted in §8.2, the feasibility of systems of the form $I(\{\hat{\mathbf{a}}, \hat{\mathbf{a}}'\})$ as well as $I(\{\hat{\mathbf{a}}\}, \{\hat{\mathbf{a}}'\})$ can be determined via one-point tests. In particular, for the subtables $T(i,i)$ on the diagonal of the whole relation table, the entry $[\mathbf{a}_l^{(i)}, \mathbf{a}_m^{(i)}]$ can be determined by solving the linear programming problem $LP(\{\hat{\mathbf{a}}_l^{(i)}\}\,;\, \hat{\mathbf{a}}_m^{(i)})$:

$$\text{Maximize } \langle -\mathbf{a}_m^{(i)}, \boldsymbol{\alpha} \rangle + h$$

$$\text{subject to } \begin{cases} \left\langle -\mathbf{a}_l^{(i)}, \boldsymbol{\alpha} \right\rangle + h = \omega_i(\mathbf{a}_l^{(i)}) \\ \quad \langle -\mathbf{a}, \boldsymbol{\alpha} \rangle + h \le \omega_i(\mathbf{a}) \qquad \text{for all} \quad \mathbf{a} \in S_i. \end{cases}$$

If the maximum reaches $\omega_i(\mathbf{a}^{(i)})$, then $[\mathbf{a}_l^{(i)}, \mathbf{a}_m^{(i)}] = 1$, otherwise $[\mathbf{a}_l^{(i)}, \mathbf{a}_m^{(i)}] = 0$. Similarly, for subtables $T(i,j)$ with $i \neq j$, the entry $[\mathbf{a}_l^{(i)}, \mathbf{a}_m^{(j)}]$ can be determined by solving the linear programming problem $LP(\{\hat{\mathbf{a}}_l^{(i)}\}\,;\, \hat{\mathbf{a}}_m^{(j)})$:

$$\text{Maximize } \langle -\mathbf{a}_m^{(j)}, \boldsymbol{\alpha} \rangle + h_2$$

$$\text{subject to } \begin{cases} \langle -\mathbf{a}_l^{(i)}, \boldsymbol{\alpha} \rangle + h_1 = \omega_i(\mathbf{a}_l^{(i)}) \\ \quad \langle -\mathbf{a}, \boldsymbol{\alpha} \rangle + h_1 \le \omega_i(\mathbf{a}) \qquad \text{for all} \quad \mathbf{a} \in S_i \\ \quad \langle -\mathbf{a}, \boldsymbol{\alpha} \rangle + h_2 \le \omega_j(\mathbf{a}) \qquad \text{for all} \quad \mathbf{a} \in S_j. \end{cases}$$

Note that one-point tests of this type involve two supports and therefore the technique described in §8.4 can be used to quickly eliminate a large number of such extensions and thus filling in 0 entries in the relation table without actually performing one-point tests. Conversely, the important technique described in §8.3 for quickly revealing positive extensions in the process of solving the one-point test problem also applies here. It can fill in many 1's in the relation table without solving the corresponding one-point test problems. Actually, *most* entries in $T(i, j)$ are known by these two techniques; therefore, the relation table can usually be filled out relatively fast even having a large number of entries. Moreover, as noted in §8.3.2, the potentially costly "phase one" aspect of these linear programming problems can be avoided in most cases. In fact, the phase one problems only need to be solved once. (See §8.3.2 for details.)

Once filled, the relation table is a great data structure for significantly reducing the computation cost in the following three ways. First, the pairwise relation enables the elimination of a substantial number of one-point tests. As a simple example, let $(\{\hat{\mathbf{a}}_1^{(1)}, \hat{\mathbf{a}}_1^{(2)}\})$ be a (1)-subface of $\hat{S}_1$. If a point $\mathbf{b} \in S_2$ has either $[\mathbf{a}_1^{(1)}, \mathbf{b}] = 0$ or $[\mathbf{a}_2^{(1)}, \mathbf{b}] = 0$, then the one-point test $LP(\{\hat{\mathbf{a}}_1^{(1)}, \hat{\mathbf{a}}_1^{(2)}\}; \hat{\mathbf{b}})$ will be bound to fail, since by Proposition 8.3, if $(\{\hat{\mathbf{a}}_1^{(1)}\}, \{\hat{\mathbf{b}}\})$ or $(\{\hat{\mathbf{a}}_2^{(1)}\}, \{\hat{\mathbf{b}}\})$ are not subfaces, the tuple $(\{\hat{\mathbf{a}}_1^{(1)}, \hat{\mathbf{a}}_2^{(1)}\}, \{\hat{\mathbf{b}}\})$ can never be a subface. In general, for a subface $(F_1, \ldots, F_r)$, a point $\mathbf{b}$ in one of the remaining supports $S_{r+1}, \ldots, S_n$ having $[\mathbf{a}^{(i)}, \mathbf{b}] = 0$ for any $\hat{\mathbf{a}}^{(i)} \in F_i$, $i = 1, \ldots, r$ can be eliminated from the extension process as it is impossible to have an extension containing both $(F_1, \ldots, F_r)$ and $\hat{\mathbf{b}}$. This event alone greatly reduces the number of one-point tests expected to be carried out. Secondly, as explained in §8.3.4, in solving each one-point test problem $LP(F_1, \ldots, F_r; \hat{\mathbf{b}})$, we can remove all constraints corresponding to those points $\mathbf{a}_m^{(\ell)} \in S_\ell$ for $\ell = 1, \ldots, r$ whose relations with any $\hat{\mathbf{a}}_k^{(i)} \in F_i$ are known to be negative, that is, $[\mathbf{a}_m^{(\ell)}, \mathbf{a}_k^{(i)}] = 0$. Finally, for a given subface $(F_1, \ldots, F_r)$, if there is a support S_ℓ in which the number of points having positive relation with *all* points in $F_1, \ldots, F_r$ is less than 2, then the progressive extensions on this subface will *eventually* terminates, and therefore *all* extension attempts start from this subface can be eliminated.

8.6. Support ordering

The mixed volume is clearly independent of the ordering of the supports. That is, for any permutation $(S'_1, \ldots, S'_n)$ of $(S_1, \ldots, S_n)$,

$$\mathcal{M}(S'_1, \ldots, S'_n) = \mathcal{M}(S_1, \ldots, S_n).$$

Similarly, the permutation of components of mixed cells gives rise to mixed cells of the permuted supports. More precisely, if $\mathcal{D}$ is a fine mixed subdivision of the supports $(S_1, \ldots, S_n)$, then for any permutation σ of n objects, $\{\sigma(C_1, \ldots, C_n) \mid (C_1, \ldots, C_n) \in \mathcal{D}\}$ is a fine mixed subdivision of $\sigma(S_1, \ldots, S_n)$. It follows that any support ordering can be used in the mixed cell emulation via the process of subface extensions discussed above.

However, from a computational standpoint, because the possibility of further extensions of subfaces can be eliminated at any (especially early) stage of the extension process, the ordering of the supports generally has a profound effect in the overall cost of the computation.

Normally, it is beneficial to choose the support ordering so that there are a smallest number of starting points for the extension process.[2] This can be achieved by examining the relation table. Before the attempts to extend a $(k_1, \ldots, k_r)$-subface to a $(k_1, \ldots, k_r, 0)$-subface, a potentially large number of points in $S_{r+1}, \ldots, S_n$ can be removed from consideration by information provided by the relation table, as explained in §8.5. Complementing this with the techniques described in §8.4 which can instantly eliminate even more extension possibilities in $S_{r+1}, \ldots, S_n$ will greatly reduce the number of one-point tests in most practical problems. Consequently, a much refined tally of potential candidates for further extensions from each of the remaining supports $S_{r+1}, \ldots, S_n$ is available. A novel idea, initiated in [77, 78], was the realization that at this stage the optimal choice for the next support would be the support with the least amount of remaining potential candidates for successful extension attempts from the current subface. Similar decisions

[2]The opposite choice may be useful in other contexts. In [16] which focuses on performing mixed cell enumeration on computer clusters with potentially a large number of processors, the support with the largest number of (1)-subface (which are the starting point of the extension process) is often chosen to be the first support S'_1 to maximize the initial parallelism. This is of particular importance in cases where some support has a small number of possible (1)-subfaces. For example, if the first support is chosen so that there are only two (1)-subfaces then using these two subfaces as starting points, the all but two processors (in a computer cluster) must stay idle, wasting CPU-time, until more possibility for extension has been discovered.

are made at *every stage* of the extension process whenever a new support is needed. In this way, the support ordering is chosen dynamically at a subface level, and different route in the extension can use different support orderings. As reported in [52] and [77], such a dynamic ordering of the supports has a remarkable effect in the overall efficiency, it substantially reduces the total amount of one-point tests needed in the mixed volume computation. This idea has been incorporated into MixedVol-2.0 [52] as well as many implementations after that (e.g. [16] and [15]).

9. Mixed volume and mixed cells of semi-mixed systems

A polynomial system $P(\mathbf{x}) = (p_1(\mathbf{x}), \ldots, p_n(\mathbf{x}))$ with $\mathbf{x} = (x_1, \ldots, x_n)$ and support $S = (S_1, \ldots, S_n)$ is called **semi-mixed** of type $(k_1, \ldots, k_m)$ when the supports S_j's are not all distinct, but they are equal within m blocks of sizes $k_1, \ldots, k_m$, i.e., there are m sets $S^{(1)}, \ldots, S^{(m)} \subset \mathbb{N}_0^n$ such that $S^{(i)} = S_{i1} = \cdots = S_{ik_i}$ where

$$S_{il} \in \{S_1, \ldots, S_n\} \quad \text{for } 1 \le i \le m,\ 1 \le l \le k_i,$$

and $k_1 + \cdots + k_m = n$. $P(\mathbf{x})$ is called **unmixed** if $m = 1$, that is, all the $S_j's$ are identical, and **fully mixed** if $m = n$, that is, all the $S_j's$ are distinct. We abbreviate $S = (S^{(1)}, k_1; S^{(2)}, k_2; \ldots;\ S^{(m)}, k_m)$, and $Q = (Q^{(1)}, k_1; Q^{(2)}, k_2; \ldots; Q^{(m)}, k_m)$, with $Q^{(i)} = \operatorname{conv} S^{(i)}$ for $i = 1, \ldots, m$. Let $P^{(i)}(\mathbf{x})$ be the subsystem of polynomials in $P(\mathbf{x}) = (p_1(\mathbf{x}), \ldots, p_n(\mathbf{x}))$ having support $S^{(i)}$. Namely, each polynomial of $P^{(i)}(\mathbf{x})$ can be written as

$$p_{il}(\mathbf{x}) = \sum_{\mathbf{a} \in S^{(i)}} c_{il\mathbf{a}} \mathbf{x}^{\mathbf{a}} \quad \text{for } 1 \le i \le m,\ 1 \le l \le k_i. \tag{9.1}$$

For further abbreviation, when no ambiguities exist, we write $S = (S^{(1)}, \ldots, S^{(m)})$ and $P(\mathbf{x}) = (P^{(1)}(\mathbf{x}), \ldots, P^{(m)}(\mathbf{x}))$ at times.

We may, of course, solve a semi-mixed polynomial system $P(\mathbf{x}) = (P^{(1)}(\mathbf{x}), \ldots, P^{(m)}(\mathbf{x}))$ in $\mathbb{C}^n$ by employing the standard polyhedral homotopy procedure in §7 without taking the semi-mixed structure of its supports into account. However, if a special attention is paid to this particular structure, a revised polyhedral homotopy procedure can be developed with a great reduction in the amount of computation, especially when $P(\mathbf{x})$ is unmixed, such as the 9-point problem in mechanism design [114].

To calculate the mixed volume of a semi-mixed system with support $S = (S^{(1)}, k_1;\ S^{(2)}, k_2;\ \ldots; S^{(m)}, k_m)$ alternatively, recall that with $Q^{(i)} =$

$\operatorname{conv} S^{(i)}$ for $i = 1, \ldots, m$ the mixed volume $\mathcal{M}(Q^{(1)}, k_1; Q^{(2)}, k_2; \ldots; Q^{(m)}, k_m)$ is the coefficient of $\prod_{j=1}^{k_1} \lambda_{1,j} \cdots \prod_{j=1}^{k_m} \lambda_{m,j}$ in the expansion of the homogeneous polynomial

$$\operatorname{Vol}_n \left(\sum_{i=1}^{m} \sum_{j=1}^{k_i} \lambda_{ij} Q^{(i)} \right) = \operatorname{Vol}_n \left(\sum_{i=1}^{m} \left(\sum_{j=1}^{k_i} \lambda_{ij} \right) Q^{(i)} \right).$$

Let

$$\beta_i = \sum_{j=1}^{k_i} \lambda_{i,j} \quad \text{for } i = 1, \ldots, m,$$

then the above expression becomes

$$R(\beta_1, \ldots, \beta_m) := \operatorname{Vol}_n(\beta_1 Q^{(1)} + \beta_2 Q^{(2)} + \cdots + \beta_m Q^{(m)})$$

which is a homogeneous polynomial of degree n in the β_i's. Notice that by the multinomial expansion,

$$\beta_i^k = \left(\sum_{j=1}^{k_i} \lambda_{ij} \right)^k = \sum_{t_1 + \cdots + t_{k_i} = k} \frac{k!}{t_1! \cdots t_{k_i}!} \prod_{j=1}^{k_i} \lambda_{ij}^{t_j}.$$

Therefore a monomial in β_i's can be expanded as

$$\beta_1^{r_1} \cdots \beta_m^{r_m} = \prod_{i=1}^{m} \left(\sum_{t_{i1} + \cdots + t_{ir_i} = r_i} \frac{r_i!}{t_{i1}! \cdots t_{ik_i}!} \prod_{j=1}^{k_i} \lambda_{ij}^{t_{ij}} \right).$$

Clearly, such an expansion involves the monomial $\prod_{i=1}^{m} \prod_{j=1}^{k_i} \lambda_{i,j}$ precisely when $t_{i,j} = 1$ for all $i = 1, \ldots, m$ and $j = 1, \ldots, k_i$ which yields $r_i = \sum_{j=1}^{k_i} t_{i,j} = \sum_{j=1}^{k_i} 1 = k_i$. Therefore in

$$R(\beta_1, \ldots, \beta_m) = R\left(\left(\sum_{j=1}^{k_1} \lambda_{1,j} \right), \ldots, \left(\sum_{j=1}^{k_m} \lambda_{m,j} \right) \right),$$

only the monomial $\beta_1^{k_1} \cdots \beta_m^{k_m}$ involves $\prod_{i=1}^{m} \prod_{j=1}^{k_i} \lambda_{i,j}$ and it appears with coefficient $k_1! \cdots k_m!$. We thus have the following:

Proposition 9.1. *For semi-mixed system* $P(\mathbf{x}) = (P^{(1)}(\mathbf{x}), \ldots, P^{(m)}(\mathbf{x}))$ *with support* $S = (S^{(1)}, k_1; S^{(2)}, k_2; \ldots; S^{(m)}, k_m)$, *the mixed volume*

$\mathcal{M}(Q^{(1)}, k_1;\ Q^{(2)}, k_2; \ldots; Q^{(m)}, k_m)$ is the coefficient of $\beta_1^{k_1} \cdots \beta_m^{k_m}$ in the polynomial

$$\mathrm{Vol}_n(\beta_1 Q^{(1)} + \beta_2 Q^{(2)} + \cdots + \beta_m Q^{(m)})$$

multiplied by $k_1! \cdots k_m!$.

To utilize this knowledge in mixed volume computation, we shall slightly extend the framework of *cells* in which the mixed subdivision is defined for the Minkowski sum of polytopes in previous sections. With $S^{(j)}$ being a finite subset of $\mathbb{N}_0^n$ for $j = 1, \ldots, m$, and $m \leq n$, a **cell** of the m-tuple $S = (S^{(1)}, \ldots, S^{(m)})$ is now an m-tuple $C = (C_1, \ldots, C_m)$ of nonempty subsets $C_j \subseteq S^{(j)}$, for $j = 1, \ldots, m$. With similar notations

$$\begin{aligned} \mathrm{type}\, C &= (\dim(\mathrm{conv}\, C_1), \ldots, \dim(\mathrm{conv}\, C_m)) \\ \mathrm{conv}\, C &= \mathrm{conv}\, C_1 + \cdots + \mathrm{conv}\, C_m \\ \mathrm{Vol}_n\, C &= \mathrm{Vol}_n(\mathrm{conv}\, C), \end{aligned}$$

it follows the definition:

Definition 9.2 (Fine semi-mixed subdivision). A **fine semi-mixed subdivision** $\mathcal{D}$ of the m-tuple $S = (S^{(1)}, \ldots, S^{(m)})$ is a collection of cells $C = (C_1, \ldots, C_m)$ of $S = (S^{(1)}, \ldots, S^{(m)})$ such that

(a): $\dim(\mathrm{conv}\, C) = n$ for all $C \in \mathcal{D}$;

(b): For a distinct pair $A, B \in \mathcal{D}$, if $(\mathrm{conv}\, A) \cap (\mathrm{conv}\, B)$ is nonempty, then it is a common face of both;

(c): $\bigcup_{C \in \mathcal{D}} \mathrm{conv}\, C = \mathrm{conv}\, S_1 + \cdots + \mathrm{conv}\, S_m$;

(d1): For each cell $C = (C_1, \ldots, C_m) \in \mathcal{D}$, $\sum_{j=1}^m \dim(\mathrm{conv}\, C_j) = n$

(d2): For distinct pair of cells $A = (A_1, \ldots, A_m), B = (B_1, \ldots B_m) \in \mathcal{D}$,

$$(\mathrm{conv}\, A) \cap (\mathrm{conv}\, B) = \sum_{j=1}^{m} (\mathrm{conv}\, A_j \cap \mathrm{conv}\, B_j);$$

(e): For each cell $C = (C_1, \ldots, C_m) \in \mathcal{D}$, $\mathrm{conv}\, C_j$ is a simplex of dimension $\#C_j - 1$ for $j = 1, \ldots, m$.

Notice that replacing all the m's in the above definition by n yields exactly the same *fine mixed subdivision* in Definition 6.8 for a fully mixed system. Most importantly, the properties of fine mixed subdivisions proved

in §6.2 can be preserved with minor adjustments. In particular, the scaling invariance of a fine semi-mixed subdivision remains valid:

Proposition 9.3. *If $\mathcal{D}$ is a fine semi-mixed subdivision of the m-tuple $S = (S^{(1)}, \ldots, S^{(m)})$, and $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_m) \in (\mathbb{R}^+)^m$, the set*

$$\boldsymbol{\beta} \circ \mathcal{D} := \{\boldsymbol{\beta} \circ C := (\beta_1 C_1, \ldots, \beta_m C_m) \ : \ C = (C_1, \ldots, C_m) \in \mathcal{D}\}$$

forms a fine semi-mixed subdivision of $\boldsymbol{\beta} \circ S := (\beta_1 S^{(1)}, \ldots, \beta_m S^{(m)})$.

Also, similar to (6.11), the volume of $\beta_1 Q^{(1)} + \beta_2 Q^{(2)} + \cdots + \beta_m Q^{(m)}$ can be expressed in terms of volumes of individual cells of a fine semi-mixed subdivision:

Proposition 9.4. *If $\mathcal{D}$ is a fine semi-mixed subdivision of the m-tuple $S = (S^{(1)}, \ldots, S^{(m)})$, and $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_m) \in (\mathbb{R}^+)^m$,*

$$\mathrm{Vol}_n(\beta_1 Q^{(1)} + \cdots + \beta_m Q^{(m)}) = \sum_{C \in \mathcal{D}} \beta_1^{r_1} \cdots \beta_m^{r_m} \mathrm{Vol}_n\,(\mathrm{conv}\, C) \tag{9.2}$$

where $(r_1, \ldots, r_m)$ denotes the type of $C = (C_1, \ldots, C_m) \in \mathcal{D}$.

For a mixed cell $C = (C_1, \ldots, C_m)$ in the fine semi-mixed subdivision $\mathcal{D}$ of the m-tuple $S = (S^{(1)}, \ldots, S^{(m)})$ with type $C = (r_1, \ldots, r_m)$, where $r_j = \dim(\mathrm{conv}\, C_j)$, we have, by definition, $r_1 + \cdots + r_m = n$, and each $\mathrm{conv}\, C_j$ is a simplex of dimension $r_j = \#C_j - 1$. Write $C_j = \{\mathbf{a}_0^j, \ldots, \mathbf{a}_{r_j}^j\}$ for each $j = 1, \ldots, m$, and define the $n \times r_j$ (empty when $r_j = 0$) matrix

$$V(C_j) := \begin{bmatrix} \mathbf{a}_1^j - \mathbf{a}_0^j & \cdots & \mathbf{a}_{r_j}^j - \mathbf{a}_0^j \end{bmatrix}, \tag{9.3}$$

and combining them to construct the block matrix

$$V(C) = \begin{bmatrix} V(C_1) & \cdots & V(C_m) \end{bmatrix}. \tag{9.4}$$

with size $n \times (r_1 + \cdots + r_m) = n \times n$ in which the block $V(C_j)$ will not appear if $\#C_j = 1$. When $\mathcal{D}$ is a fine semi-mixed subdivision, similar to (6.13), the volume of a cell $C = (C_1, \ldots, C_m) \in \mathcal{D}$ with type $C = (r_1, \ldots, r_m)$, can be computed via the formula

$$\mathrm{Vol}_n(C) = \frac{1}{r_1! \cdots r_m!} \, |\det V(C)|.$$

This yields the expression of the mixed volume of a semi-mixed system as a generalization of (6.13):

Proposition 9.5. *Let $\mathcal{D}$ be a fine semi-mixed subdivision of the support $S = (S^{(1)}, k_1; \ldots;\ S^{(m)}, k_m)$ of a semi-mixed system $P(\mathbf{x}) = (P^{(1)}(\mathbf{x}), \ldots, P^{(m)}(\mathbf{x}))$. The mixed volume $\mathcal{M}(Q^{(1)}, k_1; \ldots;\ Q^{(m)}, k_m)$ of this system is*

$$\mathcal{M}(Q^{(1)}, k_1; Q^{(2)}, k_2; \ldots; Q^{(m)}, k_m) = \sum_{\substack{C \in \mathcal{D} \\ \text{type } C = (k_1, \ldots, k_m)}} |\det V(C)|, \tag{9.5}$$

with $V(C)$ as defined in (9.4).

Moreover, similar to inducing a fine mixed subdivision for a fully mixed system $S = (S_1, \ldots, S_n)$ by a generic lifting function as described in §6.3, the same procedure can be followed almost line by line to induce a fine semi-mixed subdivision by a generic lifting function for the semi-mixed support $S = (S^{(1)}, k_1;\ \ldots;\ S^{(m)}, k_m)$ of a semi-mixed polynomial system. Hence Proposition 6.16 can be generalized to the semi-mixed system:

Proposition 9.6 (Induced fine semi-mixed subdivision). *Let $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_m)$ be a generic lifting function for the m-tuple $S = (S^{(1)}, \ldots, S^{(m)})$. Define*

$$\hat{S}^{(j)} := \{(\mathbf{a}, \omega_j(\mathbf{a})) : \mathbf{a} \in S^{(j)}\}$$

for each $j = 1, \ldots, m$. Let $\hat{\mathcal{D}}_{\boldsymbol{\omega}}$ be the collection of all $\hat{C} = (\hat{C}_1, \ldots, \hat{C}_m)$ with $\hat{C}_j \subseteq \hat{S}^{(j)}$ for each $j = 1, \ldots, m$ such that

1) *$\operatorname{conv} \hat{C}_j$ is a lower face of $\operatorname{conv} \hat{S}^{(j)}$ for each $j = 1, \ldots, m$;*
2) *The m lower faces $\operatorname{conv} \hat{C}_j$ of $\operatorname{conv} \hat{S}^{(j)}$ for $j = 1, \ldots, m$ respectively share a common inner normal of the form $\hat{\boldsymbol{\alpha}} = (\boldsymbol{\alpha}, 1)$ with $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n) \in \mathbb{R}^n$; and*
3) *$\dim(\operatorname{conv} \hat{C}_1) + \cdots + \dim(\operatorname{conv} \hat{C}_m) = n$.*

Then the projections of those cells

$$\mathcal{D}_{\boldsymbol{\omega}} = \{(\pi(\hat{C}_1), \ldots, \pi(\hat{C}_m)))\ :\ (\hat{C}_1, \ldots, \hat{C}_m) \in \hat{\mathcal{D}}_{\boldsymbol{\omega}}\}$$

form a fine semi-mixed subdivision of $S = (S^{(1)}, \ldots, S^{(m)})$, and it is called the ***fine semi-mixed subdivision induced by the lifting function $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_m)$.***

For cell $C = (C_1, \ldots, C_m) \in \mathcal{D}_{\boldsymbol{\omega}}$, the vector $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n) \in \mathbb{R}^n$ above will also be referred to as its *inner normal*. To emphasize their connection, we write $C^{\boldsymbol{\alpha}} = (C_1, \ldots, C_m)$ at times.

For $P(\mathbf{x}) = (p_1(\mathbf{x}), \ldots, p_n(\mathbf{x}))$ with support $S = (S_1, \ldots, S_n)$ in general position, again we assume, for simplicity, all the p_j's have constant terms, namely, $S_j = S'_j = S_j \bigcup \{0\}$ for $j = 1, \ldots, n$. Recall that at the beginning of the polyhedral homotopy procedure, we first assign, in *Step 1.2*, a generic lifting $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_n)$ on $S = (S_1, \ldots, S_n)$. Now for semi-mixed system $P(\mathbf{x}) = (P^{(1)}(\mathbf{x}), \ldots, P^{(m)}(\mathbf{x}))$ with support $S = (S^{(1)}, k_1; \ldots; S^{(m)}, k_m)$ and generic $c_{il\mathbf{a}} \in \mathbb{C}^*$ as given in (9.1), we choose generic lifting $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_m)$ on $S = (S^{(1)}, \ldots, S^{(m)})$ where $\omega_j : S^{(j)} \to \mathbb{Q}$ for $j = 1, \ldots, m$ and look at the homotopy $Q(\mathbf{x}, t) = (Q^{(1)}(\mathbf{x}, t), \ldots, Q^{(m)}(\mathbf{x}, t)) = \mathbf{0}$ where equations in $Q^{(j)}(\mathbf{x}, t) = \mathbf{0}$ for $1 \le j \le m$ are

$$q_{jl}(\mathbf{x}, t) = \sum_{\mathbf{a} \in S^{(j)}} c_{jl\mathbf{a}} \mathbf{x}^{\mathbf{a}} t^{\omega_j(\mathbf{a})} = 0, \quad 1 \le l \le k_j. \tag{9.6}$$

Immediately, $Q(\mathbf{x}, 1) = P(\mathbf{x})$. Let $\mathcal{D}_{\boldsymbol{\omega}}$ be the fine semi-mixed subdivision of $S = (S^{(1)}, \ldots, S^{(m)})$ induced by the lifting function $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_m)$, and let $C^{\boldsymbol{\alpha}} = (C_1, \ldots, C_m)$ be a cell of type $(k_1, \ldots, k_m)$ in $\mathcal{D}_{\boldsymbol{\omega}}$ having inner normal $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n) \in \mathbb{Q}^n$. With $\hat{\boldsymbol{\alpha}} = (\boldsymbol{\alpha}, 1)$ and $\hat{\mathbf{a}} = (\mathbf{a}, \omega_j(\mathbf{a}))$ for $\mathbf{a} \in S^{(j)}$, the cell $C^{\boldsymbol{\alpha}} = (C_1, \ldots, C_m)$ where $C_j = \{\mathbf{a}^j_0, \ldots, \mathbf{a}^j_{k_j}\} \subset S^{(j)}$ for $j = 1, \ldots, m$ must satisfy the following condition:

(A') *For* $1 \le j \le m$

$$\langle \hat{\mathbf{a}}^j_l, \hat{\boldsymbol{\alpha}} \rangle = \langle \hat{\mathbf{a}}^j_{l'}, \hat{\boldsymbol{\alpha}} \rangle \quad \text{for} \quad 0 \le l, l' \le k_j$$

$$\textit{and} \quad \langle \hat{\mathbf{a}},\ \hat{\boldsymbol{\alpha}} \rangle > \langle \hat{\mathbf{a}}^j_l,\ \hat{\boldsymbol{\alpha}} \rangle \quad \text{for} \quad 0 \le l \le k_j \ \textit{and}\ \mathbf{a} \in S^{(j)} \backslash C_j.$$

With coordinate transformation $\mathbf{y} = t^{-\boldsymbol{\alpha}}\mathbf{x}$ where $y_i = t^{-\alpha_i} x_i$ for $i = 1, \ldots, n$, equations in (9.6) becomes

$$q_{jl}(\mathbf{y} t^{\boldsymbol{\alpha}}, t) = \sum_{\mathbf{a} \in S^{(j)}} c_{jl\mathbf{a}} \mathbf{y}^{\mathbf{a}} t^{\langle \hat{\mathbf{a}}, \hat{\boldsymbol{\alpha}} \rangle}, \quad 1 \le j \le m,\ 1 \le l \le k_j.$$

Let

$$\beta_j = \min_{\mathbf{a} \in S^{(j)}} \langle \hat{\mathbf{a}}, \hat{\boldsymbol{\alpha}} \rangle, \quad j = 1, \ldots, m,$$

and consider the homotopy

$$H^{\boldsymbol{\alpha}}(\mathbf{y}, t) = (H^{\boldsymbol{\alpha}}_1(\mathbf{y}, t), \ldots, H^{\boldsymbol{\alpha}}_m(\mathbf{y}, t)) = \mathbf{0} \tag{9.7}$$

on $(C^*)^n \times [0,1]$ where for $1 \le j \le m$ equations in $H_j^{\boldsymbol{\alpha}}(\mathbf{y},t) = \mathbf{0}$ are

$$\begin{aligned} h_{jl}^{\boldsymbol{\alpha}}(\mathbf{y},t) &= t^{-\beta_j} q_{jl}(\mathbf{y}t^{\boldsymbol{\alpha}},t) \\ &= \sum_{\mathbf{a}\in S^{(j)}} c_{jl\mathbf{a}} \mathbf{y}^{\mathbf{a}} t^{\langle \hat{\mathbf{a}},\hat{\boldsymbol{\alpha}}\rangle - \beta_j} \\ &= \sum_{\substack{\mathbf{a}\in S^{(j)} \\ \langle \hat{\mathbf{a}},\hat{\boldsymbol{\alpha}}\rangle = \beta_j}} c_{jl\mathbf{a}} \mathbf{y}^{\mathbf{a}} + \sum_{\substack{\mathbf{a}\in S^{(j)} \\ \langle \hat{\mathbf{a}},\hat{\boldsymbol{\alpha}}\rangle > \beta_j}} c_{jl\mathbf{a}} \mathbf{y}^{\mathbf{a}} t^{\langle \hat{\mathbf{a}},\hat{\boldsymbol{\alpha}}\rangle - \beta_j} = 0, \quad 1 \le l \le k_j. \end{aligned}$$

When $t = 0$, equations in $H_j^{\boldsymbol{\alpha}}(\mathbf{y},0) = \mathbf{0}$ become, by condition (A'),

$$h_{jl}^{\boldsymbol{\alpha}}(\mathbf{y},0) = \sum_{\mathbf{a}\in C_j = \{\mathbf{a}_0^j,\ldots,\mathbf{a}_{k_j}^j\}} c_{jl\mathbf{a}} \mathbf{y}^{\mathbf{a}} = 0, \quad 1 \le l \le k_j. \tag{9.8}$$

For each $1 \le j \le m$, the above system consists of k_j equations, each one has the same $k_j + 1$ monomials $\{\mathbf{y}^{\mathbf{a}_0^j}, \ldots, \mathbf{y}^{\mathbf{a}_{k_j}^j}\}$. By applying Gaussian elimination to its $k_j \times (k_j+1)$-coefficient matrix $(c_{jl\mathbf{a}})$, we can replace $H_j^{\boldsymbol{\alpha}}(\mathbf{y},0) = \mathbf{0}$ by an equivalent binomial system

$$\begin{aligned} c'_{j11}\mathbf{y}^{\mathbf{a}_1^j} + c'_{j10}\mathbf{y}^{\mathbf{a}_0^j} &= 0 \\ &\vdots \\ c'_{jk_j1}\mathbf{y}^{\mathbf{a}_{k_j}^j} + c'_{jk_j0}\mathbf{y}^{\mathbf{a}_0^j} &= 0. \end{aligned}$$

Write

$$V(C_j) := \left[\mathbf{a}_1^j - \mathbf{a}_0^j \quad \cdots \quad \mathbf{a}_{k_j}^j - \mathbf{a}_0^j\right].$$

Repeating this process for each $H_j^{\boldsymbol{\alpha}}(\mathbf{y},0) = \mathbf{0}$, $j = 1,\ldots,m$, and combining all those binomial equations, a system of $k_1 + \cdots + k_m = n$ binomial equations in n variables is produced. This binomial system is equivalent to the start system $H^{\boldsymbol{\alpha}}(\mathbf{y},0) = \mathbf{0}$ of the homotopy $H^{\boldsymbol{\alpha}}(\mathbf{y},t) = \mathbf{0}$ in (9.7), which admits $|\det V(C^{\boldsymbol{\alpha}})|$ nonsingular isolated zeros in $(\mathbb{C}^*)^n$ as shown in Lemma 4.1 where

$$V(C^{\boldsymbol{\alpha}}) = \left[V(C_1) \quad \cdots \quad V(C_m)\right].$$

Following solution paths of $H^{\boldsymbol{\alpha}}(\mathbf{y},t) = \mathbf{0}$ emanating from those isolated zeros, we will reach $|\det V(C^{\boldsymbol{\alpha}})|$ isolated zeros of $P(\mathbf{x})$.

As in Proposition 9.5, the mixed volume $\mathcal{M}(S^{(1)}, k_1; \ldots; S^{(m)}, k_m)$, the total root count of $P(\mathbf{x})$ in $(\mathbb{C}^*)^n$, is equal to

$$\mathcal{M}(S^{(1)}, k_1; \ldots; S^{(m)}, k_m) = \sum_{\substack{C^{\boldsymbol{\alpha}} \in \mathcal{D}_{\boldsymbol{\omega}} \\ \text{type}(C^{\boldsymbol{\alpha}}) = (k_1, \ldots, k_m)}} |\det V(C^{(\boldsymbol{\alpha})})|,$$

Therefore, we may obtain all isolated zeros of $P(\mathbf{x})$ by repeating the above process for all cells of type $(k_1, \ldots, k_m)$ in $\mathcal{D}_{\boldsymbol{\omega}}$, the fine semi-mixed subdivision of $S = (S^{(1)}, \ldots, S^{(m)})$ induced by the lifting function $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_m)$.

Remark 9.7. The above procedure for the special case of $m = 1$ (known as the unmixed systems) will be elaborated in §13.3 in more details.

10. Finding isolated zeros in $\mathbb{C}^n$ via stable cells

As remarked in the end of §4, in order to reach all isolated zeros of a polynomial system $P(\mathbf{x}) = (p_1(\mathbf{x}), \ldots, p_n(\mathbf{x}))$ with support $S = (S_1, \ldots, S_n)$ in $\mathbb{C}^n$, we need to follow $k = \mathcal{M}(S'_1, \ldots, S'_n)$ homotopy paths, where $S'_j = S_j \cup \{\mathbf{0}\}$, $j = 1, \ldots, n$. By Theorem 6.5, the number k represents an upper bound for the root count of the system $P(\mathbf{x})$ in $\mathbb{C}^n$. However, as shown in [43], this bound may not be exact, and in [43] a tighter upper bound for the root count in $\mathbb{C}^n$ of the system $P(\mathbf{x}) = (p_1(\mathbf{x}), \ldots, p_n(\mathbf{x}))$ was given. Based on this root count, one may employ alternative algorithms to approximate all isolated zeros of $P(\mathbf{x})$ in $\mathbb{C}^n$ by following fewer homotopy paths. For simplicity, we only focus on fully mixed polynomial systems.

For a given generic lifting $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_n)$ on $S' = (S'_1, \ldots, S'_n)$, we write $\hat{\mathbf{a}} = (\mathbf{a}, \omega_j(\mathbf{a}))$ for $\mathbf{a} \in S'_j$ and $\hat{C}_j = \{\hat{\mathbf{a}} \mid \mathbf{a} \in C_j\}$ for $C_j \subset S'_j$. Let $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n) \in \mathbb{Q}^n$ be the inner normal of cell $C = (C_1, \ldots, C_n)$ in the subdivision $\mathcal{D}_{\boldsymbol{\omega}}$ of $S = (S'_1, \ldots, S'_n)$ induced by $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_n)$. Such cell $C = (C_1, \ldots, C_n)$ is denoted by $C^{\boldsymbol{\alpha}}$ as mentioned before. When $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n)$ is nonnegative, *i.e.*, $\alpha_j \geq 0$ for all $j = 1, \ldots, n$, we call $C^{\boldsymbol{\alpha}}$ a *stable cell* of $S' = (S', \ldots, S'_n)$ *with respect to the lifting* $\boldsymbol{\omega}$. The term **stable cell** alone, without specification of its corresponding lifting, is reserved for stable cells with respect to the particular lifting $\boldsymbol{\omega}^1_0 = (\omega^{01}_1, \ldots, \omega^{01}_n)$ where $\omega^{01}_j : S'_j \to \mathbb{Q}$ for $j = 1, \ldots, n$ is defined as:

$$\begin{aligned} \omega^{01}_j(\mathbf{0}) &= 1 \quad \text{if } \mathbf{0} \notin S_j \\ \omega^{01}_j(\mathbf{a}) &= 0 \quad \text{for } \mathbf{a} \in S_j. \end{aligned}$$

Obviously, $S = (S, \ldots, S_n)$ itself is a stable cell with inner normal $\boldsymbol{\alpha} = (0, \ldots, 0)$ (with respect to the particular lifting $\boldsymbol{\omega}_0^1 = (\omega_1^{01}, \ldots, \omega_n^{01})$).

Definition 10.1. The stable mixed volume of $S = (S, \ldots, S_n)$, denoted by $\mathcal{S}M(S_1, \ldots, S_n)$, is the sum of mixed volumes of all stable cells of $S = (S, \ldots, S_n)$.

With this definition, a tighter bound for the root count of $P(\mathbf{x})$ in $\mathbb{C}^n$ is given in the following

Theorem 10.2 ([43]). *For polynomial system $P(\mathbf{x}) = (p_1(\mathbf{x}), \ldots, p_n(\mathbf{x}))$ with support $S = (S, \ldots, S_n)$, the stable mixed volume $\mathcal{S}M(S_1, \ldots, S_n)$ satisfies:*

$$\mathcal{M}(S_1, \ldots, S_n) \leq \mathcal{S}M(S_1, \ldots, S_n) \leq \mathcal{M}(S_1 \cup \{\mathbf{0}\}, \ldots, S_n \cup \{\mathbf{0}\}). \tag{10.1}$$

Moreover, it provides an upper bound for the root count of $P(\mathbf{x})$ in $\mathbb{C}^n$.

Based on the derivation of Theorem 10.2, it was suggested in [43] that one may find all isolated zeros of polynomial system $P(\mathbf{x}) = (p_1(\mathbf{x}), \ldots, p_n(\mathbf{x})$ in $\mathbb{C}^n$ with support $(S_1, \ldots, S_n)$ where

$$p_j(\mathbf{x}) = \sum_{\mathbf{a} \in S_j} c_{j,\mathbf{a}} \mathbf{x}^{\mathbf{a}}, \quad j = 1, \ldots, n$$

by the following procedure:

Step 1: Identify all stable cells $C^{\boldsymbol{\alpha}} = (C_1, \ldots, C_n)$ of $(S_1, \ldots, S_n)$.

Step 2: For each stable cell $C^{\boldsymbol{\alpha}} = (C_1, \ldots, C_n)$ with inner normal $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n) \geq \mathbf{0}$, find all isolated zeros in $(\mathbb{C}^*)^n$ of the *support* system $P^{\boldsymbol{\alpha}}(\mathbf{x}) = (p_1^{\boldsymbol{\alpha}}(\mathbf{x}), \ldots, p_n^{\boldsymbol{\alpha}}(\mathbf{x}))$ where for $j = 1, \ldots, n$,

$$p_j^{\boldsymbol{\alpha}}(\mathbf{x}) = \sum_{\mathbf{a} \in C_j \cap S_j} c_{j,\mathbf{a}} \mathbf{x}^{\mathbf{a}} + \epsilon_j \tag{10.2}$$

here, $\epsilon_j = 0$ if $0 \in S_j$, otherwise it is an arbitrary nonzero number.

Step 3: For each isolated zero $\mathbf{z} = (z_1, \ldots, z_n)$ of $P^{\boldsymbol{\alpha}}(\mathbf{x})$ in $(\mathbb{C}^*)^n$, let $\bar{\mathbf{z}} = (\bar{z}_1, \ldots, \bar{z}_n)$ where for $j = 1, \ldots, n$,

$$\begin{aligned} \bar{z}_j &= z_j \quad \text{if } \alpha_j = 0 \\ \bar{z}_j &= 0 \quad \text{if } \alpha_j \neq 0. \end{aligned}$$

Then $\bar{\mathbf{z}}$ is an isolated zero of $P(\mathbf{x}) = (p_1(\mathbf{x}), \ldots, p_n(\mathbf{x}))$.

Inevitably, zeros $\mathbf{z} = (z_1, \ldots, z_n)$ of $P^{\boldsymbol{\alpha}}(\mathbf{x})$ will depend on $\boldsymbol{\epsilon} = (\epsilon_1, \ldots, \epsilon_n)$. However, it can be shown [43] that the transition from $\mathbf{z}$ to $\bar{\mathbf{z}}$ given above actually eliminates this dependency.

In Step 2 above, when polyhedral homotopy is used to find all isolated zeros in $(\mathbb{C}^*)^n$ of the support system $P^{\boldsymbol{\alpha}}(\mathbf{x})$ corresponding to the stable cell $C^{\boldsymbol{\alpha}} = (C_1, \ldots, C_n)$, one follows $\mathcal{M}(C_1, \ldots, C_n)$ homotopy paths. Accordingly, the total number of homotopy paths one needs to follow to reach all isolated zeros of $P(\mathbf{x})$ in $\mathbb{C}^n$ equals to the stable mixed volume $\mathcal{S}M(S_1, \ldots, S_n)$, which is strictly fewer than $\mathcal{M}(S_1 \cup \{\mathbf{0}\}, \ldots, S_n \cup \{\mathbf{0}\})$ in general, therefore admitting less extraneous paths.

However, there are difficulties to implement this procedure efficiently. First of all, types of those stable cells are undetermined. They may not be mixed cells, cells of type $(1, \ldots, 1)$ with respect to the lifting $\boldsymbol{\omega}_0^1 = (\omega_1^{01}, \ldots, \omega_n^{01})$, which invalidates the algorithm we developed in §8 for finding mixed cells. This makes their identification in Step 1 rather difficult. Secondly, when polyhedral homotopy is used in Step 2 to solve $P^{\boldsymbol{\alpha}}(\mathbf{x}) = \mathbf{0}$ in $(\mathbb{C}^*)^n$, one must find all mixed cells of a subdivision of $C^{\boldsymbol{\alpha}} = (C_1, \ldots, C_n)$ induced by a further generic lifting on $C^{\boldsymbol{\alpha}}$ in the first place. This accumulated work for all the stable cells can be very costly, which may not be more favorable compared to solving $P(\mathbf{x}) = \mathbf{0}$ in $\mathbb{C}^n$ by following the polyhedral homotopy procedure given in §7 directly with a generic lifting on $S' = (S'_1, \ldots, S'_n)$ permitting some of the homotopy paths be extraneous.

A revision of the above procedure appeared in [31]. To begin, for $k \geq 0$, let $\boldsymbol{\omega}_0^k = (\omega_1^{0k}, \ldots, \omega_n^{0k})$, be the lifting on $S' = (S'_1, \ldots, S'_n)$ where for $j = 1, \ldots, n$,

$$\begin{aligned} \omega_j^{0k}(\mathbf{0}) &= k \quad \text{if } \mathbf{0} \notin S_j \\ \omega_j^{0k}(\mathbf{a}) &= 0 \quad \text{for } \mathbf{a} \in S_j. \end{aligned} \tag{10.3}$$

It is easy to see that the set of stable cells with respect to $\boldsymbol{\omega}_0^k$ remains invariant for different k's. For instance, if $C = (C_1, \ldots, C_n)$ is a stable cell with respect to the lifting $\boldsymbol{\omega}_0^{k_1}$ with inner normal $\boldsymbol{\alpha} \geq \mathbf{0}$, then $C = (C_1, \ldots, C_n)$ is also a stable cell with respect to the lifting $\boldsymbol{\omega}_0^{k_2}$ with inner normal $\frac{k_2}{k_1}\boldsymbol{\alpha} \geq \mathbf{0}$. Denote this set of stable cells by $\mathcal{T}$. Let $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_n)$ be a generic lifting on $S' = (S'_1, \ldots, S'_n)$ where for $j = 1, \ldots, n$,

$$\begin{aligned} \omega_j(\mathbf{0}) &= k \quad \text{for } \mathbf{0} \notin S_j \\ \omega_j(\mathbf{a}) &= \text{a generic number in } (0, 1) \quad \text{for } \mathbf{a} \in S_j. \end{aligned} \tag{10.4}$$

For a cell $C = (C_1, \ldots, C_n)$ in the subdivision of $S' = (S'_1, \ldots, S'_n)$ induced by the lifting $\boldsymbol{\omega}_0^k = (\omega_1^{0k}, \ldots, \omega_n^{0k})$, let $\boldsymbol{\omega}^C$ be the restriction of $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_n)$ on C, which can, of course, be considered as a generic lifting on $C = (C_1, \ldots, C_n)$. It was shown in [31] that if k is sufficiently large, mixed cell $D = (D_1, \ldots, D_n)$ of subdivision $S_{\boldsymbol{\omega}}$ of $S' = (S'_1, \ldots, S'_n)$ induced by the lifting $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_n)$ is also a mixed cell of subdivision S_{ω^C} of certain cell $C = (C_1, \ldots, C_n)$ of $S_{\boldsymbol{\omega}}$ induced by the lifting $\boldsymbol{\omega}^C$. In this situation, stable cell $C = (C_1, \ldots, C_n)$ in $\mathcal{T}$ can be assembled by grouping a collection of proper cells in $S_{\boldsymbol{\omega}}$, and consequently, mixed cells in this collection provides all the mixed cells of subdivision $S_{\boldsymbol{\omega}^C}$ of $C = (C_1, \ldots, C_n)$. To be more precise, when $k \geq n(n+1)d^n$ [31] where $d = \max_{1 \leq j \leq n} \deg p_j(\mathbf{x})$, any mixed cell $D = (D_1, \ldots, D_n)$ in the subdivision $S_{\boldsymbol{\omega}}$ induced by the lifting $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_n)$ on $S' = (S'_1, \ldots, S'_n)$ given in (10.4) is a mixed cell of subdivision $S_{\boldsymbol{\omega}^C}$ induced by the lifting $\boldsymbol{\omega}^C$ of certain cell $C = (C_1, \ldots, C_n)$ in the subdivision $S_{\boldsymbol{\omega}_0^k}$ induced by the lifting $\boldsymbol{\omega}_0^k = (\omega_1^{0k}, \ldots, \omega_n^{0k})$ on $S' = (S'_1, \ldots, S'_n)$ given in (10.3).

Let $D^* = (D_1, \ldots, D_n)$ be any cell in the subdivision $S_{\boldsymbol{\omega}}$ which may or may not be of type $(1, \ldots, 1)$. Let

$$D_j = \{\mathbf{a}_{j0}, \ldots, \mathbf{a}_{jk_j}\}, \quad j = 1, \ldots, n,$$

where $k_1 + \cdots + k_n = n$. For $j = 1, \ldots, n$ and $\mathbf{a} \in S'_j$, write $\hat{\mathbf{a}}(k) = (\mathbf{a}, \omega_j^{0k}(\mathbf{a}))$. Let $\hat{D}_j(k) = \{\hat{\mathbf{a}}(k) \mid \mathbf{a} \in D_j\}$ for $j = 1, \ldots, n$ and $\hat{D}^*(k) = (\hat{D}_1(k), \ldots, \hat{D}_n(k))$. Apparently, the $n \times (n+1)$ matrix

$$V(\hat{D}^*(k)) = \begin{bmatrix} \hat{\mathbf{a}}_{11}^\top(k) - \hat{\mathbf{a}}_{10}^\top(k) \\ \vdots \\ \hat{\mathbf{a}}_{1k_1}^\top(k) - \hat{\mathbf{a}}_{10}^\top(k) \\ \vdots \\ \hat{\mathbf{a}}_{n1}^\top(k) - \hat{\mathbf{a}}_{n0}^\top(k) \\ \vdots \\ \hat{\mathbf{a}}_{nk_n}^\top(k) - \hat{\mathbf{a}}_{n0}^\top(k) \end{bmatrix}$$

is of rank n. Let $\boldsymbol{\alpha} \in \mathbb{Q}^n$ be the unique vector where $\hat{\boldsymbol{\alpha}} = (\boldsymbol{\alpha}, 1)$ is in the kernel of $V(\hat{D}^*(k))$. This $\boldsymbol{\alpha}$ is the inner normal of D^* with respect to $\boldsymbol{\omega}_0^k$. Let $\mathcal{T}(\boldsymbol{\alpha})$ be the collection of all mixed cells in $S_{\boldsymbol{\omega}}$ with the same nonnegative inner normal $\boldsymbol{\alpha}$ with respect to $\boldsymbol{\omega}_0^k$ and let $D = (\{\mathbf{a}_{10}, \mathbf{a}_{11}\}, \ldots, \{\mathbf{a}_{n0}, \mathbf{a}_{n1}\})$ where $\{\mathbf{a}_{j0}, \mathbf{a}_{j1}\} \subset S'_j$ for $j = 1, \ldots, n$ be any mixed cell in $\mathcal{T}(\boldsymbol{\alpha})$. Let $C =$

$(C_1, \ldots, C_n)$ where

$$C_j = \{\mathbf{a} \in S'_j \mid \langle \hat{\mathbf{a}}(k), \hat{\boldsymbol{\alpha}} \rangle = \langle \hat{\mathbf{a}}_{j0}(k), \hat{\boldsymbol{\alpha}} \rangle\}, \quad j = 1, \ldots, n,$$

This cell satisfies, for $j = 1, \ldots, n$,

$$\begin{aligned} &\langle \hat{\mathbf{a}}(k), \hat{\boldsymbol{\alpha}} \rangle = \langle \hat{\mathbf{b}}(k), \hat{\boldsymbol{\alpha}} \rangle \quad \text{for } \mathbf{a}, \mathbf{b} \in C_j \\ &\langle \hat{\mathbf{a}}(k), \hat{\boldsymbol{\alpha}} \rangle < \langle \hat{\mathbf{d}}(k), \hat{\boldsymbol{\alpha}} \rangle \quad \text{for } \mathbf{a} \in C_j, \ \ \mathbf{d} \in S'_j \backslash C_j. \end{aligned}$$

It is therefore a stable cell with respect to $\boldsymbol{\omega}_0^k$ with inner normal $\boldsymbol{\alpha}$, which, as mentioned above, is also a stable cell with respect to $\boldsymbol{\omega}_0^1$ with inner normal $\frac{1}{k}\boldsymbol{\alpha}$. In the meantime, the cells in the collection $\mathcal{T}(\boldsymbol{\alpha})$ gives *all* the mixed cells in the subdivision $S_{\boldsymbol{\omega}^C}$ of $C = (C_1, \ldots, C_n)$ induced by the lifting $\boldsymbol{\omega}^C$.

From what we have discussed above, the previously listed procedure for solving system $P(\mathbf{x}) = (p_1(\mathbf{x}), \ldots, p_n(\mathbf{x}))$ with $S = (S_1, \ldots, S_n)$ in $\mathbb{C}^n$ suggested in [43] may now be revised as follows:

Step 0: Let $d = \max_{1 \le i \le n} \deg p_i(x)$. Choose a real number $k > n(n+1)d^n$ at random.

Step 1: Lift the support $S' = (S'_1, \ldots, S'_n)$ by a random lifting $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_n)$ as defined in (10.4) where for $j = 1, \ldots, n$,

$$\begin{aligned} &\omega_j(\mathbf{0}) = k \quad \text{if } \mathbf{0} \notin S_j, \\ &\omega_j(\mathbf{a}) = \text{a randomly chosen number in (0,1)} \quad \text{if } \mathbf{a} \in S_j. \end{aligned}$$

Find cells of type $(1, \ldots, 1)$ in the induced fine mixed subdivision $S_{\boldsymbol{\omega}}$ of $S' = (S'_1, \ldots, S'_n)$.

Step 2: For cell $D = (\{\mathbf{a}_{10}, \mathbf{a}_{11}\}, \ldots, \{\mathbf{a}_{n0}, \mathbf{a}_{n1}\})$ of type $(1, \ldots, 1)$ in $S_{\boldsymbol{\omega}}$, let $\hat{\mathbf{a}}_{ji}(k) = (\mathbf{a}_{ji}, l)$ where for $j = 1, \ldots, n$, and $i = 0, 1$,

$$\begin{aligned} &l = k \quad \text{if } \mathbf{a}_{ji} = \mathbf{0} \notin S_j \\ &l = 0 \quad \text{if } \mathbf{a}_{ji} \in S_j. \end{aligned}$$

Form the $n \times (n+1)$ matrix

$$V = \begin{bmatrix} \hat{\mathbf{a}}_{11}^\top(k) - \hat{\mathbf{a}}_{10}^\top(k) \\ \vdots \\ \hat{\mathbf{a}}_{n1}^\top(k) - \hat{\mathbf{a}}_{n0}^\top(k) \end{bmatrix},$$

and find the unique vector $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n)$ where $\hat{\boldsymbol{\alpha}} = (\boldsymbol{\alpha}, 1)$ is in the kernel of V. This $\boldsymbol{\alpha}$ is the inner normal of D with respect to $\boldsymbol{\omega}_0^k$. Let

$T(\boldsymbol{\alpha})$ be the collection of all cells of type $(1,\ldots,1)$ in $S_{\boldsymbol{\omega}}$ with the same nonnegative inner normal $\boldsymbol{\alpha}=(\alpha_1,\ldots,\alpha_n)$ with respect to $\boldsymbol{\omega}_0^k$.

Step 3: (**a**) Choose any mixed cell $D=(\{\mathbf{a}_{10},\mathbf{a}_{11}\},\ldots,\{\mathbf{a}_{n0},\mathbf{a}_{n1}\})$ from $\mathcal{T}(\boldsymbol{\alpha})$, let

$$C_j=\{\mathbf{a}\in S'_j \mid \langle\hat{\mathbf{a}}(k),\hat{\boldsymbol{\alpha}}\rangle=\langle\hat{\mathbf{a}}_{j0}(k),\hat{\boldsymbol{\alpha}}\rangle\},\quad j=1,\ldots,n,$$

where $\hat{\mathbf{a}}(k)=(\mathbf{a},l)$ with

$$\begin{array}{ll} l=k & \text{if } \mathbf{a}=\mathbf{0}\notin S_j\\ l=0 & \text{if } \mathbf{a}\in S_j.\end{array}$$

Then $C=(C_1,\ldots,C_n)$ is a stable mixed cell of $S'=(S'_1,\ldots,S'_n)$ with respect to the inner normal $\boldsymbol{\alpha}$ in $S_{\boldsymbol{\omega}_0^k}$. Notice that

$$S_{\boldsymbol{\omega}^C}=\{(D_1,\ldots,D_n)\in S_{\boldsymbol{\omega}} \mid D_j\subseteq C_j \text{ for all } 1\le j\le n\}$$

is the fine mixed subdivision of C induced by $\boldsymbol{\omega}^C$, the restriction of $\boldsymbol{\omega}$ on C, and $\mathcal{T}(\boldsymbol{\alpha})$ provides all the mixed cells of type $(1,\ldots,1)$ in $S_{\boldsymbol{\omega}^C}$.

(**b**) Find all the isolated zeros of the system

$$P^{\alpha}(x)=(p_1^{\boldsymbol{\alpha}}(\mathbf{x}),\ldots,p_n^{\boldsymbol{\alpha}}(\mathbf{x}),\tag{10.5}$$

where

$$p_j^{\boldsymbol{\alpha}}(\mathbf{x})=\sum_{\boldsymbol{\alpha}\in C_j\cap S_j} c_{j,\mathbf{a}}\mathbf{x}^{\mathbf{a}}+\epsilon_j,\quad j=1,\ldots,n,$$

and

$$\begin{array}{ll} \epsilon_j=0 & \text{if } \mathbf{0}\in S_j,\\ \epsilon_j=1 & \text{if } \mathbf{0}\notin S_j\end{array}$$

in $(\mathbb{C}^*)^n$ by employing the polyhedral homotopy procedure with lifting $\boldsymbol{\omega}^C$.

(**c**) For zeros $\mathbf{e}=(e_1,\ldots,e_n)$ of $P^{\boldsymbol{\alpha}}(\mathbf{x})$ found in (**b**), let

$$\begin{array}{ll} \bar{e}_j=e_i & \text{if } \alpha_j=0,\\ \bar{e}_j=0 & \text{if } \alpha_j\neq 0.\end{array}$$

Then $\bar{\mathbf{e}}=(\bar{e}_1,\ldots,\bar{e}_n)$ is a zero of $P(\mathbf{x})$ in $\mathbb{C}^n$.

Step 4: Repeat Step 3 for all $\mathcal{T}(\boldsymbol{\alpha})$ with $\boldsymbol{\alpha}\ge\mathbf{0}$.

Remark 10.3. For $d_j = \deg p_j(\mathbf{x})$, $j = 1, \ldots, n$, we may assume, without loss, $d_1 \leq d_2 \leq \cdots \leq d_n$. It was mentioned in [31], in Step 0 of the above procedure, d may be replaced by $d_2 \times \cdots \times d_n \times d_n$ which usually results in a much smaller number.

Remark 10.4. It is commonly known that when the polyhedral homotopy method is used to solve polynomial systems, large differences between the powers of parameter t in the polyhedral homotopies may cause computational instability when homotopy curves are followed. In the algorithm above, the point $\mathbf{0}$ often receives very large lifting value k, compared to the rest of the lifting values in $(0, 1)$. It was shown in [31] that the stability of the algorithm is independent of the large lifting value k when polyhedral homotopies are used in Step 3($\mathbf{b}$).

The revised procedure listed above has been successfully implemented in [31] with remarkable numerical results.

11. Solving nonsquare systems polynomial system by randomization technique

By this time our discussions have been restricted to the "square" polynomial systems where the number of variables and equations are the same. However, nonsquare polynomial systems of the form

$$P(\mathbf{x}) = \begin{cases} p_1(x_1, \ldots, x_n) = 0 \\ \qquad \vdots \\ p_m(x_1, \ldots, x_n) = 0 \end{cases}$$

where $m \neq n$ arise naturally in many applications. If $m > n$, then the number of equations is greater than the number of variables, and the system is said to be *overdetermined*. If $m < n$, the number of equations is less than the number of variables, the system is said to be *underdetermined*.

For an underdetermined system, it is commonly known that the solutions of the system, if exist, cannot be isolated. The study of such "nonisolated" (a.k.a. positive dimensional) solutions is the main subject of §12. In this section, we shall only focus on overdetermined systems. In particular, one of our main goals is to find isolated solutions, if exist, of an overdetermined system. Among a range of different techniques, the "randomization" technique, introduced in [102], fits nicely in the "probability one" framework of the homotopy continuation methods. A comprehensive list of the variations on this scheme can be found in [103].

Consider the overdetermined polynomial system $P(x_1, \ldots, x_n) = (p_1(x_1, \ldots, x_n), \ldots, p_m(x_1, \ldots, x_n))$ as a column vector of m entries. A formal product of this column vector with an $n \times m$ matrix $A = (a_{ij})$ with full row rank ($\operatorname{rank} A = n$)

$$A \cdot P(\mathbf{x}) = \begin{bmatrix} a_{11} & \cdots & a_{1m} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nm} \end{bmatrix} \begin{bmatrix} p_1 \\ \vdots \\ p_m \end{bmatrix} = \begin{bmatrix} a_{11}p_1 + \cdots + a_{1m}p_m \\ \vdots \\ a_{n1}p_1 + \cdots + a_{nm}p_m \end{bmatrix}$$

results in a vector of n entries, giving rise to a new square system of equations

$$(A \cdot P)(\mathbf{x}) := \begin{cases} a_{11}p_1(\mathbf{x}) + \cdots + a_{1m}p_m(\mathbf{x}) \\ \vdots \\ a_{n1}p_1(\mathbf{x}) + \cdots + a_{nm}p_m(\mathbf{x}) \end{cases}$$

known as a **randomization** of $P(\mathbf{x})$.

Obviously, every solution of $P(\mathbf{x}) = \mathbf{0}$ is also a solution of $A \cdot P(\mathbf{x}) = \mathbf{0}$. Moreover, it can be shown that for generic choice of the matrix A, every isolated nonsingular solution of the original system $P(\mathbf{x}) = \mathbf{0}$ is an isolated nonsingular solution of the randomization system $A \cdot P(\mathbf{x}) = \mathbf{0}$. One can therefore solve the square system $A \cdot P(\mathbf{x}) = \mathbf{0}$ by using the homotopy continuation methods discussed in the previous sections and locate all the isolated nonsingular solutions of the original system.

However, it is possible that the randomization induces extraneous solutions. In particular, any $\mathbf{x} \in \mathbb{C}^n$ for which $P(\mathbf{x}) \in \operatorname{Ker} A \setminus \{\mathbf{0}\}$ would be a solution of the randomized system $A \cdot P(\mathbf{x}) = \mathbf{0}$ but not the original system $P(\mathbf{x}) = \mathbf{0}$. The technique of randomization therefore transforms an overdetermined system into a square system at the cost of introducing extraneous solutions. These extraneous solutions can generally be filtered out easily. Thus the handling of overdetermined system can be summarized as the following simple procedures:

Step 1: Choose a random (full row rank) $n \times m$ matrix and form the square system $A \cdot P(\mathbf{x})$ of n equations in n variables.

Step 2: Solve the randomization system $(A \cdot P)(\mathbf{x}) = \mathbf{0}$ and collect all the isolated nonsingular solutions.

Step 3: Filter out the extraneous solutions by checking if $P(\mathbf{x}) = \mathbf{0}$ for each isolated nonsingular solution $\mathbf{x}$ obtained in Step 2.

12. Positive dimensional solutions

A solution $\mathbf{x} = \hat{\mathbf{x}} \in \mathbb{C}^n$ of a polynomial system $P(\mathbf{x}) = (p_1(\mathbf{x}), \ldots, p_m(\mathbf{x})) = \mathbf{0}$ is said to be *isolated* if there is a neighborhood of $\hat{\mathbf{x}}$ in which $\mathbf{x} = \hat{\mathbf{x}}$ is the only solution of $P(\mathbf{x}) = \mathbf{0}$. The collection of all nonisolated solutions is known as the *positive dimensional* solution set of $P(\mathbf{x}) = \mathbf{0}$. Actually, our discussions on solving polynomial systems have only focused on finding isolated solutions by this time.

The existence of positive dimensional solution set of $P(\mathbf{x}) = \mathbf{0}$ is a common occurrence in application. Sometimes they are unpleasant side shows [102] that happen with a system generated using a model for which only the isolated regular solutions are of interest; and sometimes, the positive dimensional solution set is of primary interest. In either case, dealing with positive dimensional solution set is usually computationally difficult. Initiated in [102], the computation of positive dimensional solution set via homotopy methods has been developed into a rich and active field known as *Numerical Algebraic Geometry*. This section provides an overview of certain basic concepts and techniques in Numerical Algebraic Geometry, but defer to references for the technical detail. Readers are encouraged to consult the books [103] and [8].

The solution set of $P(\mathbf{x}) = (p_1(\mathbf{x}), \ldots, p_m(\mathbf{x})) = \mathbf{0}$ in $\mathbb{C}^n$ is known as an *algebraic set*. An algebraic set can always be decomposed into a finite union of *irreducible components* which are the algebraic sets that cannot be further decomposed into nontrivial unions of other algebraic sets. Each irreducible component has its well defined *dimension* and *degree*.[3] We delay the slightly more technical explanation of "degree" to §12.1. The concept of "dimension", leaving aside the formal definition, has an intuitive meaning: It can be shown that near almost all points in an irreducible component, the component locally looks like a neighborhood of a smooth manifold of a fixed dimension.This dimension is the dimension of the component, and the collection of all such points is known as the *smooth part* of the component. To understand the decomposition as well as the dimension and degree of each component via numerical methods, especially homotopy-based, is the main object of this section.

[3]From the point of view of modern algebraic geometric where algebraic sets are treat as geometric objects independent from equations describing them, the degree of an irreducible component of an algebraic set is not an intrinsic invariant; rather, it is a property of how the set is embedded in $\mathbb{C}^n$ (or projective spaces). However, in applications, the degree is often a useful piece of information in describing and classifying the algebraic sets.

At first, one must be reminded of the meaning of "reduced" solution components of $P(\mathbf{x}) = \mathbf{0}$. In the case of an isolated solution, the "nonreducedness" manifests itself as its multiplicity structure. For positive dimensional solution sets, the situation is much more complex. For example, the two equations $xy = 0$ and $xy^2 = 0$ clearly define the same algebraic set in $\mathbb{C}^2$, *i.e.*, the union of two axes $x = 0$ and $y = 0$. However, this set is a *nonreduced* solution set of the equation $xy^2 = 0$ as the x-axis, in a naive sense, should have "multiplicity" 2 as a solution component. By and large, *reduced* is synonymous with *multiplicity-one*, while *nonreduced* implies a multiplicity greater than one [39, 103]. From the standpoint of numerical computation, the "nonreducedness" will significantly complicate the environment. For simplicity, this section will focus solely on "generically reduced" positive dimensional solution sets which are solution sets that locally look like a reduced one near all but a nowhere dense closed subset.

The discussion in this section is guided by three main questions:

- First of all, on a global scale, the question is whether there exists a positive dimensional solution set of $P(\mathbf{x}) = \mathbf{0}$? If there is, can one obtain some sample points from each component? These are discussed in §12.1.
- Secondly, given a solution $\mathbf{x} = \hat{\mathbf{x}} \in \mathbb{C}^n$ of the polynomial system $P(\mathbf{x}) = \mathbf{0}$, is $\hat{\mathbf{x}}$ an isolated solution? or does it belong to a positive dimensional solution component? This question is the subject of §12.2.
- Finally, can the global structure of the solution set of a polynomial system (ignoring the nonreduced structure) be studied via "numerical irreducible decomposition"? This will be discussed in §12.3.

Interestingly, the development of techniques for detecting and studying positive dimensional solution sets has contributed new ideas to the problem of finding isolated solutions. Of the particular importance are the *Diagonal Homotopy* [101] and the *Regeneration Homotopy* [40] which can be used for finding isolated solutions.

12.1. Global sampling via linear slicing

The main subject of this section is the "global sampling" of an algebraic set; meaning, we wish to get at least one sample point on each irreducible component. An important technique in this regard is the "linear slicing" developed in [102]. This technique has now become one of the basic building blocks of the emerging subject *Numerical Algebraic Geometry*. Geometrically, it is

the procedure of intersecting an irreducible component with an affine space of complimentary dimension, yielding isolated intersection points as sample points of the component.

One of the theoretical results supports this technique is the Noether's Normalization Lemma [39, 95]. It essentially states that any d-dimensional irreducible algebraic set in $\mathbb{C}^n$ can be realized as a "finite branched cover" over $\mathbb{C}^d$ via a linear projection.

Restricted to curves (1-dimensional algebraic set), this is a familiar technique: When studying a space curve in $\mathbb{C}^3$, for instance, it is common to project the curve to one of the coordinates axis, say x-axis via $\pi(x, y, z) = x$. This projection is a "finite-to-one" surjective map, or more precisely, $\pi^{-1}(x)$ consists of finitely many points on the curve for all x. More abstractly, via the projection π, the curve is realized as a *finite branched cover* over $\mathbb{C}$. In this setup, by fixing an x value and solving the equation $\pi(\mathbf{p}) = x$ for points $\mathbf{p}$ on the curve, one obtains a "cross-section" of the curve. Indeed, for almost all choices of x, the number of complex points in $\pi^{-1}(x)$ is a fixed number, called the *degree* of the curve.

Clearly, this procedure would fail if the entire curve is contained in a plane perpendicular to the x-axis, as the projection would map the entire curve to a single point on the x-axis while the fiber over any other point would be empty. One important consequence of the Noether's Normalization Lemma is that this can happen via a projection onto *some* one-dimensional subspace of $\mathbb{C}^3$. In fact, with projections $\pi : \mathbb{C}^3 \to \mathbb{C}$ of the form

$$\pi(x, y, z) = a_1 x + a_2 y + a_3 z$$

where $a_1, a_2, a_3 \in \mathbb{C}$, one can show that for generic choices of a_1, a_2, a_3, the map π, when restricted to the curve in question is a surjective finite-to-one map.

This procedure can be generalized to higher dimension. For a d-dimensional irreducible algebraic set $X \subset \mathbb{C}^n$, we shall take similar linear projections of X to a d-dimensional subspace: For a $d \times n$ matrix $A = (a_{ij})$ of full row rank, let the linear map $\pi : \mathbb{C}^n \to \mathbb{C}^d$ be given by

$$\pi(\mathbf{x}) = A\mathbf{x}. \tag{12.1}$$

We are interested in the restriction $\pi|_X : X \to \mathbb{C}^d$ which projects X to $\mathbb{C}^d$. For a point $\mathbf{b} \in \mathbb{C}^d$, $(\pi|_X)^{-1}(\mathbf{b})$ is called a *linear slicing* of X.

By the Noether's Normalization Lemma, for generic choices of the matrix A and a point $\mathbf{b} \in \mathbb{C}^d$, $(\pi|_X)^{-1}(\mathbf{b})$ consists of a finite number of points in the smooth part X_{reg} of X. Moreover, for almost all $\mathbf{b} \in \mathbb{C}^d$, the number of

points in $(\pi|_X)^{-1}(\mathbf{b})$ is a fixed number, called the *degree* of X, denoted by $\deg X$. In such occasions, the linear slicing $(\pi|_X)^{-1}(\mathbf{b})$ is said to be *generic with respect to* X. Notice that $(\pi|_X)^{-1}(\mathbf{b})$ is the set of intersection points between X and the $(n-d)$-dimensional affine space defined by $A\mathbf{x} = \mathbf{b}$, and is called a *generic linear slicing* of X.

Algebraically, if X is an irreducible component of the solution set of the polynomial system $P(\mathbf{x}) = \mathbf{0}$, then $(\pi|_X)^{-1}(\mathbf{b})$ is a subset of the solution set of the augmented system $(P(\mathbf{x}), A\mathbf{x} - \mathbf{b})$, that is,

$$\left\{\begin{array}{r} p_1(x_1,\ldots,x_n) = 0 \\ \vdots \\ p_m(x_1,\ldots,x_n) = 0 \\ a_{11}x_1 + \cdots + a_{1n}x_n = b_1 \\ \\ \vdots \\ a_{d1}x_1 + \cdots + a_{dn}x_n = b_d. \end{array}\right. \tag{12.2}$$

Since the generic linear slicing intersects each irreducible d-dimensional component of the zero set of $P(\mathbf{x}) = \mathbf{0}$, among the zero-dimensional solution set of (12.2), there is at least one point on each d-dimensional irreducible component. It is, of course, possible for the system (12.2) to have positive dimensional solution set. But for generic choice of A and $\mathbf{b}$, these point will not belong to any d-dimensional irreducible components. Therefore, after solving (12.2) via homotopy methods described in the previous sections, local dimension tests (§12.2) must be performed to filter out all nonisolated solutions.

This procedure essentially provides the solution to the global sampling problem: to find at least one sample point on each irreducible component of a given dimension, and it can be summarized as the following steps:

Step 1: For the given target dimension d, pick a random $d \times n$ matrix $A \in M_{d\times n}(\mathbb{C})$ and a random vector $\mathbf{b} \in \mathbb{C}^d$. With these, one constructs the augmented "linear slicing" system (12.2).

Step 2: Solve (12.2) via numerical homotopy methods, possibly in conjunction with randomization techniques discussed in §11 if (12.2) is not square.

Step 3: Apply local dimension test techniques to the solutions of (12.2) obtained and filter out nonisolated solutions. The remaining solutions

contain at least one point on each irreducible d-dimensional component of the solution set of the original system $P(\mathbf{x}) = \mathbf{0}$.

Additional sample points can be generated by moving the linear slicing. That is, one could consider the one-parameter family of linear slicings given by $A_t\,\mathbf{x} = \mathbf{b}_t$ where A_0 and $\mathbf{b}_0$ represents the linear slicing in (12.2) and the homotopy

$$H(\mathbf{x}, t) := \begin{cases} P(\mathbf{x}) \\ A_t\,\mathbf{x} - \mathbf{b}_t. \end{cases} \tag{12.3}$$

Then the sample points obtained by solving (12.2) are isolated nonsingular solutions of $H(\mathbf{x}, 0) = \mathbf{0}$ and can thus be used as the starting points of the homotopy paths of this homotopy. As long as the linear slicing represented by A_t and $\mathbf{b}_t$ remains generic with respect to the irreducible components for each $t \in [0, 1]$, path tracking algorithms discussed in previous sections can then be used to generate additional sample points. This technique of "moving linear slicing" is also used for other purposes. §12.3 describes its use in Numerical Irreducible Decomposition.

12.2. Local dimension test

Let $P(\mathbf{x}) = (p_1(\mathbf{x}), \ldots, p_m(\mathbf{x}))$ be a system of m polynomial equations in the n unknowns $\mathbf{x} = (x_1, \ldots, x_n)$. For simplicity, we assume $m \geq n$. When a numerical solution $\mathbf{x}_0$ of $P(\mathbf{x}) = \mathbf{0}$ is obtained, we want to determine whether $\mathbf{x}_0$ is an isolated solution of $P(\mathbf{x}) = \mathbf{0}$ in the first place.

In theory, $\mathbf{x}_0$ distinguishes itself as an isolated nonsingular solution of $P(\mathbf{x}) = \mathbf{0}$ when none of the singular values of the Jacobian of $P(\mathbf{x})$, denoted by $DP(\mathbf{x})$, at $\mathbf{x}_0$ vanish. In practical computation, one would commonly admit $\mathbf{x}_0$ as an isolated nonsingular solution if the smallest singular value of $DP(\mathbf{x}_0)$ is not too *small*. When $DP(\mathbf{x}_0)$ allows very small singular values, $\mathbf{x}_0$ may lie on a solution component of $P(\mathbf{x}) = \mathbf{0}$ of positive dimension or it may still be an isolated solution with multiplicity no less than 2. In this subsection, we give a brief discussion of a method that is capable of differentiating those cases. More generally, the method determines the dimension of the solution component $\mathcal{X}$ of $P(\mathbf{x}) = \mathbf{0}$ to which $\mathbf{x}_0$ belongs. When $\dim \mathcal{X} = 0$, $\mathbf{x}_0$ is of course an isolated solution of $P(\mathbf{x}) = \mathbf{0}$.

The main strategy of the method can be briefly described as follows. As noted above, when the Jacobian $DP(\mathbf{x}_0)$ has no small singular values, then $\mathbf{x}_0$ can be classified as an isolated nonsingular solution. If it permits only

one singular value that appears very small and if $\mathbf{x}_0$ is not geometrically isolated, then x_0 must lie on a one dimensional solution path of $P(\mathbf{x}) = \mathbf{0}$. We will begin to trace this path to a substantial length by a special designed path following scheme. If this attempt fails, no such solution path may exist and $\mathbf{x}_0$ will be classified as an isolated solution of $P(\mathbf{x}) = \mathbf{0}$. When $DP(\mathbf{x}_0)$ has $k > 1$ very small singular values, we augment $P(\mathbf{x}) = \mathbf{0}$ with $k-1$ generic hyperplanes $\mathbf{a}_i^H(\mathbf{x} - \mathbf{x}_0) = 0$, $i = 1, \ldots, k-1$, at $\mathbf{x}_0$. It follows from the *linear slicing* elaborated in the last section, with minor adjustment, the enlarged system

$$\bar{P}(\mathbf{x}) = \begin{cases} P(\mathbf{x}) = \mathbf{0} \\ \mathbf{a}_1^H(\mathbf{x} - \mathbf{x}_0) = 0 \\ \qquad \vdots \\ \mathbf{a}_{k-1}^H(\mathbf{x} - \mathbf{x}_0) = 0 \end{cases} \tag{12.4}$$

will produce a one dimensional solution component $\bar{\mathcal{X}}$ of $\bar{P}(\mathbf{x}) = \mathbf{0}$ at $\mathbf{x}_0$ if the solution component $\mathcal{X}$ of $P(\mathbf{x}) = \mathbf{0}$ to which $\mathbf{x}_0$ belongs is of dimension k. Thus the assertion $\dim \mathcal{X} = k$ is accurate only if we can identify $\bar{\mathcal{X}}$ by tracing $\bar{\mathcal{X}}$ to a satisfactory length. If this path following can not be carried out successfully, such component $\bar{\mathcal{X}}$ may not exist. We will then remove hyperplane $\mathbf{a}_{k-1}^H(\mathbf{x} - \mathbf{x}_0) = 0$ in (12.4) and restart our effort to identify the one dimensional solution component $\bar{\bar{\mathcal{X}}}$ produced by the system

$$\bar{\bar{P}}(\mathbf{x}) = \begin{cases} P(\mathbf{x}) = \mathbf{0} \\ \mathbf{a}_1^H(\mathbf{x} - x_0) = 0 \\ \qquad \vdots \\ \mathbf{a}_{k-2}^H(\mathbf{x} - \mathbf{x}_0) = 0. \end{cases} \tag{12.5}$$

The existence of such component $\bar{\bar{\mathcal{X}}}$ implies the solution component $\mathcal{X}$ of $P(\mathbf{x}) = \mathbf{0}$ is of dimension $k-1$. If it fails, the process may be continued in the same manner and the dimension of $\mathcal{X}$ will ultimately (very soon in practice) be determined. For details of the specially designed "*path following scheme*", please see [49].

The above algorithm is particularly valuable when the homotopy continuation method is used to solve polynomial systems. The homotopy method follows homotopy paths emanating from solutions of known systems and solutions of target system lie at the end of those paths. It was widely believed that it is non-generic for the repeated appearances of the same solution to occur on a solution component with a positive dimension. Consequently, a solution that repeats itself at the end of different paths will always be

taken as a multiple isolated solution if no apparent *curve jumpings* exist. It turns out the results of the algorithm show that this sort of repeated appearances actually happens in positive dimensional solution components frequently. For a simple example, the cyclic-4 problem [7, 107] has no isolated solutions. But, by polyhedral homotopy, all 16 (= the mixed volume of the system) homotopy paths converge to 8 particular solutions repeatedly, twice for each one, regardless of what starting systems were used, and our algorithm accurately determined the correct dimension of the solution components on each individual case.

Remark 12.1. An inevitable part of this method to determine the dimension of a solution component is the rank revealing of the Jacobian matrix $DP(\mathbf{x}_0)$ at the solution point $\mathbf{x}_0$. This can normally be achieved by computing the SVD (Singular Value Decomposition) of $DP(\mathbf{x}_0)$ and deciding which singular values are significantly nonzero. However, in our context the rank deficiency of $DP(\mathbf{x}_0)$ is usually small compared to its size. It is therefore unnecessary to compute the full SVD which is quite expensive. One could adopt the rank revealing technique developed in [56] which computes singular values and their associate singular vectors in ascending order from the smallest one. By which, it needs only compute those singular values which are smaller than the given threshold to determine the rank of $DP(\mathbf{x}_0)$.

12.3. Numerical irreducible decomposition via monodromy

As noted previously, the algebraic set $\mathcal{V}(P) = \{\mathbf{x} \in \mathbb{C}^n \mid P(\mathbf{x}) = \mathbf{0}\}$ can be decomposed into a finite union of *irreducible components* each having a well defined dimension. In other words, there is an *irreducible decomposition* of the form

$$\mathcal{V}(P) = \bigcup_{d=0}^{n} V_d = \bigcup_{d=0}^{n} \bigcup_{i \in I_d} V_{d,i} \tag{12.6}$$

where each V_d is the union of all d-dimensional components, each $V_{d,i}$ is an irreducible component of dimension d, and the index sets I_d are finite and possibly empty. Notice that the set of isolated zeros of P appears in the above decomposition as the zero-dimensional components V_0 with each point being a component. The main goal here is to find a *numerical irreducible decomposition* that mirrors the irreducible decomposition of (12.6).

Encapsulating this process is the foundation of the new subject *Numerical Algebraic Geometry*.

We shall first focus on the case of *pure dimensional* algebraic sets, say V_d, in which all components have the same dimension d. We wish to decompose V_d into the union of all d-dimensional components $V_d = \bigcup_{i \in I_d} V_{d,i}$ numerically, and expect, with the same process, other cases can be handled dimension-by-dimension.

Since positive dimensional components contain infinitely many solution points, a finite encoding of the components suitable for numerical computation is therefore required. The linear slicing discussed in §12.1 turns out to be a great way to construct such an encoding which will be called "witness sets".

In §12.1, to answer the global sampling problem (finding at least one point in each component), a generic linear slicing represented by $L(\mathbf{x}) = A\mathbf{x} - \mathbf{b}$ where $A \in M_{(n-d)\times n}$ and $\mathbf{b} \in \mathbb{C}^{n-d}$ are used to select generic sample points from the d-dimensional components. Let χ be the set of all isolated nonsingular points in the linear slicing $\mathcal{V}(P(\mathbf{x})) \cap \mathcal{V}(L(\mathbf{x}))$, then for generic choice of the slicing, $\chi \subset V_d$ and, among finitely many points in χ there are at least one point on each irreducible component of V_d. Moreover, the number of points of χ on each irreducible component is precisely the degree of the component. The data structure $W := (\chi, L, P)$ is called a *witness set* of V_d, and it is the numerical representation of V_d. Here, the witness set W carries extra information beyond the finite set χ. In particular, it depends on the systems L and P. Evidently for generic choices of L, each point in W belongs to precisely one irreducible component in $V_d = \bigcup_{i \in I_d} V_{d,i}$. To partition the points in W by the components to attain a grouping $W = \bigcup_{i \in I_d} W_i$ so that $W_i \subset V_{d,i}$, a basic tool is the *monodromy* technique developed in [99].

The main structure that enables this tool is the smooth part of each positive dimensional irreducible component is path connected while the smooth parts of different components are disjoint. The *monodromy* technique can be understood as a tool to achieve the partitioning of W via the construction of paths within the smooth parts of the components joining different witness points in the same components respectively.

For simplicity, consider a one parameter family of linear slicing $\hat{L}_z$ of V_d parametrized by a single complex variable z together with the family of witness set W_z it defines. $\hat{L}_z$ can be constructed so that for generic choices of $z \in \mathbb{C}$, $\hat{L}_z(\mathbf{x})$ represents a linear slicing generic with respect to V_d (as discussed in §12.1). That is, let $U \subseteq \mathbb{C}$ be the collection of all choices of z for which $\hat{L}_z$ is a linear slicing generic with respect to (the components of) V_d, then U is open and dense in $\mathbb{C}$. As z move continuously within U,

points in the corresponding witness sets W_z simply move accordingly while remaining on the smooth parts of the components. Outside U, however, there still can be a set of isolated "branch points": as z moves across a branch point, certain witness points in W_z may collide.

Central to the monodromy technique is the phenomenon known as *non-trivial monodromy action*: As z moves along a simple loop parametrized by $\gamma : [0,1] \to \mathbb{C}$ with $\gamma(0) = \gamma(1)$ that contains a branch point in its interior, the movement of a particular witness point in $W_{\gamma(0)}$ may trace out a path that reaches another witness point in the same witness set $W_{\gamma(0)} = W_{\gamma(1)}$.

Example 12.2. Consider a simple yet illuminating example of the equation $P(x,y) = x^2 - y = 0$ which defines a quadratic curve — a parabola — in $\mathbb{C}^2$. Using the linear slicing $L_z(x,y) = y - z = 0$ yields the combined system

$$\begin{aligned} x^2 - y &= 0 \\ y - z &= 0. \end{aligned}$$

Apparently, for any fixed value of $z \in \mathbb{C}$ and $z \neq 0$, there are precisely two witness points given by $(x,y) = (\pm\sqrt{z}, z)$ where $\sqrt{z}$ is any branch of the square root of z. As $z \to 0$, the two witness points would collide, making $z = 0$ a branch point.

The monodromy technique in this setting essentially lets z run in a loop around the branch point at $z = 0$. Consider, for example, a small circle of radius 1 centered at $z = 0$ which is parametrized by $z = \gamma(t) = e^{i\,2\pi t}$ with $\gamma(0) = \gamma(1) = 1$ (i.e., a closed loop). At $t = 0$, the two witness points, as defined by $(P, L_{\gamma(0)}) = (0,0)$ are $(x^{(1)}(0), y^{(1)}(0)) = (1,1)$ and $(x^{(2)}(0), y^{(2)}(0)) = (-1,1)$ respectively. As t varies from 0 to 1, the corresponding witness points trace out two smooth curves which can be expressed as

$$\begin{aligned} x^{(1)}(t) &= e^{i\pi t} & x^{(2)}(t) &= -e^{i\pi t} = e^{i\pi t + \pi} \\ y^{(1)}(t) &= e^{i2\pi t} & y^{(2)}(t) &= e^{i2\pi t}. \end{aligned}$$

An interesting phenomenon is that when t reaches 1, while the image of $\gamma(t)$ returns to its starting point closing the loop, the two curves traced out by the corresponding witness points do not return to their starting points. Indeed, $x^{(1)}(1) = x^{(2)}(0)$ and $x^{(2)}(1) = x^{(1)}(0)$. That is, as $\gamma(t)$ (and hence the linear slicing $L_{\gamma(t)}$) completes a circle around the branch point, the first witness point $(1,1)$ moves accordingly and reach the second witness point $(-1,1)$. Moreover, it is easy to check that for all $t \in [0,1]$, the solutions of

$(P, L_{\gamma(t)}) = (0, 0)$ are smooth points of $\mathcal{V}(P)$ and they form a smooth curve. This curve that connects the two witness points while staying inside the smooth part shows that they belong to the same irreducible component.

Described formally, let $W_0 = W_{\gamma(0)}$ which consists of isolated regular common zeros of $P(\mathbf{x})$ and $\hat{L}_{\gamma(0)}(\mathbf{x})$. For a witness point $\mathbf{x}^{(0)} \in W_0$, we consider the path $\mathbf{x}(t)$ where $\mathbf{x}(0) = \mathbf{x}^{(0)}$ and

$$\begin{aligned} P(\mathbf{x}(t)) &= \mathbf{0} \\ \hat{L}_{\gamma(t)}(\mathbf{x}(t)) &= \mathbf{0} \end{aligned}$$

for all $t \in [0, 1]$. Since $\gamma(0) = \gamma(1)$, the end point $\mathbf{x}(1)$ must be in $W_{\gamma(1)} = W_0$. Note that since $\gamma(t) \in U$ for all $t \in [0, 1]$, the path $\mathbf{x}(t)$ is contained in the smooth part of a irreducible component. However, if $\mathbf{x}(1) \neq \mathbf{x}(0)$ then the path $\mathbf{x}(t)$ connecting them (through the smooth part of a component) provides a definitive proof that the two witness points are inside the same irreducible component.

The monodromy technique can be summarized as the following numerically implementable steps centered around the path tracking:

Step 1: For a fixed dimension d, let W_0 be the witness set of V_d in (12.6) obtained by solving (12.2) using homotopy methods.

Step 2: A one-parameter family of linear slicings $\hat{L}_z(\mathbf{x}) : \mathbb{C}^n \to \mathbb{C}^d$, $z \in \mathbb{C}$ is constructed with the property that for a generic choice of z, $L(\mathbf{x}) := \hat{L}_z(\mathbf{x})$ defines a linear slicing that is generic with respect to the components of V_d. Within the set of "generic choices", a simple loop $\gamma : [0, 1] \to \mathbb{C}$ is chosen.

Step 3: Using witness points in W_0 as starting points, one tracks the solution paths defined by

$$H(\mathbf{x}, t) := \begin{cases} P(\mathbf{x}) = \mathbf{0} \\ \hat{L}_{\gamma(t)}(\mathbf{x}) = \mathbf{0} \end{cases}$$

for t from 0 to 1. The end points of these solution paths are necessarily inside the same witness set W_0. Most importantly, when a solution path joining two different witness points, then these two points must be in the same irreducible components.

Step 2 and 3 can be repeated with a different family of slicings and potentially reveal the connectedness (via solution paths) between other witness points. The *trace test* [100] can be used as a stopping criteria.

It is certainly possible for all solution paths tracked in Step 3 to come back to their respective starting point, that is, $\mathbf{x}(0) = \mathbf{x}(1)$ leading to a "trivial monodromy action". Such paths, of course, provide no information concerning the proper subdivision of the witness points. How to generate loops (and family of linear slicings) that will give rise to nontrivial monodromy actions (and produce useful information for grouping witness points) is still an open problem (See [8, 103]).

A simple yet useful choice of loop of linear slicings can be constructed based on a technique known as the *gamma-trick* [103]: For two linear slicings $L^{(0)}(\mathbf{x})$ and $L^{(1)}(\mathbf{x})$, one constructs the one parameter family

$$L_z(\mathbf{x}) = (1-z)L^{(0)}(\mathbf{x}) + z\,L^{(1)}(\mathbf{x}). \tag{12.7}$$

Within this family, we choose the loop $\gamma : [0,1] \to \mathbb{C}$

$$z = \gamma(t) = \begin{cases} 2t & \text{for } 0 \le t \le 1/2 \\ \dfrac{2e^{i\theta}(t-1)}{2(e^{i\theta}-1)(t-1)-1} & \text{for } t > 1/2 \end{cases} \tag{12.8}$$

where $\theta \neq 0$ is a randomly chosen real number.

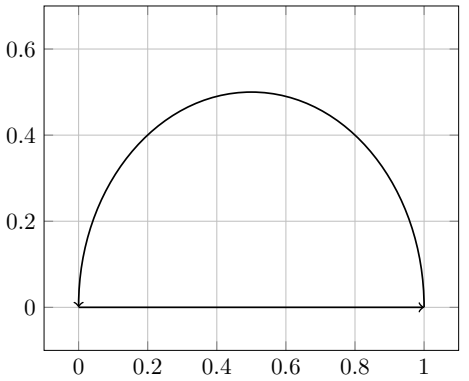


Figure 25. The loop $\gamma(t)$ within the one parameter family of linear slicings

Clearly, $L_{\gamma(t)}$ form a closed loop in the space of linear slicings as t varies from 0 to 1. Despite the somewhat complicated formula, the loop is geometrically simple: On the first leg of the trip when $t \in [0, 1/2]$, $\gamma(t)$ simply moves along the real axis on the complex plane from 0 to 1. The remainder of the loop takes the form of a Mobius transformation $\frac{2e^{i\theta}(t-1)}{2(e^{i\theta}-1)(t-1)-1}$ that maps the line segment $[1/2, 1]$ on the complex plane to an arc, as depicted in

Figure 25. Since $L_z(\mathbf{x})$ in (12.7) is linear in z, after substituting $z(t)$ in (12.8) into L_z, one can clear the denominators and obtain the simpler formulation

$$L_t(\mathbf{x}) = \begin{cases} (1-2t)L^{(0)}(\mathbf{x}) + 2tL^{(1)}(\mathbf{x}) & \text{for } 0 \le t \le 1/2 \\ (2t-1)L^{(0)}(\mathbf{x}) + 2e^{i\theta}(1-t)L^{(1)}(\mathbf{x}) & \text{for } t > 1/2. \end{cases}$$

Extensive experiences within the community of Numerical Algebraic Geometry seem to suggest that this construction is sufficient in many situations for discovering all the nontrivial monodromy actions necessary for numerical irreducible decompositions.

13. Positive dimensional $\mathbb{C}^*$-solution sets of systems of binomial equations

The above discussion summarized the general techniques for studying positive dimensional solution sets of polynomial systems. This section highlights special techniques for computing the positive dimensional solution set of a subclass of polynomial systems — binomial systems. The reason behind singling out binomial systems is threefold: First, the binomial systems arise naturally in many applications and theoretical studies (e.g. in the context of toric varieties). Second, specialized techniques allow much efficient handling of large binomial systems. Finally, the description of these techniques unites several seemingly unrelated topics discussed in previous sections which highlights the integral nature of this subject.

In §7 we have provided tools for finding isolated solutions of a square binomial equations in which the number of equations matches the number of variables in $(\mathbb{C}^*)^n$. This section will elaborate the techniques for studying the positive dimensional solution set of a system of binomial equations that may or may not be a square system.

13.1. Structure of positive dimensional $\mathbb{C}^*$-solution sets of Laurent binomial systems

We shall reuse many of the notations and concepts in §7: For positive integers m and n, $M_{n\times m}(\mathbb{Z})$ denotes the set of all $n \times m$ matrices with integer entries. A square integer matrix is said to be **unimodular** if its determinant is ± 1. Note that such a matrix $A \in M_{n\times n}(\mathbb{Z})$ has a unique inverse $A^{-1} = \frac{1}{\det A}\operatorname{adj} A$ which is also in $M_{n\times n}(\mathbb{Z})$, where adj A is the adjugate matrix of A. The $n \times n$ identity matrix in $M_{n\times n}(\mathbb{Z})$ is denoted by I_n as usual. Just like in §7, the theory of binomial systems is more naturally developed in the context

of the generalized "Laurent binomial systems" where negative exponents are allowed. For variables $\mathbf{x} = (x_1, \ldots, x_n)$, a **Laurent monomial** in $\mathbf{x}$ is an expression of the form $x_1^{\alpha_1} \cdots x_n^{\alpha_n}$ where $\alpha_1, \ldots, \alpha_n$ are integers (which may be zero or negative). For a vector $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n)^\top \in \mathbb{Z}^n$, we still use the notation

$$\mathbf{x}^{\boldsymbol{\alpha}} := (x_1, \ldots, x_n) \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_n \end{bmatrix} = x_1^{\alpha_1} \cdots x_n^{\alpha_n}.$$

As before, for a matrix $A \in M_{n\times m}(\mathbb{Z})$ with columns $\boldsymbol{\alpha}^{(1)}, \ldots, \boldsymbol{\alpha}^{(m)} \in \mathbb{Z}^n$,

$$\mathbf{x}^A = \mathbf{x}^{[\,\boldsymbol{\alpha}^{(1)} \,\cdots\, \boldsymbol{\alpha}^{(m)}\,]} := (\mathbf{x}^{\boldsymbol{\alpha}^{(1)}}, \ldots, \mathbf{x}^{\boldsymbol{\alpha}^{(m)}}). \tag{13.1}$$

With this notation, the familiar identities $\mathbf{x}^{I_n} = \mathbf{x}$ and $(\mathbf{x}^A)^B = \mathbf{x}^{AB}$ are still valid. Since the exponents here may be negative, it is only meaningful to deal with the function $\mathbf{x} \mapsto \mathbf{x}^A$ when each x_i is restricted to be nonzero. Hence, throughout this section, we shall let $x_i \in \mathbb{C}^*$ for each $i = 1, \ldots, n$. In this situation, each matrix $A \in M_{n\times m}(\mathbb{Z})$ induces a function from $(\mathbb{C}^*)^n$ to $(\mathbb{C}^*)^m$ via $\mathbf{x} \mapsto \mathbf{x}^A$. Of particular importance is the function induced by a unimodular matrix $A \in M_{n\times n}(\mathbb{Z})$, since, in this case, A^{-1} is also in $M_{n\times n}(\mathbb{Z})$. Thus functions $\mathbf{x} \mapsto \mathbf{x}^A$ and $\mathbf{x} \mapsto \mathbf{x}^{A^{-1}}$ are inverses to each other ($(\mathbf{x}^A)^{A^{-1}} = \mathbf{x}^{A\,A^{-1}} = \mathbf{x}^{I_n} = \mathbf{x}$).

A **Laurent binomial** is an expression of the form $c_1\mathbf{x}^{\boldsymbol{\alpha}} + c_2\mathbf{x}^{\boldsymbol{\beta}}$ for some $c_1, c_2 \in \mathbb{C}^*$ and $\boldsymbol{\alpha}, \boldsymbol{\beta} \in \mathbb{Z}^n$. It is just a linear combination of *two* Laurent monomials. This section will focus on the structure of positive dimensional solution set of a system of Laurent binomial equations, or simply **Laurent binomial systems**. We shall restrict our attention to the portion of the solution set inside $(\mathbb{C}^*)^n$ which, in a sense, is the natural setting for studying binomial systems. That is, given exponent vectors $\boldsymbol{\alpha}^{(1)}, \ldots, \boldsymbol{\alpha}^{(m)}, \boldsymbol{\beta}^{(1)}, \ldots, \boldsymbol{\beta}^{(m)} \in \mathbb{Z}^n$ and the coefficients $c_{i,j} \in \mathbb{C}^*$, our goal is to describe the set of all $\mathbf{x} \in (\mathbb{C}^*)^n$ that satisfy the system of equations

$$\begin{cases} c_{1,1}\mathbf{x}^{\boldsymbol{\alpha}^{(1)}} + c_{1,2}\mathbf{x}^{\boldsymbol{\beta}^{(1)}} = 0 \\ \qquad\qquad\qquad\qquad\quad \vdots \\ c_{m,1}\mathbf{x}^{\boldsymbol{\alpha}^{(m)}} + c_{m,2}\mathbf{x}^{\boldsymbol{\beta}^{(m)}} = 0. \end{cases}$$

Since only the solutions in $(\mathbb{C}^*)^n$ are in concern, this system is clearly equivalent to

$$(\mathbf{x}^{\boldsymbol{\alpha}^{(1)}-\boldsymbol{\beta}^{(1)}}, \ldots, \mathbf{x}^{\boldsymbol{\alpha}^{(m)}-\boldsymbol{\beta}^{(m)}}) = (-c_{1,2}/c_{1,1}, \ldots, -c_{m,2}/c_{m,1})$$

which can simply be written as

$$\mathbf{x}^A = \mathbf{b} \quad \text{or equivalently} \quad \mathbf{x}^A - \mathbf{b} = \mathbf{0} \tag{13.2}$$

where $A \in M_{n\times m}(\mathbb{Z})$, having columns $\boldsymbol{\alpha}^{(1)} - \boldsymbol{\beta}^{(1)}, \ldots, \boldsymbol{\alpha}^{(m)} - \boldsymbol{\beta}^{(m)}$, represents the exponents appeared in the Laurent monomials and the vector $\mathbf{b} = (-c_{1,2}/c_{1,1}, \ldots, -c_{m,2}/c_{m,1})$ collects all the coefficients. Its solution set in $(\mathbb{C}^*)^n$ will be denoted by

$$\mathcal{V}^*(\mathbf{x}^A - \mathbf{b}) := \{\, \mathbf{x} \in (\mathbb{C}^*)^n \;\mid\; \mathbf{x}^A - \mathbf{b} = \mathbf{0} \,\}. \tag{13.3}$$

In §7, we discussed a special case of this setup where A is a nonsingular square matrix and $\mathbf{x}^A - \mathbf{b} = \mathbf{0}$ is a system of n Laurent binomial equations in n variables. In the following we shall deal with the general cases.

In the first place, we briefly review some basic facts about the $\mathbb{C}^*$-solution set of a Laurent binomial system which usually fall under the subject of *toric algebraic geometry* and *combinatorial commutative algebra*. For more details, we refer to standard references such as [23, 26, 27, 75, 105] for comprehensive discussions of the theoretical aspects. Certain computational aspects have been studied in [45, 46]. Recent developments in the aspect of parallel numerical computation have been presented in the article [17, 20].

In §7, the *Hermite Normal Form* of the matrix A has been used to solve for the isolated solutions of a square binomial system in $(\mathbb{C}^*)^n$. To understand the structure of the positive dimensional solution set of a general Laurent binomial system, a stronger form known as the *Smith Normal Form* is needed. It is known that there are unimodular square matrices $P \in M_{n\times n}(\mathbb{Z})$ and $Q \in M_{m\times m}(\mathbb{Z})$ such that

$$P\,A\,Q \;=\; \begin{bmatrix} \overbrace{\begin{matrix} d_1 & & \\ & \ddots & \\ & & d_r \end{matrix}}^{r} & \overbrace{\begin{matrix} & & \\ & & \\ & & \end{matrix}}^{m-r} \\ \begin{matrix} 0 & & \\ & \ddots & \\ & & 0 \end{matrix} & \end{bmatrix} \begin{matrix} \left.\begin{matrix} \\ \\ \\ \end{matrix}\right\} r \\ \left.\begin{matrix} \\ \\ \\ \end{matrix}\right\} n-r \end{matrix} \tag{13.4}$$

with nonzero integers $d_1 \,|\, d_2 \,|\, \cdots \,|\, d_r$ for $r = \operatorname{rank} A$, unique up to the signs. Here, $a \,|\, b$ means a divides b as usual. The matrix on the right hand side of (13.4) is the Smith Normal Form of A. This decomposition of the matrix

A provides important topological information about $\mathcal{V}^*(\mathbf{x}^A - \mathbf{b}) \subset (\mathbb{C}^*)^n$ summarized in the following proposition:

Proposition 13.1 (Topological description [26]). *If $\mathcal{V}^*(\mathbf{x}^A - \mathbf{b})$ in $(\mathbb{C}^*)^n$ is not empty, then it consists of a finite number of connected components. Furthermore,*

1) *the number of components is exactly $\left|\prod_{j=1}^{r} d_j\right|$.*
2) *each solution component has codimension equal to* $\operatorname{rank} A = r$.

This description can be strengthened significantly. For P and Q in the Smith Normal Form decomposition of A in (13.4), let $P_r \in M_{r\times n}(\mathbb{Z})$ and $P_0 \in M_{(n-r)\times n}(\mathbb{Z})$ be the top r rows and remaining $n-r$ rows of P respectively; and, in the mean time, let $Q_r \in M_{m\times r}(\mathbb{Z})$ and $Q_0 \in M_{m\times(m-r)}(\mathbb{Z})$ be the left r columns and remaining $m-r$ columns of Q respectively. With these notations, the equation in (13.4) becomes

$$\begin{pmatrix} P_r \\ P_0 \end{pmatrix} A \begin{pmatrix} Q_r & Q_0 \end{pmatrix} = \begin{pmatrix} D & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix} \tag{13.5}$$

with $D = \operatorname{diag}(d_1, \ldots, d_r) \in M_{r\times r}(\mathbb{Z})$ and $\mathbf{0}$'s representing zero block matrices of appropriate sizes. Consequently, the binomial system $\mathbf{x}^A = \mathbf{b}$ can be translated into a form from which more detailed information can be extracted.

Since P and Q are both unimodular the maps $\mathbf{z} \mapsto \mathbf{z}^P$ and $\mathbf{y} \mapsto \mathbf{y}^Q$ are both bijections on $(\mathbb{C}^*)^n$ and $(\mathbb{C}^*)^m$ respectively. Therefore, as regard to the solution set in $(\mathbb{C}^*)^n$, the original system $\mathbf{x}^A = \mathbf{b}$ is equivalent to $(\mathbf{x}^A)^Q = \mathbf{x}^{AQ} = \mathbf{b}^Q$. Similarly, solution sets remain equivalent after change of variables $\mathbf{x} = \mathbf{z}^P$, and

$$(\mathbf{z}^P)^{AQ} = \mathbf{z}^{PAQ} = \mathbf{z}^{\left(\begin{smallmatrix} D & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{smallmatrix}\right)} = (\mathbf{z}^{\left(\begin{smallmatrix} D \\ \mathbf{0} \end{smallmatrix}\right)}, \mathbf{z}^{\left(\begin{smallmatrix} \mathbf{0} \\ \mathbf{0} \end{smallmatrix}\right)}) = \mathbf{b}^Q = (\mathbf{b}^{Q_r}, \mathbf{b}^{Q_0}) .$$

Since $D = \operatorname{diag}(d_1, \ldots, d_r) \in M_{r\times r}(\mathbb{Z})$, the original system $\mathbf{x}^A = \mathbf{b}$ can now be decomposed into a combined system

$$(z_1, \ldots, z_r) \begin{pmatrix} d_1 & & \\ & \ddots & \\ & & d_r \end{pmatrix} = \mathbf{b}^{Q_r} \tag{13.6}$$

$$\mathbf{1} = \mathbf{b}^{Q_0} \tag{13.7}$$

$$z_{r+1}, \ldots, z_n : \text{ free} \tag{13.8}$$

in which (13.7) appears when $r < m$ where $\mathbf{1} = (1, \ldots, 1) \in (\mathbb{C}^*)^{m-r}$, and (13.8) appears when $r < n$. The word "free" in (13.8) means the system imposes no constraints on the $n - r$ variables $z_{r+1}, \ldots, z_n$.

From the above decomposed system, we can see that if $r < m$, then the system is inconsistent unless $\mathbf{1} = \mathbf{b}^{Q_0}$. If the system is consistent (namely, (13.7) holds), then the solutions to (13.6) are exactly

$$\begin{cases} z_1 = e^{2k_1\pi/d_1}\zeta_1 & \text{for} \quad k_1 = 0, \ldots, d_1 - 1 \\ z_2 = e^{2k_2\pi/d_2}\zeta_2 & \text{for} \quad k_2 = 0, \ldots, d_2 - 1 \\ \quad \vdots & \quad \vdots \\ z_r = e^{2k_r\pi/d_r}\zeta_r & \text{for} \quad k_r = 0, \ldots, d_r - 1 \end{cases} \tag{13.9}$$

where each ζ_j is a fixed choice of the d_j-th root of j-th coordinate of $\mathbf{b}^Q$. Clearly, all of them are isolated and the total number of these solutions is $|\prod_{j=1}^r d_j| = |\det D|$. If $r < n$, then the solution set of the decomposed system (13.6)–(13.8) in $(\mathbb{C}^*)^n$ breaks into "components" of the form $\{(e^{2k_1\pi/d_1}\zeta_1, \ldots, e^{2k_r\pi/d_r}\zeta_r, z_{r+1}, \ldots, z_n) : (z_{r+1}, \ldots, z_n) \in (\mathbb{C}^*)^{n-r}\}$, and they are in one-to-one correspondence with solutions in (13.9). Since each component is parametrized by the $n - r$ free variables $z_{r+1}, \ldots, z_n$, it is smooth and of dimension $n - r$. Furthermore, they are disjoint, because these components have distinct $z_1, \ldots, z_r$ coordinates.

To translate the above description of the $(\mathbb{C}^*)^n$-solution set of the decomposed system (in $\mathbf{z}$) into a description of the original solution set $\mathcal{V}^*(\mathbf{x}^A - \mathbf{b})$, one may simply apply the change of variables $\mathbf{x} = \mathbf{z}^P$. Note that this map and its inverse $\mathbf{z} = \mathbf{x}^{P^{-1}}$ are both given by monomials (*bi-regular* maps [39]), the basic properties of the solution set, such as, the number of solution components, their dimensions, and smoothness are therefore preserved. To summarize, the above elaborations assert the following proposition.

Proposition 13.2 (Global parametrization [26, 45, 105]). *For the solution set $\mathcal{V}^*(\mathbf{x}^A - \mathbf{b})$ in $(\mathbb{C}^*)^n$, let P, Q, Q_0 and D be those matrices appeared in the decompositions of A in* (13.4) *and* (13.5)*, and let $r = \operatorname{rank} A$.*

If $\mathbf{1} \neq \mathbf{b}^{Q_0}$ then the binomial system is inconsistent, and hence its solution set in $(\mathbb{C}^*)^n$ is empty.

If $\mathbf{1} = \mathbf{b}^{Q_0}$ then the solution set of $\mathbf{x}^A = \mathbf{b}$ in $(\mathbb{C}^*)^n$ consists of $|\prod_{j=1}^r d_j| = |\det D|$ connected components $V_{k_1,\ldots,k_r}$ for $k_1 \in \{0, \ldots, d_1 - 1\}, \ldots, k_r \in \{0, \ldots, d_r - 1\}$. Each component $V_{k_1,\ldots,k_r}$ is smooth of dimension $n - r$, and it is parametrized by the smooth global parametrization $\phi_{k_1,\ldots,k_r} : (\mathbb{C}^*)^{(n-r)} \to$

$V_{k_1,\ldots,k_r}$ given by

$$\phi_{k_1,\ldots,k_r}(t_1,\ldots,t_{n-r}) = (e^{2k_1\pi/d_1}\zeta_1,\ldots,e^{2k_r\pi/d_r}\zeta_r,t_1,\ldots,t_{n-r})^P \tag{13.10}$$

where each ζ_j is a fixed choice of the d_j-th root of the j-th coordinate of $\mathbf{b}^Q$.

Note that, as mentioned earlier, when $r = n$, the solution set $\mathcal{V}^*(\mathbf{x}^A - \mathbf{b})$ is of dimension $n - r = 0$. Thus, $\mathcal{V}^*(\mathbf{x}^A - \mathbf{b})$ consists of isolated points. In such situations, the "parametrizations" $\phi_{k_1,\ldots,k_r}$ are understood as constants each describes a single isolated point.

Remark 13.3. While the Smith Normal Form $\left[\begin{smallmatrix} D & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{smallmatrix}\right]$ is unique up to a change of signs of the diagonal entries $d_1,\ldots,d_r$ in D (as long as the requirement $d_1 \mid d_2 \mid \cdots \mid d_r$ is satisfied), the transformation matrices P and Q are generally not unique. For a simple example, let $A = \left[\begin{smallmatrix} 2 \\ 3 \end{smallmatrix}\right]$ with Smith Normal Form $\left[\begin{smallmatrix} 1 \\ 0 \end{smallmatrix}\right]$ which is unique up to a change of sign. However, different transformation matrices can be used. Indeed for any $k \in \mathbb{Z}$,

$$\begin{bmatrix} -1+3k & 1-2k \\ -3 & 2 \end{bmatrix} \begin{bmatrix} 2 \\ 3 \end{bmatrix} \begin{bmatrix} 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}.$$

Meaning, there are infinitely many different transformation matrices.

Suppose there is a different pair of unimodular matrices $\tilde{P} = \left[\begin{smallmatrix} \tilde{P}_r \\ \tilde{P}_0 \end{smallmatrix}\right] \in M_{n\times n}(\mathbb{Z})$ where $\tilde{P}_r \in M_{r\times n}(\mathbb{Z})$ and $\tilde{P}_0 \in M_{(n-r)\times n}(\mathbb{Z})$ along with $\tilde{Q} = (\tilde{Q}_r\ \tilde{Q}_0) \in M_{m\times m}(\mathbb{Z})$ where $\tilde{Q}_r \in M_{m\times r}(\mathbb{Z})$ and $\tilde{Q}_0 \in M_{m\times(m-r)}(\mathbb{Z})$ such that

$$\begin{bmatrix} \tilde{P}_r \\ \tilde{P}_0 \end{bmatrix} A \begin{pmatrix} \tilde{Q}_r & \tilde{Q}_0 \end{pmatrix} = \begin{bmatrix} D & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}.$$

Then since $A = P^{-1}\left[\begin{smallmatrix} D & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{smallmatrix}\right] Q^{-1}$,

$$\tilde{P}_0 A = \tilde{P}_0 P^{-1} \begin{pmatrix} D & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix} Q^{-1} = \begin{bmatrix} \mathbf{0} & \mathbf{0} \end{bmatrix}.$$

It follows that $\tilde{P}_0 P^{-1} = \begin{bmatrix} \mathbf{0} & G \end{bmatrix}$ for some matrix $G \in M_{(n-r)\times(n-r)}(\mathbb{Z})$, and hence

$$\tilde{P}_0 = \begin{bmatrix} \mathbf{0} & G \end{bmatrix} P = \begin{bmatrix} \mathbf{0} & G \end{bmatrix} \begin{bmatrix} P_r \\ P_0 \end{bmatrix} = GP_0.$$

Moreover, since $\tilde{P}_0$ consists of the $n - r$ rows of the unimodular matrix $\tilde{P}$, the matrix G must also be unimodular. In other words, columns of $\tilde{P}_0$ must

be the images of columns of P_0 under the linear transformation given by the unimodular matrix G.

Now the Smith Normal Form of A via $\tilde{P}A\tilde{Q}$ yields, according to Proposition 13.2, the parametrization

$$\begin{aligned}\tilde{\phi}_{k_1,\ldots,k_r}(t_1,\ldots,t_{n-r}) &= (e^{2k_1\pi/d_1}\zeta_1,\ldots,e^{2k_r\pi/d_r}\zeta_r,t_1,\ldots,t_{n-r})^{\tilde{P}}\\ &= (e^{2k_1\pi/d_1}\zeta_1,\ldots,e^{2k_r\pi/d_r}\zeta_r)^{\tilde{P}_r}\circ(t_1,\ldots,t_{n-r})^{\tilde{P}_0}\\ &= (e^{2k_1\pi/d_1}\zeta_1,\ldots,e^{2k_r\pi/d_r}\zeta_r)^{\tilde{P}_r}\circ((t_1,\ldots,t_{n-r})^{G})^{P_0}\end{aligned}$$

where "$\circ$" denotes the component-wise multiplication. Therefore while different transformation matrices induce different parametrization, each parametrization is related to all others via nonsingular bijective transformations of the form $\mathbf{t}\mapsto\mathbf{t}^G$ for some unimodular matrix G.

As indicated in Proposition 13.2, for a consistent Laurent binomial system $\mathbf{x}^A=\mathbf{b}$ where $A\in M_{n\times m}(\mathbb{Z})$ with $r=\operatorname{rank}(A)<n$, each component of the solution set in $(\mathbb{C}^*)^n$ will be of dimension $n-r>0$. In this situation, for both theoretical interest and demand from concrete applications, one often wishes to identify another important property: the *degree* of the components. As discussed in §12, the degree of a component can be understood geometrically as the number of its intersections point with a generic affine space of complimentary dimension. That is, for a component $V=V_{k_1,\ldots,k_r}$ of $\mathcal{V}^*(\mathbf{x}^A-\mathbf{b})$ for some fixed choice of $k_1,\ldots,k_r$ as given in Proposition 13.2, the number of isolated intersection point between V and a "generic" affine space of complementary dimension is a fixed number, and this number is the **degree** of V, denoted by $\deg V$.

From the computational standpoint, such a generic linear slicing can be represented by the solution set of a system of $d:=n-r$ linear equations with generic coefficients. Therefore $\deg V$ is precisely the number of points $\mathbf{x}=(x_1,\ldots,x_n)\in V$ that satisfies the system of linear equations

$$\begin{cases} c_{11}x_1+c_{12}x_2+\cdots+c_{1n}x_n=c_{10}\\ c_{21}x_1+c_{22}x_2+\cdots+c_{2n}x_n=c_{20}\\ \qquad\qquad\qquad\qquad\qquad\vdots\\ c_{d1}x_1+c_{d2}x_2+\cdots+c_{dn}x_n=c_{d0}. \end{cases}\tag{13.11}$$

where c_{ij} for $i=1,\ldots,d$ and $j=0,\ldots,n$ are generic complex numbers. But the set $V=V_{k_1,\ldots,k_r}$ is precisely the image of the injective map

$$\phi_{k_1,\ldots,k_r}(t_1,\ldots,t_d)\;=\;(e^{2k_1\pi/d_1}\zeta_1,\ldots,e^{2k_r\pi/d_r}\zeta_r,t_1,\ldots,t_d)^P$$

in Proposition 13.2. If we let $\xi = (e^{2k_1\pi/d_1}\zeta_1, \ldots, e^{2k_r\pi/d_r}\zeta_r)$ and $\mathbf{t} = (t_1, \ldots, t_d)$ then

$$\phi_{k_1,\ldots,k_r}(\mathbf{t}) = (\xi, \mathbf{t})^{\binom{P_r}{P_0}} = (\, \xi^{\mathbf{p}_r^{(1)}}\mathbf{t}^{\mathbf{p}_0^{(1)}}, \ \ldots, \ \xi^{\mathbf{p}_r^{(n)}}\mathbf{t}^{\mathbf{p}_0^{(n)}} \,)$$

where for each $j = 1, \ldots, n$, $\mathbf{p}_r^{(j)}$ and $\mathbf{p}_0^{(j)}$ are the j-th columns of P_r and P_0 respectively. In other words, V has the global parametrization $x_i = \xi^{\mathbf{p}_r^{(i)}}\mathbf{t}^{\mathbf{p}_0^{(i)}}$ for $i = 1, \ldots, n$. Therefore the intersections between V and the generic affine space defined by (13.11) are precisely the solutions of the polynomial system

$$\begin{cases} c_{11}\,\xi^{\mathbf{p}_r^{(1)}}\,\mathbf{t}^{\mathbf{p}_0^{(1)}} + c_{12}\,\xi^{\mathbf{p}_r^{(2)}}\,\mathbf{t}^{\mathbf{p}_0^{(2)}} + \cdots + c_{1n}\,\xi^{\mathbf{p}_r^{(n)}}\,\mathbf{t}^{\mathbf{p}_0^{(n)}} = c_{10} \\ c_{21}\,\xi^{\mathbf{p}_r^{(1)}}\,\mathbf{t}^{\mathbf{p}_0^{(1)}} + c_{22}\,\xi^{\mathbf{p}_r^{(2)}}\,\mathbf{t}^{\mathbf{p}_0^{(2)}} + \cdots + c_{2n}\,\xi^{\mathbf{p}_r^{(n)}}\,\mathbf{t}^{\mathbf{p}_0^{(n)}} = c_{20} \\ \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \vdots \\ c_{d1}\,\xi^{\mathbf{p}_r^{(1)}}\,\mathbf{t}^{\mathbf{p}_0^{(1)}} + c_{d2}\,\xi^{\mathbf{p}_r^{(2)}}\,\mathbf{t}^{\mathbf{p}_0^{(2)}} + \cdots + c_{dn}\,\xi^{\mathbf{p}_r^{(n)}}\,\mathbf{t}^{\mathbf{p}_0^{(n)}} = c_{d0}. \end{cases}$$

By letting $c'_{ij} := c_{ij}\xi^{\mathbf{p}_r^{(j)}} \in \mathbb{C}$ and $c'_{i0} = c_{i0}$ for each $i = 1, \ldots, d$ and $j = 1, \ldots, n$, the system above is a system of d polynomial equations in the variables $\mathbf{t} = (t_1, \ldots, t_d)$ with generic complex coefficients c'_{ij} and same set of monomials $\mathbf{t}^{\mathbf{p}_0^{(j)}}$:

$$\begin{cases} c'_{11}\mathbf{t}^{\mathbf{p}_0^{(1)}} + c'_{12}\mathbf{t}^{\mathbf{p}_0^{(2)}} + \cdots + c'_{1n}\mathbf{t}^{\mathbf{p}_0^{(n)}} = c_{10} \\ \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \vdots \\ c'_{d1}\mathbf{t}^{\mathbf{p}_0^{(1)}} + c'_{d2}\mathbf{t}^{\mathbf{p}_0^{(2)}} + \cdots + c'_{dn}\mathbf{t}^{\mathbf{p}_0^{(n)}} = c_{d0}. \end{cases} \tag{13.12}$$

This derivation equates the degree of V and the number of nonzero solutions of the above system, and this fact is summarized in the following proposition.

Proposition 13.4 (Degree via affine space cut). *If $r < n$ and $\mathcal{V}^*(\mathbf{x}^A - \mathbf{b}) \neq \varnothing$, then the degree of each component V of $\mathcal{V}^*(\mathbf{x}^A - \mathbf{b})$ agrees with the number of solutions $\mathbf{t} \in (\mathbb{C}^*)^d$ of the system of d Laurent polynomial equations*

$$\begin{cases} c_{11}\mathbf{t}^{\mathbf{p}_0^{(1)}} + c_{12}\mathbf{t}^{\mathbf{p}_0^{(2)}} + \cdots + c_{1n}\mathbf{t}^{\mathbf{p}_0^{(n)}} = c_{10} \\ \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \vdots \\ c_{d1}\mathbf{t}^{\mathbf{p}_0^{(1)}} + c_{d2}\mathbf{t}^{\mathbf{p}_0^{(2)}} + \cdots + c_{dn}\mathbf{t}^{\mathbf{p}_0^{(n)}} = c_{d0} \end{cases} \tag{13.13}$$

for generic complex coefficients $c_{ij} \in \mathbb{C}$.

The "unmixed" case of Bernshtein's Theorem (Theorem 6.4) applies here: for generic coefficients, the number of isolated $\mathbb{C}^*$-solutions is the normalized volume of the Newton polytope:

Proposition 13.5 (Degree as volume).

$$\deg V = d! \cdot \mathrm{Vol}_d(\mathrm{conv}\{\mathbf{p}_0^{(1)}, \ldots, \mathbf{p}_0^{(n)}, \mathbf{0}\}) \tag{13.14}$$

where $\mathbf{0} = (0, \ldots, 0)^\top \in \mathbb{R}^d$ *and columns* $\mathbf{p}_0^{(1)}, \ldots, \mathbf{p}_0^{(n)}$ *of the matrix* P_0 *are considered as points in* $\mathbb{R}^d$.

Remark 13.6. Even though the transformation matrices with which the Smith Normal Form of A (13.4) is constituted are not unique, as asserted in Remark 13.3, for a different pair of unimodular transformation matrices $\tilde{P} = \begin{bmatrix} \tilde{P}_r \\ \tilde{P}_0 \end{bmatrix} \in M_{n\times n}(\mathbb{Z})$ where $\tilde{P}_0 \in M_{(n-r)\times n}(\mathbb{Z})$ and $\tilde{Q} \in M_{m\times m}(\mathbb{Z})$ for which

$$\tilde{P}A\tilde{Q} = PAQ = \begin{bmatrix} D & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}$$

we must have $\tilde{P}_0 = GP_0$ for some unimodular matrix $G \in M_{(n-r)\times(n-r)}(\mathbb{Z})$. Therefore the columns of $\tilde{P}_0$ are vectors $G\mathbf{p}_0^{(1)}, \ldots, G\mathbf{p}_0^{(n)}$. Since G is unimodular, as a linear transformation it preserves the volume, and therefore

$$\mathrm{Vol}_d(\mathrm{conv}\{G\mathbf{p}_0^{(1)}, \ldots, G\mathbf{p}_0^{(n)}, \mathbf{0}\}) = \mathrm{Vol}_d(\mathrm{conv}\{\mathbf{p}_0^{(1)}, \ldots, \mathbf{p}_0^{(n)}, \mathbf{0}\}).$$

In other words, while the formulation of (13.14) depends on the choice of the transformation matrices in the Smith Normal Form decomposition of A, the actual value is nonetheless invariant.

13.2. Smith Normal Form computation

As summarized in Proposition 13.2, the key to finding the dimension, number of components, and global parametrization of the $\mathbb{C}^*$-solution set $\mathcal{V}^*(\mathbf{x}^A - \mathbf{b}) \subset (\mathbb{C}^*)^n$ is the Smith Normal Form (13.4) of the exponent matrix A. In this section, we will list a procedure for efficiently computing the Smith Normal Form of an integer matrix.

To illustrate the main idea behind this procedure, we start with the simplest case of an integer matrix

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$$

where $a_{11} \neq 0$. As in the computation of Hermite Normal Form discussed in §7, there exist, by the Bézout's identity, integers u and v such that $d'_1 := \gcd(a_{11}, a_{21}) = u\, a_{11} + v\, a_{21}$. Let

$$P_1 = \begin{bmatrix} u & v \\ -\dfrac{a_{21}}{d'_1} & \dfrac{a_{11}}{d'_1} \end{bmatrix}, \quad \text{then} \quad \det P_1 = \frac{ua_{11} + va_{21}}{d'_1} = \frac{d'_1}{d'_1} = 1,$$

and thus P_1 is unimodular. This can provide a row reduction:

$$P_1 A = \begin{bmatrix} u & v \\ -\dfrac{a_{21}}{d'_1} & \dfrac{a_{11}}{d'_1} \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} = \begin{bmatrix} d'_1 & ua_{12} + va_{22} \\ 0 & -\dfrac{a_{21}a_{12}}{d'_1} + \dfrac{a_{11}a_{22}}{d'_1} \end{bmatrix} = \begin{bmatrix} d'_1 & a'_{12} \\ 0 & * \end{bmatrix}.$$

However, our goal is to diagonalize A, hence further column reduction is needed. Similar to row reduction, there exist integers x and y such that $d_1 := \gcd(d'_1, a'_{12}) = x\, d'_1 + y\, a'_{12}$. Let

$$Q_1 = \begin{bmatrix} x & -\dfrac{a'_{12}}{d_1} \\ y & \dfrac{d'_1}{d_1} \end{bmatrix}, \quad \text{then} \quad \det Q = \frac{xd'_1 + ya'_{12}}{d_1} = \frac{d_1}{d_1} = 1,$$

and Q_1 is also unimodular. As desired, it gives column reduction

$$A_1 := P_1 A\, Q_1 = \begin{bmatrix} d_1 & 0 \\ * & * \end{bmatrix}.$$

While the progress made by the row reduction $P_1 A$ seems to be demolished by this column reduction, it is important to note that the new upper left corner entry d_1 divides the original entry a_{11} (indeed, $d_1 \mid d'_1 \mid a_{11}$).

Similar row and column reductions can be applied to produce $A_2 := P_2 A_1 Q_2$, $A_3 = P_3 A_2 Q_3$, ... But if d_i is the upper left corner entry of A_i, then we must have $0 \neq d_i \mid d_{i-1} \mid \cdots \mid d_2 \mid d_1 \mid a_{11}$. Consequently, there must be an iteration, say k-th iteration, for which $d_{k+1} = d_k$. It follows that d_k

must divide entries on both its row and column. That is, A_k has the form

$$\begin{bmatrix} d_k & s\,d_k \\ t\,d_k & * \end{bmatrix}$$

for some $s, t \in \mathbb{Z}$. At this point further multiplications by unimodular matrices

$$\begin{bmatrix} 1 & 0 \\ -t & 1 \end{bmatrix} \begin{bmatrix} d_k & s\,d_k \\ t\,d_k & * \end{bmatrix} \begin{bmatrix} 1 & -s \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} d_k & 0 \\ 0 & * \end{bmatrix}$$

provides the desired diagonal form.

In general, $n \times n$ and $m \times m$ version of the above matrices P and Q can be constructed to perform row and column reduction respectively for an $n \times m$ integer matrix.

After repeated such row and column reduction together with potential row and column permutations one can construct unimodular matrices $P^{(1)}, \ldots, P^{(k)} \in M_{n\times n}(\mathbb{Z})$ and $Q^{(1)}, \ldots, Q^{(\ell)} \in M_{m\times m}(\mathbb{Z})$ such that

$$P^{(k)} \cdots P^{(1)}\, A\, Q^{(1)} \cdots Q^{(\ell)} = \begin{pmatrix} d_1 & & & & \\ & \ddots & & & \\ & & d_r & & \\ & & & 0 & \\ & & & & \ddots \\ & & & & & 0 \end{pmatrix}$$

with $r = \operatorname{rank} A$ and $d_1, \ldots, d_r$ are nonzeros. As noted in standard references such as [34], employing further reductions can ensure $d_1 \mid d_2 \mid \cdots \mid d_r$, but for the purpose of solving binomial systems, this property does not seem necessary.

13.3. Degree computation

When the solution set consists of positive dimensional components, Proposition 13.5 provides a computationally viable means for computing the degree of each component as the volume of a convex polytope. Let V be a component of $\mathcal{V}^*(\mathbf{x}^A - \mathbf{b}) \subset (\mathbb{C}^*)^n$, $d = \dim V = n - r = n - \operatorname{rank} A$, and $P^0 = (\mathbf{p}_0^{(1)}, \ldots, \mathbf{p}_0^{(n)}) \in M_{d\times n}(\mathbb{Z})$ be the matrix appears in (13.4). Considering each $\mathbf{p}_0^{(j)}$ as a point in $\mathbb{R}^d$, let $S = \{\mathbf{p}_0^{(1)}, \ldots, \mathbf{p}_0^{(n)}, \mathbf{0}\} \subset \mathbb{R}^d$ be the finite point set. Then by Proposition 13.5,

$$\deg V = d!\,\mathrm{Vol}_d(\operatorname{conv} S). \tag{13.15}$$

Even though any algorithm for computing the volume of a convex polytope can be used to compute the degree via (13.15), there are two main reasons behind our specialized algorithms for computing the degree $\deg V$:

First, from a numerical point of view, the fact that $\deg V = d!\mathrm{Vol}_d(\mathrm{conv}S)$ must be an integer permits the use of efficient but potentially less accurate numerical methods as well as floating point arithmetic, they can still obtain the correct result. Indeed, the exact result can be achieved as long as the total absolute error remains below 1/2. This is certainly not possible for methods that are designed to compute volumes of more general convex polytopes.

Secondly, in Numerical Algebraic Geometry, the computation of witness points (§12) is a fundamental problem. The framework we establish to calculate $\deg V$ helps, as to be described in §13.4, to construct a specialized homotopy method which is ideal for efficiently providing witness points in this environment. The construction of the homotopy, however, requires the simplicial subdivision of the polytope $\mathrm{conv}\, S$. Algorithms that do not produce simplicial subdivisions are therefore not suitable for this task.

The concept of mixed volume is actually a generalization of the volume. Indeed,

$$\deg V = d\,!\,\mathrm{Vol}_d(\mathrm{conv}\, S) = \mathcal{M}(\mathrm{conv}\, S, \ldots, \mathrm{conv}\, S).$$

The computation of $\deg V$ therefore becomes a special case of the mixed volume computation discussed in §6.2, and the algorithm described in §8 can be used directly: A "lifting function" $\omega : S \to \mathbb{Q}$ with generic images is used to "lift" points in S to one higher dimension via $\mathbf{p} \mapsto (\mathbf{p}, \omega(\mathbf{p})) \in \mathbb{Q}^{d+1}$. Let $\hat{S}$ be the collection of the lifted points. Proposition 6.16 asserts that the projection of the facets on the lower hull of $\mathrm{conv}\, \hat{S}$ form a "fine unmixed subdivision" of S, and the algorithm of systematic face extensions can be used to enumerate the lower facets efficiently. Here the "unmixed subdivision" of the single polytope $\mathrm{conv}\, S$ is simply a *subdivision* in the familiar sense: a collection of simplices intersecting only on their faces yet fills the entire $\mathrm{conv}\, S$. Such a subdivision, obtained as the projection of the lower hull of a generic lifting, is known as a *regular subdivision* [51], and it will be of critical importance in the next section.

Since the cells of such a fine ummixed subdivision are simplices, their volume are easy to compute. The degree $\deg V$ is then the sum of the volume of all these cells multiplied by $d\,!$.

13.4. Computing witness sets

Witness sets, discussed in detail in §12.1 is the core concept of Numerical Algebraic Geometry. In the case of $\mathbb{C}^*$-solution set of binomial systems, the witness set can be computed by a specialized polyhedral homotopy based method.

By Proposition 13.4, the intersection between a component $V \subseteq \mathcal{V}(\mathbf{x}^A - \mathbf{b})$ and a generic affine space of complementary dimension consists of precisely the points $\mathbf{t} = (t_1, \ldots, t_d) \in (\mathbb{C}^*)^d$ that satisfy the system of d Laurent polynomial equations in d variables given by

$$\begin{aligned} c_{11}\mathbf{t}^{\mathbf{p}_0^{(1)}} + c_{12}\mathbf{t}^{\mathbf{p}_0^{(2)}} + \cdots + c_{1n}\mathbf{t}^{\mathbf{p}_0^{(n)}} &= c_{10} \\ &\vdots \\ c_{d1}\mathbf{t}^{\mathbf{p}_0^{(1)}} + c_{d2}\mathbf{t}^{\mathbf{p}_0^{(2)}} + \cdots + c_{dn}\mathbf{t}^{\mathbf{p}_0^{(n)}} &= c_{d0} \end{aligned} \tag{13.16}$$

where the coefficients depends on both the choice of the component in $\mathcal{V}^*(\mathbf{x}^A - \mathbf{b})$ and the choice of the r-dimensional affine space. Apparently, this system is *fully unmixed* (every equation has exactly the same monomials). We shall find all its isolated zeros in $(\mathbb{C}^*)^n$ by the polyhedral homotopies.

For simplicity, we reuse the notations from §13.3. Namely, we let $\mathbf{a}_0 = \mathbf{0}$, $\mathbf{a}_1 = \mathbf{t}^{\mathbf{p}_0^{(1)}}, \ldots, \mathbf{a}_n = \mathbf{t}^{\mathbf{p}_0^{(n)}}$. With $S = \{\mathbf{a}_0, \ldots, \mathbf{a}_n\}$, let $\omega : S \to \mathbb{Q}$ be the generic lifting function used for constructing regular simplicial subdivision of $\operatorname{conv} S$ in §13.3. As before, $\hat{\mathbf{a}}_j = (\mathbf{a}_j, \omega(\mathbf{a}_j))$ for $j = 0, \ldots, n$, and $\hat{S} = \{\hat{\mathbf{a}}_0, \ldots, \hat{\mathbf{a}}_n\} \subset \mathbb{Q}^{d+1}$. With a new variable $s \in [0, 1]$, consider the homotopy

$$H(\mathbf{t}, s) = \begin{cases} c_{11}\mathbf{t}^{\mathbf{a}_1}s^{\omega(\mathbf{a}_1)} + \cdots + c_{1n}\mathbf{t}^{\mathbf{a}_n}s^{\omega(\mathbf{a}_n)} - c_{10}s^{\omega(\mathbf{a}_0)} = 0 \\ \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \vdots \\ c_{d1}\mathbf{t}^{\mathbf{a}_1}s^{\omega(\mathbf{a}_1)} + \cdots + c_{dn}\mathbf{t}^{\mathbf{a}_n}s^{\omega(\mathbf{p}_0^{(n)})} - c_{d0}s^{\omega(\mathbf{a}_0)} = 0 \end{cases} \tag{13.17}$$

which is constructed by multiplying each term in (13.16) by a rational power of the new variable s whose exponent is determined by the lifting function $\omega : S \to \mathbb{Q}$. Clearly, $H(\mathbf{t}, 1) = \mathbf{0}$ is exactly the system (13.16) which we intend to solve (inside $(\mathbb{C}^*)^d$). Similar to solving fully mixed systems by the polyhedral homotopies, $H(\mathbf{t}, 0)$ cannot be used as the starting system since at $s = 0$, the system is either identically zero or undefined. Therefore certain transformation is necessary to produce a meaningful and solvable starting system.

Let $\mathcal{D}$ be a fine unmixed subdivision induced by ω. Each cell in $\mathcal{D}$ is a projection of a cell of the form, say $\{\hat{\mathbf{a}}_0, \ldots, \hat{\mathbf{a}}_d\}$, such that $\operatorname{conv}\{\hat{\mathbf{a}}_0, \ldots, \hat{\mathbf{a}}_d\}$ is a lower d-face of $\operatorname{conv} \hat{S}$. More precisely, there exists a (unique) vector of the form $\hat{\boldsymbol{\alpha}} = (\alpha_1, \ldots, \alpha_d, 1)$ for which

$$\begin{aligned} &\langle \hat{\mathbf{a}}_0, \hat{\boldsymbol{\alpha}} \rangle = \langle \hat{\mathbf{a}}_j, \hat{\boldsymbol{\alpha}} \rangle \quad \text{for } j = 1, \ldots, d \quad \text{and} \\ &\langle \hat{\mathbf{a}}_0, \hat{\boldsymbol{\alpha}} \rangle < \langle \hat{\mathbf{a}}, \hat{\boldsymbol{\alpha}} \rangle \quad \text{for all } \ \mathbf{a} \in S \backslash \{\mathbf{a}_1, \ldots, \mathbf{a}_d\}. \end{aligned} \tag{13.18}$$

For $\hat{\boldsymbol{\alpha}} = (\alpha_1, \ldots, \alpha_d, 1)$, by the change of variables with $\mathbf{y} = (y_1, \ldots, y_d)$

$$\mathbf{t} = \begin{cases} t_1 = y_1 s^{\alpha_1} \\ \quad \vdots \\ t_d = y_d s^{\alpha_d}, \end{cases} \tag{13.19}$$

then $H(\mathbf{t}, s)$ becomes

$$H(\mathbf{t}, s) = H(y_1 s^{\alpha_1}, \ldots, y_d s^{\alpha_d}, s) = \begin{cases} \displaystyle\sum_{\mathbf{a} \in S} c_{1,\mathbf{a}} \mathbf{y}^{\mathbf{a}} s^{\langle \mathbf{a}, \boldsymbol{\alpha} \rangle + \omega(\mathbf{a})} = \sum_{\mathbf{a} \in S} c_{1,\mathbf{a}} \mathbf{y}^{\mathbf{a}} s^{\langle \hat{\mathbf{a}}, \hat{\boldsymbol{\alpha}} \rangle} \\ \qquad\qquad\qquad\qquad \vdots \\ \displaystyle\sum_{\mathbf{a} \in S} c_{d,\mathbf{a}} \mathbf{y}^{\mathbf{a}} s^{\langle \mathbf{a}, \boldsymbol{\alpha} \rangle + \omega(\mathbf{a})} = \sum_{\mathbf{a} \in S} c_{d,\mathbf{a}} \mathbf{y}^{\mathbf{a}} s^{\langle \hat{\mathbf{a}}, \hat{\boldsymbol{\alpha}} \rangle}. \end{cases}$$

Let $\beta = \langle \hat{\mathbf{a}}_0, \hat{\boldsymbol{\alpha}} \rangle$ and define a new homotopy

$$H^{\boldsymbol{\alpha}, \beta}(\mathbf{y}, s) = s^{-\beta} H(y_1 s^{\alpha_1}, \ldots, y_d s^{\alpha_d}, s) = \begin{cases} \displaystyle s^{-\beta} \sum_{\mathbf{a} \in S} c_{1,\mathbf{a}} \mathbf{y}^{\mathbf{a}} s^{\langle \hat{\mathbf{a}}, \hat{\boldsymbol{\alpha}} \rangle} \\ \qquad\qquad \vdots \\ \displaystyle s^{-\beta} \sum_{\mathbf{a} \in S} c_{d,\mathbf{a}} \mathbf{y}^{\mathbf{a}} s^{\langle \hat{\mathbf{a}}, \hat{\boldsymbol{\alpha}} \rangle}. \end{cases}$$

Note that the new homotopy still has the necessary property that $H^{\boldsymbol{\alpha}, \beta}(\mathbf{y}, 1) = \mathbf{0}$ is identical to the system of equations in (13.16). Moreover, by (13.18), there are precisely $d+1$ terms in each equation of $H^{\boldsymbol{\alpha}, \beta}(\mathbf{y}, s)$ having no power of s (the terms corresponding to $\mathbf{a}_0, \ldots, \mathbf{a}_d$), and all other terms have positive powers of s. Consequently, at $s = 0$, terms with positive powers of s vanish, leaving only

$$\begin{cases} c_{1,\mathbf{a}_0} \mathbf{y}^{\mathbf{a}_0} + c_{1,\mathbf{a}_1} \mathbf{y}^{\mathbf{a}_1} + \cdots + c_{1,\mathbf{a}_d} \mathbf{y}^{\mathbf{a}_d} = 0 \\ c_{2,\mathbf{a}_0} \mathbf{y}^{\mathbf{a}_0} + c_{2,\mathbf{a}_1} \mathbf{y}^{\mathbf{a}_1} + \cdots + c_{2,\mathbf{a}_d} \mathbf{y}^{\mathbf{a}_d} = 0 \\ \qquad\qquad\qquad\qquad\qquad\qquad \vdots \\ c_{d,\mathbf{a}_0} \mathbf{y}^{\mathbf{a}_0} + c_{d,\mathbf{a}_1} \mathbf{y}^{\mathbf{a}_1} + \cdots + c_{d,\mathbf{a}_d} \mathbf{y}^{\mathbf{a}_d} = 0. \end{cases} \tag{13.20}$$

Let

$$C = \begin{pmatrix} c_{1,\mathbf{a}_0} & \cdots & c_{1,\mathbf{a}_d} \\ \vdots & \ddots & \vdots \\ c_{d,\mathbf{a}_0} & \cdots & c_{d,\mathbf{a}_d} \end{pmatrix} \quad \text{and} \quad \Gamma = \begin{pmatrix} \mathbf{a}_0 & \cdots & \mathbf{a}_d \end{pmatrix}^\top,$$

then the above equation can be written as

$$C \cdot (\mathbf{y}^{\Gamma})^{\top} = \mathbf{0}. \tag{13.21}$$

For generic choices of the coefficients, there exists a nonsingular matrix $G \in M_{d\times d}(\mathbb{C})$ such that

$$GC = \begin{pmatrix} c_{11}^* & & & c_{12}^* \\ & c_{21}^* & & c_{22}^* \\ & & \ddots & \vdots \\ & & c_{d1}^* & c_{d2}^* \end{pmatrix}. \tag{13.22}$$

for some $c_{ij}^* \in \mathbb{C}^*$. Thus, without altering its solution set, (13.21) can be converted to the equivalent system

$$GC(\mathbf{y}^{\Gamma})^{\top} = \begin{cases} c_{11}^*\mathbf{y}^{\mathbf{a}_0} & & & + & c_{12}^*\mathbf{y}^{\mathbf{a}_d} & = & 0 \\ & c_{21}^*\mathbf{y}^{\mathbf{a}_1} & & + & c_{22}^*\mathbf{y}^{\mathbf{a}_d} & = & 0 \\ & & & \vdots & \vdots & \vdots & \\ & & c_{d1}^*\mathbf{y}^{\mathbf{a}_{d-1}} & + & c_{d2}^*\mathbf{y}^{\mathbf{a}_d} & = & 0, \end{cases} \tag{13.23}$$

which is apparently a square Laurent binomial system. The algorithm developed earlier for solving square Laurent binomial system can be used here to solve this system. The solutions are precisely the solutions of the starting system (13.20) for the homotopy $H^{\boldsymbol{\alpha},\beta}$. Tracing solution paths of $H^{\boldsymbol{\alpha},\beta} = \mathbf{0}$ emanating from those solutions will locate solutions to the target system (13.16) when $s = 1$. They are points in the witness set of the component V of $\mathcal{V}^*(\mathbf{x}^A - b)$.

The construction of the homotopy $H^{\boldsymbol{\alpha},\beta}$ relies on a cell whose convex hull is a lower d-face of $\operatorname{conv}\hat{S}$ in the fine unmixed subdivision $\mathcal{D}$ of $\operatorname{conv} S$. It is typical for $\mathcal{D}$ to contain more than one such cells. Evidently, each cell can induce a different homotopy in the form of $H^{\boldsymbol{\alpha},\beta}$. Just like solving fully mixed polynomial systems by the polyhedral homotopies before, when one goes through all those cells in $\mathcal{D}$, the resulting homotopies of the form $H^{\boldsymbol{\alpha},\beta}$ will find all isolated solutions of (13.16) which constitute the witness set of V.

13.5. Verifying the consistency numerically

While the binomial system is assumed to be consistent throughout previous sections, in more general cases as stated in Proposition 13.2, when the

number of equations in the given binomial system $\mathbf{x}^A = \mathbf{b}$ is greater than the rank of the matrix A, the system may become inconsistent. With the notations in (13.5), the system is consistent if and only if

$$\mathbf{b}^{Q_0} = \mathbf{1}$$

where $\mathbf{1} = (1, \ldots, 1) \in (\mathbb{C}^*)^{m-r}$. So the consistency of the binomial system can be verified by simply checking the above equality. This can certainly be accomplished quite easily when $\mathbf{b}$ is given in the exact form. When $\mathbf{b}$ is only given approximately, however, verifying $\mathbf{b}^{Q_0} = \mathbf{1}$ becomes an *ill-posed* problem, it should be avoided at all cost. After all, a generic perturbation in $\mathbf{b}$, however small in magnitude, will break the above equality.

The main strategy is to rephrase the question of consistency into a question of closeness: *How close is the binomial system from being consistent?* More precisely, let W be the algebraic set in $(\mathbb{C}^*)^n$ defined by

$$\mathbf{b}^{Q_0} = \mathbf{1}$$

then the binomial system is consistent if and only if $\mathbf{b}$ lies in W. Therefore, the distance between $\mathbf{b}$ and the smooth part of W may be used as a measure of how *close* the binomial system is being consistent. Under this substitution, the resulting problem becomes *well-posed*, and can be answered via numerical computation.

While the distance between a point and an algebraic set may be generally difficult to compute, the distance in the log-norm space can be obtained quite easily: since $\mathbf{b}^{Q_0} = \mathbf{1}$ is equivalent to

$$\begin{cases} (\mathrm{Re}(\log \mathbf{b})) \cdot Q_0 = \mathbf{0} \\ (\mathrm{Im}(\log \mathbf{b})) \cdot Q_0 = \mathbf{0} \pmod{2\pi} \end{cases}$$

where Re and Im denote the component-wise real and imaginary parts respectively. So the distance, in the log-norm sense, can be computed simply as the distance between $\mathrm{Re}(\log \mathbf{b})$ and the kernel of Q_0 and the distance between $\mathrm{Im}(\log \mathbf{b})$ and the kernel of Q_0 modulo 2π, and this distance should provide a coarse indication for the consistency of the binomial system $\mathbf{x}^A - \mathbf{b}$ over $(\mathbb{C}^*)^n$.

14. Numerical considerations

14.1. Scaling of the coefficients

In applications, it is not uncommon to encounter polynomial systems with "unbalanced" coefficients — some coefficients are much larger than the others (see examples listed in [53]) which often results in ill-conditioned Jacobian matrix of the homotopy function. This will, in turn, affect the efficiency of the path tracing algorithm. The idea of scaling the system to balance the magnitudes of the coefficients of the polynomials first appeared in [80]. We will illustrate the scaling method by an example.

Example 14.1. Consider the following system of two equations in two unknowns

$$8000\,x_1^2x_2^2 - 2000\,x_1 + 1 = 0 \tag{14.1}$$

$$5000\,x_1x_2 - 30 = 0. \tag{14.2}$$

To scale the variables, let $x_1 = 10^{c_1}z_1$ and $x_2 = 10^{c_2}z_2$, and to scale the equations, multiply (14.1) by 10^{c_3} and multiply (14.2) by 10^{c_4}. This gives

$$\begin{aligned} 10^{c_3}(8000 * 10^{2c_1+2c_2}\,z_1^2z_2^2 - 2000 * 10^{c_1}z_1 + 1) &= 0 \\ 10^{c_4}(5000 * 10^{c_1+c_2}\,z_1z_2 - 30) &= 0. \end{aligned}$$

Or,

$$\begin{aligned} 10^{E_1}z_1^2z_2^2 - 10^{E_2}z_1 + 10^{E_3} &= 0 \\ 10^{E_4}z_1z_2 - 10^{E_5} &= 0 \end{aligned}$$

where

$$\begin{aligned} E_1 &= 2c_1 + 2c_2 + c_3 + \log_{10}(8000) \\ E_2 &= c_1 + c_3 + \log_{10}(2000) \\ E_3 &= c_3 \\ E_4 &= c_1 + c_2 + c_4 + \log_{10}(5000) \\ E_5 &= c_4 + \log_{10}(30). \end{aligned}$$

To have the numerical stability afforded by coefficients centered about unity, we want each E_i to be close to 0. Furthermore, to reduce variability among

the magnitude of the coefficients in each equation, we want the difference between each pair of $E_i's$ in an equation to be close to 0. Thus, setting

$$r_1 \equiv E_1^2 + E_2^2 + E_3^2 + E_4^2 + E_5^2$$
$$r_2 \equiv [(E_1 - E_2)^2 + (E_2 - E_3)^2 + (E_1 - E_3)^2] + [(E_4 - E_5)^2],$$

we wish to minimize $r = r_1 + r_2$. More explicitly,

$$\begin{aligned} r = {} & (2c_1 + 2c_2 + c_3 + \log(8000))^2 + (c_1 + c_3 + \log(2000))^2 + c_3^2 \\ & + (c_1 + c_2 + c_4 + \log(5000))^2 + (c_4 + \log(30))^2 \\ & + (c_1 + 2c_2 + \log(8000) - \log(2000))^2 \\ & + (2c_1 + 2c_2 + \log(8000))^2 + (c_1 + \log(2000))^2 \\ & + (c_1 + c_2 + \log(5000) - \log(30))^2. \end{aligned} \tag{14.3}$$

In [80], r is considered as a second degree polynomial in four unknowns $c_1,\ c_2,\ c_3,\ c_4$ and is minimized by the solution of

$$\frac{\partial r}{\partial c_i} = 0 \quad \text{for } i = 1, 2, 3, 4.$$

Actually, r in (14.3) can be written as

$$r = \left\| \underbrace{\begin{pmatrix} 2 & 2 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 2 & 0 & 0 \\ 2 & 2 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix}}_{\mathrm{A}} \underbrace{\begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{pmatrix}}_{x} - \underbrace{\begin{pmatrix} -\log(8000) \\ -\log(2000) \\ 0 \\ -\log(5000) \\ -\log(30) \\ \log(2000) - \log(8000) \\ -\log(8000) \\ -\log(2000) \\ \log(30) - \log(5000) \end{pmatrix}}_{b} \right\|_2^2$$

$$= \| Ax - b \|_2^2 ,$$

its minimization is therefore the solution of a linear least squares problem. With the solution of this least squares problem $c_1 = -3.3437,\ c_2 =$

1.3495, $c_3 = 0.0427$, and $c_4 = -1.5909$, the original equations then become

$$\begin{aligned} 0.9064\, z_1^2 z_2^2 - z_1 + 1.1033 &= 0 \\ 1.2996\, z_1 z_2 - 0.7695 &= 0. \end{aligned}$$

Clearly, the new system has coefficients with magnitudes smaller than those of the original one. They are closer to unity and to each other. When solutions $z = (z_1, z_2)$ of the new system are located, the solutions $x = (x_1, x_2)$ can be attained by applying the transformation $x_1 = 10^{c_1} z_1$ and $x_2 = 10^{c_2} z_2$.

In general occasions where equations in the systems have many terms, we will ignore the requirement that reduces variability among the magnitude of the coefficients in each equation. Namely, we only minimize r_1 above, making the corresponding linear least squires problem much easier to solve.

14.2. Endgames

14.2.1. Deflation. At the end of tracking a homotopy path where $t = 1$, usually Newton's iterations is applied for the final approximation of the solution. While Newton's method converges rapidly with high accuracy at nonsingular solutions of a polynomial system, the desired number of significant digits for a singular solution may not be achievable, as the following example shows.

Example 14.2. The system of polynomial equations

$$\begin{cases} x_1^2 + x_1 + x_2 + x_3 + x_4 + x_5 - 2x_1 - 4 = 0 \\ x_2^2 + x_1 + x_2 + x_3 + x_4 + x_5 - 2x_2 - 4 = 0 \\ x_3^2 + x_1 + x_2 + x_3 + x_4 + x_5 - 2x_3 - 4 = 0 \\ x_4^2 + x_1 + x_2 + x_3 + x_4 + x_5 - 2x_4 - 4 = 0 \\ x_5^2 + x_1 + x_2 + x_3 + x_4 + x_5 - 2x_5 - 4 = 0 \end{cases}$$

has a solution $(x_1, x_2, x_3, x_4, x_5) = (1, 1, 1, 1, 1)$ of multiplicity 16. After the final stage of the homotopy method, the best approximation we are able to achieve by using the double-precision IEEE floating point arithmetic have about 4-digit accuracy, such as

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix} = \begin{pmatrix} 0.99997588488142196729 \\ 0.99997475062867600718 \\ 1.0000433370416735036 \\ 1.0000382630564498376 \\ 0.99996776439177868437 \end{pmatrix}.$$

To improve the accuracy of singular solutions we further use the so-called *deflation* method [55, 86, 87]. For a general polynomial system $P(\mathbf{x}) = (p_1(\mathbf{x}), \ldots, p_n(\mathbf{x}))$ with $\mathbf{x} = (x_1, \ldots, x_n)$, let $\mathbf{x}^* = (x_1^*, \ldots, x_n^*)$ be a solution of $P(\mathbf{x}) = \mathbf{0}$ with multiplicity ≥ 2. Assume $DP(x^*)$, the Jacobian of $P(\mathbf{x})$ at x^*, is of rank $n - d$ where $0 < d < n$. (This d can be found efficiently by the technique developed in [56].) Then for almost all $d \times n$ random matrix R, the matrix

$$\begin{bmatrix} DP(\mathbf{x}^*) \\ R \end{bmatrix}$$

is of full column rank. Let $e_1 := (1, 0, \ldots, 0)^T \in \mathbb{R}^d$. It is clear that the linear system

$$\begin{bmatrix} DP(\mathbf{x}^*) \\ R \end{bmatrix} \mathbf{y} = \begin{bmatrix} 0 \\ e_1 \end{bmatrix}$$

has a unique solution $\mathbf{y} = \hat{\mathbf{y}}$ in $\mathbb{C}^n$. Then we construct a new $(2n + d) \times 2n$ system

$$Q(\mathbf{x}, \mathbf{y}) := \begin{bmatrix} P(\mathbf{x}) \\ \begin{bmatrix} DP(\mathbf{x}) \\ R \end{bmatrix} \mathbf{y} - \begin{bmatrix} 0 \\ e_1 \end{bmatrix} \end{bmatrix} = 0.$$

If $\hat{\mathbf{z}} := (\hat{\mathbf{x}}, \hat{\mathbf{y}})$ is a simple zero of $Q(\mathbf{z}) := Q(\mathbf{x}, \mathbf{y})$, $DQ(\hat{\mathbf{z}})$ must be of full rank. Denote

$$(DQ(\mathbf{z}))^\dagger := [(DQ(\mathbf{z}))^T (DQ(\mathbf{z}))]^{-1} (DQ(\mathbf{z}))^T.$$

Then the Gaussian-Newton iterations

$$\mathbf{z}^{(j+1)} = \mathbf{z}^{(j)} - (DQ(\mathbf{z}^{(j)}))^\dagger Q(\mathbf{z}^{(j)}) \quad \text{for } j = 0, 1, \ldots$$

with $\mathbf{z}^{(0)} := (\mathbf{x}^*, \hat{\mathbf{y}})$ can be used, until the residue $\|Q(\mathbf{z}^{(j+1)})\|_2$ is within the desired accuracy. This will lead to a more accurate approximation of $\mathbf{x}^*$.

If $\hat{\mathbf{z}}$ is a multiple zero of $Q(\mathbf{z}) := Q(\mathbf{x}, \mathbf{y})$, the deflation procedure given above can be repeated on $Q(\mathbf{z})$ until a satisfactory $\tilde{x}^*$ is achieved.

Returning to Example 14.2, evaluating the Jacobian of the system at the solution $x^* = (1, 1, 1, 1, 1)$, we have $d = 4$, and when the deflation method is used the solution can be approximated much accurately as shown below

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix} = \begin{pmatrix} 0.99999999999999999998 \\ 0.99999999999999999981 \\ 1.00000000000000000002 \\ 1.00000000000000000000 \\ 1.00000000000000000000 \end{pmatrix}.$$

14.2.2. The local geometry of a homotopy path. While, by this time, in the construction of the homotopies $H(\mathbf{x}, t)$ the parameter t is normally taken as a real variable and any solution path is a subset of $\mathbb{C}^n \times [0, 1]$, in the context of singular endgame, it is crucially important to recognize the rich geometric information that lie beyond the real parameter space $[0, 1]$. In particular, we need to consider t as a complex variable and H as a function $H : \mathbb{C}^n \times \mathbb{C} \to \mathbb{C}^n$. A solution path $\gamma \subset H^{-1}(\{\mathbf{0}\})$ is then a subset of $\mathbb{C}^n \times \mathbb{C}$.

Near a "regular point" $(\mathbf{x}_0, t_0) \in H^{-1}(\{\mathbf{0}\})$ where the Jacobian matrix $H_{\mathbf{x}}$ of H with respect to $\mathbf{x}$ is nonsingular, the geometric structure of a solution path γ is very simple: By the Implicit Function Theorem (for holomorphic functions), γ is locally parametrized by t in a small open disk of $\mathbb{C}$ centered at t_0 (via a holomorphic map). Therefore, locally, γ is simply a copy of an open disk in $\mathbb{C}$ embedded in $\mathbb{C}^n$ and hence disjoint from other paths. The geometry near a "singular point" is much more complicated, and to understand it requires more powerful machinery from several complex variables.

Here the solution path γ of $H(\mathbf{x}, t) = \mathbf{0}$ is a subset of $\mathbb{C}^n \times \mathbb{C}$ (locally) defined as the common zero set of polynomial or holomorphic functions, known as an *analytic set*. Since only local behavior of the path near a point is of interest, we will focus on the "germ of the analytic set" γ, which can be vaguely understood as an infinitely small part of a point on γ. Formally, two analytic sets V_1 and V_2 both containing a point $\mathbf{p}$ are said to be *equivalent* at $\mathbf{p}$ if there exists a neighborhood U of $\mathbf{p}$ for which $V_1 \cap U = V_2 \cap U$. For a fixed point $\mathbf{p}$, this defines an equivalence relation, and the resulting equivalence classes are known as the *germs of analytic sets at* $\mathbf{p}$. A germ of an analytic set at a point is *reducible* if it is the union of two proper germs of analytic sets at that point, and is *irreducible* otherwise. Despite being infinitesimal distillations of analytic sets, irreducible germs of analytic sets indeed have well defined dimensions.

In this context, the endgame can be understood as the geometric characterization of the irreducible germ of the solution path $\gamma \subset \mathbb{C}^n \times \mathbb{C}$ at its end

point $(\boldsymbol{\zeta}, 1)$ as well as the computation of a better estimate for this end point using these geometric information. The singular endgame hinges on a few nontrivial observations: γ determines an one-dimensional irreducible germ at its end point $(\boldsymbol{\zeta}, 1)$. By Local Parametrization Theorem [36] this irreducible germ can be realized as a finite branched covering over an open disk in $\mathbb{C}$ (commonly known as the "local uniformization"). Topologically, the finite branched covering must be isomorphic to the standard finite branched covering given by $z \mapsto z^m$ where m is the number of sheets. These observations are made precise by the following important theorem:

Theorem 14.3. *With the above notations, let* $(\boldsymbol{\zeta}, 1)$ *be an end point of the path* γ *in* $\mathbb{C}^n \times \mathbb{C}$. *Then each point* $(\mathbf{x}, t) = (x_1, \ldots, x_n, t)$ *on* γ *sufficiently close to* $(\boldsymbol{\zeta}, 1)$ *can be expressed by a fixed convergent power series of the form*

$$x_j = \sum_{k=0}^{\infty} a_{jk} s^k$$
$$t = 1 - s^m$$

for each $j = 1, \ldots, n$, *and some fixed* $m \in \mathbb{Z}^+$.

While power series expansions in the above form as well as the integer m are not unique (via change of variables $s = \sigma^k$ for some $k > 1$, one obtains a different power series expansion with a different value of m), it is important to note that the smallest such m is unique. This m is known as the *winding number* of the solution path γ at the end point $(\boldsymbol{\zeta}, 1)$.

The above formulation also reveals an important geometric property of the homotopy paths near its end point: Fix any m-th root of unity ω (i.e. $\omega^m = 1$). Under the substitution $s = \sigma\omega$, the corresponding t value remains the same since $t = 1 - s^m = 1 - \sigma^m\omega^m = 1 - \sigma^m$ regardless which m-th root of unity is chosen. In particular, as σ moves toward 0 along the real axis, the corresponding t converges to 1 at the same rate independent from the choice of ω. Yet the m different choices of the m-th root of unity ω would result in m distinct trajectories of $\mathbf{x}$-values as σ goes to 0 all converging to $(a_{10}, \ldots, a_{n0})$. Consequently, there are at least m distinct homotopy paths converging to the same end point.

Now it is possible that the solution path γ fails to converge to any point in $\mathbb{C}^n$. This may occur, by the smoothness, only when $\|\mathbf{x}\|$ grows boundlessly as $t \to 1$ along γ, the path is then characterized as "diverge to infinity". Theorem 14.3 can be generalized to include such cases.

Theorem 14.4. *With the same notations, for* $(\mathbf{x}, t) = (x_1, \ldots, x_n, t)$ *on* γ *as* t *close to* 1, *there are integers* $d_1, \ldots, d_n$, *possibly negative, and a positive integer* m *for which* $(\mathbf{x}, t)$ *on the path* γ *has a convergent Laurent series expansion of the form*

$$\begin{aligned} x_j &= \sum_{k=d_j}^{\infty} a_{jk} s^k \quad \text{for some } a_{jd_j} \neq 0 \\ t &= 1 - s^m \end{aligned}$$

for $j = 1, \ldots, n$ *and* $m \in \mathbb{Z}^+$.

Similarly, the smallest m in the above series expansion is unique and it is called the *winding number* of the (possibly divergent) solution path γ. If $d_j < 0$ for at least one $j \in \{1, \ldots, n\}$, the corresponding variable x_j would grow unboundedly as $s \to 0$ causing the path to diverge.

For a simpler expression in the discussions below as well as numerical considerations (see the next Remark), we change the path parameter via $t = 1 - \tau$. The paths defined by $H(\mathbf{x}, t) = H(\mathbf{x}, 1 - \tau) = \mathbf{0}$ are now parametrized by τ, and a standard notation of the end point of a path becomes $(\lim_{\tau \to 0} \mathbf{x}, 0)$. Furthermore, the Laurent series expansion of the path in Theorem 14.4 takes the form

$$\begin{aligned} x_j &= \sum_{k=d_j}^{\infty} a_{jk} s^k \\ \tau &= s^m. \end{aligned} \tag{14.4}$$

Remark 14.5. In numerical computation, *floating point* numbers are almost aways used for approximating real numbers (and complex numbers by extension). Floating point numbers, designed as a compromise between precision and range, have a distributions of varying density among the real numbers that depends on the magnitude. This distribution is biased toward zero. That is, there are more floating point numbers near zero than there are in the ranges of larger magnitude. Numbers near zero therefore can be approximated more accurately. Moreover, *denormal numbers* (a.k.a. *denormalized numbers* or *subnormal numbers*[4]) has been formalized in IEEE 754 standard to fill in the "underflow gap" (numbers that are too small to be represented

[4]While in the 2008 edition of the IEEE 754 standard, "subnormal numbers" became the official name, the term "denormal numbers" remain widely used in the community of numerical analysts.

in the normal floating point format). It is therefore beneficial to use the path parameter $\tau = 1 - t$ near the end point where $\tau = 0$.

A variety of endgames were developed based on the above formulation. We briefly outline some of the most commonly used techniques below.

14.2.3. Cauchy integral endgame. Developed in [83, 84], the "Cauchy integral endgame" is an efficient and effective technique for accurately approximating singular endpoints of a homotopy path. Assuming the path γ has a singular end point in $\mathbb{C}^n$ and let m be the winding number of this path. Following the above discussion, with an abuse of notations, we shall consider each $x_j = x_j(s)$ and $\tau = \tau(s)$ as holomorphic functions of a parameter s within some small neighborhood D of $s = 0$ (although for $m > 1$, x_j will not be holomorphic in the path parameter τ). The *Cauchy Integral Formula* thus provides an alternative means for computing end point of the path via

$$x_j(0) = \frac{1}{2\pi i} \oint_\Gamma \frac{x_j(s)}{s} ds$$

where Γ is a sufficiently small circle inside D_j and the contour integral is taken in a counter-clockwise direction. Moreover, the value of this integral is independent from the choice of the circle as long as $s = 0$ is the only singularity in the interior of Γ and each x_j remain holomorphic on Γ. Let r be the radius of the circle, then Γ can be parametrized by $s = re^{i\theta}$ with which $ds = ire^{i\theta} d\theta$. Therefore

$$\begin{aligned} x_j(0) &= \frac{1}{2\pi i} \oint_\Gamma \frac{x_j(s)}{s} ds = \frac{1}{2\pi i} \int_0^{2\pi} \frac{x_j(re^{i\theta})}{re^{i\theta}} ire^{i\theta} d\theta \\ &= \frac{1}{2\pi} \int_0^{2\pi} x_j(re^{i\theta}) d\theta. \end{aligned} \tag{14.5}$$

However, this integral is not directly computable, after all, the variable s does not appear in the formulation of the homotopies and hence cannot be manipulated independently. An indirect computation is nonetheless possible by observing that as s goes around the small circle Γ once (in the counter-clockwise), $\tau = s^m$ would move around a circle of a different radius m times. Therefore the sample values needed for approximating the integral in (14.5) can be obtained indirectly by tracking the movement of x_j for $j = 1, \ldots, n$ as τ circle around 0 for m times. In other words, assuming m is already

known, one can track along the solution path defined by

$$\widetilde{H}(\mathbf{x}, \theta) := H(\mathbf{x}, \tilde{r}e^{i\theta}) \tag{14.6}$$

where the path parameter τ is parametrized by $\tilde{r}e^{i\theta}$ starting from a point $(\mathbf{x}^{(0)}, \tau^{(0)})$ on the path γ. The points visited by the path tracking algorithm are recorded. After θ reaches $2m\pi$, that is, after τ goes around 0 along the circle m times, the corresponding s variable would have made one complete revolution (since $\tau = s^m$). Then with the sample points obtained by path tracking along the circle, the integral (14.5) can be approximated by numerical integration techniques. This yields an approximation of the end point $\boldsymbol{\zeta} = (x_1(0), \ldots, x_n(0))$.

A key question is the value of the winding number m which is usually unknown. Fortunately, this missing piece of information can be obtained as a by-product of the process that collects the sample points for computing the Cauchy integral. Observe that as θ in (14.5) reaches $2m\pi$ (at which point we terminate the path tracking), the corresponding s makes one full revolution around 0 and comes back to the very same complex number that corresponds to the starting point at $\theta = 0$. Consequently, the corresponding values of $x_1, \ldots, x_n$, having power series expressions in terms of s comes back to the same values as in the starting point. Recall that from a topological point of view m is the number of sheets in the irreducible germ of the path γ as a branched cover over $\tau = 0$. Subsequently, m is the least number of revolutions one must make before the corresponding x_j's come back to the same values. In other words, by collecting the points $(\mathbf{x}^{(0)}, 0)$, $(\mathbf{x}^{(1)}, 2\pi)$, $(\mathbf{x}^{(2)}, 4\pi), \ldots, (\mathbf{x}^{(k)}, 2k\pi), \ldots$, as the solution path defined by (14.5) is tracked, m is the smallest positive integer such that $\mathbf{x}^{(0)} = \mathbf{x}^{(m)}$.

14.2.4. Laurent series method for identifying divergent paths. In this section, a technique for identifying divergent paths using the Laurent series formulation in (14.4) is presented. By "factoring out" the leading terms (terms with the lowest power), from each x_j in (14.4), it follows that

$$x_j = \sum_{k=d_j}^{\infty} a_{jk} s^k = a_{jd_j} s^{d_j} \left(1 + \sum_{k=1}^{\infty} \frac{a_{jk}}{a_{jd_j}} s^k \right).$$

Clearly, the path diverges (to infinity) if any of the d_j is negative. This method relies on the numerical identification of the signs of d_j's.

Within the (punctured) disk of convergences, by taking a fixed branch of the complex logarithm function, one obtains

$$\log x_j = \log a_{jd_j} + d_j \log s + \log\left(1 + \sum_{k=1}^{\infty} \frac{a_{jk}}{a_{jd_j}} s^k\right)$$
$$\log \tau = m \log s$$

which are holomorphic away from $s = 0$. Since $1 + \sum_{k=1}^{\infty} \frac{a_{jk}}{a_{jd_j}} s^k \to 1$ as $s \to 0$, $\log\left(1 + \sum_{k=1}^{\infty} \frac{a_{jk}}{a_{jd_j}} s^k\right)$ is holomorphic at $s = 0$ and has a convergent power series expansion of the form

$$\log\left(1 + \sum_{k=1}^{\infty} \frac{a_{jk}}{a_{jd_j}} s^k\right) = c_1 s^1 + c_2 s^2 + \cdots .$$

Moreover, substituting $\log s = \frac{\log \tau}{m}$ yields

$$\log x_j = \log a_{jd_j} + \frac{d_j}{m} \log \tau + c_1 s^1 + c_2 s^2 + \cdots .$$

By taking two sets of approximations of $\log x_1, \ldots, \log x_n, \log \tau$ along the path, say $(\log x_j^{(1)})_{j=1}^n$ and $(\log x_j^{(2)})_{j=1}^n$ along with $\tau^{(1)}$ and $\tau^{(2)}$ respectively, one can approximate

$$\log x_j^{(1)} - \log x_j^{(2)} = \frac{d_j}{m}(\log \tau^{(1)} - \log \tau^{(2)}) + \mathcal{O}(\tau^{\frac{1}{m}}) \tag{14.7}$$

where $\mathcal{O}(\tau^{\frac{1}{m}})$ denotes the collection of terms that will vanish with order $\frac{1}{m}$ as $|\tau| \to 0$. Therefore, the approximation can be taken as

$$\frac{\log x_j^{(1)} - \log x_j^{(2)}}{\log \tau^{(1)} - \log \tau^{(2)}} \approx \frac{d_j}{m} \tag{14.8}$$

as long as a suitable branch of logarithm is used. If it is determined that $\frac{d_j}{m} < 0$ for some j, the path is divergent and can therefore be discarded as it will not converge to a point in $\mathbb{C}^n$.

Clearly the formulation in (14.8) may be unstable near the end of a divergent path (which we want to identify) since τ would approach 0 and at least one of the $x_j \to \infty$. More practical numerical improvements on this scheme were developed in [44] and [53]. The accuracy of this estimate is limited by the winding number m since the $\mathcal{O}(\tau^{\frac{1}{m}})$ terms in (14.7) have

been discarded in the derivation. For larger m values, (14.8) becomes a poor estimate. The improvement on this scheme for handling divergent paths with larger winding number is still an open problem at this time.

14.3. Projective path tracking

Though the various methods of affine path tracking in $\mathbb{C}^n$ described in §4 are sufficient in dealing with many problems, they may not be able to differentiate between solution paths that escape $\mathbb{C}^n$ and "diverge to infinity" or solution paths that converge to end points with very large norms. In §14.2.4 Laurent series (or, equivalently, Puiseux series) expansions with negative exponents are used to identify divergent solution paths near their end points. In this section we present a much stronger environment where the entire path tracking process is carried out in a "compactified" space where solution paths cannot diverge.

Divergent paths exist, in part, because $\mathbb{C}^n$ is not compact as a topological space. If $\mathbb{C}^n$ is replaced by a compact topological space W, a *compactification* of $\mathbb{C}^n$, in which $\mathbb{C}^n$ is embedded as a dense subset, then all homotopy paths, now in $W \times [0,1]$, must converge to points inside W at $t = 1$ and have finite arc length [58]. One of the most commonly used compactification of $\mathbb{C}^n$ in the context of algebraic geometry is the complex projective space $\mathbb{CP}^n$. Recall that

$$\mathbb{CP}^n = (\mathbb{C}^{n+1} \backslash \{(0, \ldots, 0)\}) / \sim$$

where $\mathbf{x} \sim \mathbf{y}$ for $\mathbf{x}, \mathbf{y} \in \mathbb{C}^{n+1}$ if $\mathbf{x} = \lambda \mathbf{y}$ for $\lambda \in \mathbb{C} \setminus \{0\}$, and points of $\mathbb{CP}^n$ are one dimensional linear subspaces of $\mathbb{C}^{n+1}$ with "origin" removed. The notation $[x_0 : \cdots : x_n]$ is commonly used for the **homogeneous coordinate** of a point in $\mathbb{CP}^n$ with $[x_0 : \cdots : x_n]$ being equivalent to $[\lambda x_0 : \cdots : \lambda x_n]$ for any $\lambda \in \mathbb{C} \setminus \{0\}$. With such coordinates, $\mathbb{CP}^n$ can be covered by subsets $U_j = \{[x_0 : \cdots : x_n] \,|\, x_j \neq 0\}$ for $j = 0, \ldots, n$, called **standard charts**. Clearly, each standard chart U_j is isomorphic to $\mathbb{C}^n$, as a set. These charts equip the set $\mathbb{CP}^n$ with a $2n$-dimensional smooth manifold structure (as well as an n-dimensional complex manifold structure).

The zero sets of polynomials in $\mathbb{CP}^n$ are not well defined in general since each point in $\mathbb{CP}^n$ has infinitely many different coordinates. However, recall that given any polynomial $f \in \mathbb{C}[x_1, \ldots, x_n]$ of degree d, its **homogenization**

$$\hat{f}(x_0, \ldots, x_n) = x_0^d \cdot f\left(\frac{x_1}{x_0}, \ldots, \frac{x_n}{x_0}\right)$$

has the property that for $\mathbf{x} = (x_0, \ldots, x_n)$, $\hat{f}(\lambda \cdot \mathbf{x}) = \lambda^d \cdot \hat{f}(\mathbf{x})$. Hence the zero set of $\hat{f}$ is well defined in $\mathbb{CP}^n$, since for any $\lambda \neq 0$, $\hat{f}(\lambda \cdot \mathbf{x}) = 0$ if and only if $\hat{f}(\mathbf{x}) = 0$. Furthermore, since $\hat{f}(1, x_1, \ldots, x_n) = f(x_1, \ldots, x_n)$, then, whenever $x_0 \neq 0$, there is a one to one correspondence between the zero sets of $\hat{f}$ and f. This common construction allows us to "lift" a problem into the complex projective space. In particular, for a given homotopy $H = (h_1, \ldots, h_n)$, its homogenization $\hat{H}(x_0, x_1, \ldots, x_n) = (\hat{h}_1, \ldots, \hat{h}_n)$ with respect to the variables $(x_1, \ldots, x_n)$ can be considered as a homotopy construction that defines paths in $\mathbb{CP}^n \times [0, 1]$ which will simply be called **projective paths**. Note that these "projective" paths are closely related to the original "affine" paths defined by $H(\mathbf{x}, t) = \mathbf{0}$ in the sense that for any affine path $\gamma \subset \mathbb{C}^n \times (0, 1)$, the corresponding path $\hat{\gamma} = \{([1, x_1, \ldots, x_n], t) \mid (x_1, \ldots, x_n, t) \in \gamma\} \subset \mathbb{CP}^n \times (0, 1)$ must satisfy the equation $\hat{H} = \mathbf{0}$. The path $\hat{\gamma}$ will be called a **projective path** corresponds to γ. A main advantage of working in $\mathbb{CP}^n$ is its *compactness* as a topological space, thus all projective paths defined by $\hat{H} = \mathbf{0}$ must converge and have finite length. In working with $\mathbb{CP}^n$, it is particularly convenient to use the unit sphere S^{2n+1} as the model of computation via the well known consideration of $\mathbb{CP}^n$ as the quotient manifold S^{2n+1}/S^1 which we shall briefly review.

Let $S^{2n+1} = \{\mathbf{x} \in \mathbb{C}^{n+1} : \|\mathbf{x}\|_2 = 1\}$ be the unit sphere of $\mathbb{C}^{n+1}$, which is a smooth manifold of $2n + 1$ (real) dimension. It is standard to view $\mathbb{CP}^n$ as the quotient of S^{2n+1} under the action of the circle group: First of all, each point $(x_0, \ldots, x_n) \in S^{2n+1}$ represents a point in $\mathbb{CP}^n$ via the map $\pi : S^{2n+1} \to \mathbb{CP}^n$ given by $(x_0, \ldots, x_n) \mapsto [x_0 : \cdots : x_n]$, which is clearly onto. However, the representative of a point in $\mathbb{CP}^n$ is not unique, i.e., π is not 1-to-1, as $\pi(\mathbf{x}) = \pi(\lambda\,\mathbf{x})$ for any $\lambda \in \mathbb{C}^*$. To leave S^{2n+1} invariant, we must have $|\lambda| = 1$, i.e., $\lambda = e^{i\theta}$. So for $\mathbf{x} \in S^{2n+1}$, the points of the form $e^{i\theta}\mathbf{x}$ with $\theta \in \mathbb{R}$ are exactly those that represent the same point as $\mathbf{x}$ does. Therefore, $\mathbb{CP}^n$ can be identified with the set of equivalent classes $\{[\mathbf{x}] : \mathbf{x} \in S^{2n+1}\}$ where

$$[\mathbf{x}] := \{\, e^{i\theta}\mathbf{x} \mid \theta \in \mathbb{R} \,\}.$$

In fact, this identification is more than set theoretical. Let $S^1 = \{e^{i\theta} \mid \theta \in \mathbb{R}\}$ be the unit circle of $\mathbb{C}$. With it, the set $[\mathbf{x}]$ can be considered as the orbit of $\mathbf{x}$ under the action of a compact Lie group S^1. This identifies $\mathbb{CP}^n$ with the quotient S^{2n+1}/S^1. This quotient is a smooth manifold in its own right. On the other hand, it has a unique smooth structure for which π is a smooth submersion. With this smooth structure, one can show that S^{2n+1}/S^1 is diffeomorphic to $\mathbb{CP}^n$ whose smooth structure is given by the standard charts. Furthermore, since S^{2n+1} is a Riemannian manifold, with its Riemannian

metric $g_{S^{2n+1}}$ inherited from the standard inner product of $\mathbb{C}^{n+1} \approx \mathbb{R}^{2n+2}$, the quotient map π also gives us a natural choice of the Riemannian metric on $\mathbb{CP}^n \approx S^{2n+1}/S^1$. Since π is a submersion, at each point $\mathbf{x} \in S^{2n+1}$, its pushforward π_* has a constant rank of $2n$. Its kernel $\mathcal{V}_\mathbf{x} \subset T_\mathbf{x} S^{2n+1}$, of real-dimension 1, is known as the **vertical space**, which is simply the tangent space of the fiber over $\pi(\mathbf{x}) = [\mathbf{x}]$. Its orthogonal complement with respect to $g_{S^{2n+1}}$

$$\mathcal{H}_\mathbf{x} = \{\mathbf{h} \in T_\mathbf{x} S^{2n+1} \mid g_{S^{2n+1}}(\mathbf{h}, \mathbf{v}) = 0 \; \forall \; \mathbf{v} \in \mathcal{V}_\mathbf{x}\}$$

is known as the **horizontal space**, and it is a representation of the tangent space of the quotient S^{2n+1}/S^1. There is a unique Riemannian metric g, called **Fubini-Study metric**, on $\mathbb{CP}^n$, such that π is also a *Riemannian submersion*, i.e., at each point $\mathbf{x} \in S^{2n+1}$,

$$g_{S^{2n+1}}(\mathbf{h}_1, \mathbf{h}_2) = g(\pi_*(\mathbf{h}_1), \pi_*(\mathbf{h}_2))$$

for any $\mathbf{h}_1, \mathbf{h}_2 \in \mathcal{H}_\mathbf{x}$. In other words, π_* is an isometry on the horizontal space $\mathcal{H}_\mathbf{x}$.

The benefit of using S^{2n+1} as our model of computation (i.e., points of S^{2n+1} are used to represent points of $\mathbb{CP}^n$) is that, in additional to being compact, all points in S^{2n+1} have coordinates with norm 1, a numerically favorable environment. Note that the "points at infinity" are represented by points $\mathbf{x} = (x_0, x_1, \ldots, x_n) \in S^{2n+1}$ with $x_0 = 0$ much like the situation with the homogeneous coordinates. To track a smooth solution path $\hat{\gamma} \subset \mathbb{CP}^n \times [0, 1]$ defined by $\hat{H} = \mathbf{0}$ with parametrization $\hat{\mathbf{x}} : [0, 1] \to \mathbb{CP}^n$, it is sufficient to track a representation $\mathbf{x} : [0, 1] \to S^{2n+1}$ in S^{2n+1} of the projective path $\hat{\mathbf{x}}$ in the sense that $\pi(\mathbf{x}(t)) = \hat{\mathbf{x}}(t)$ for all $t \in [0, 1]$. Unfortunately, there are infinitely many such representations in S^{2n+1}. In particular, if $\mathbf{x} : [0, 1] \to S^{2n+1}$ is such a representation, then so is

$$\mathbf{x}^{(1)}(t) = e^{i \cdot \theta(t)} \mathbf{x}(t)$$

for any smooth function $\theta : [0, 1] \to \mathbb{R}$. While, in principle, any choice of the representation would allow us to obtain our desirable end point $\hat{\mathbf{x}}(1)$, the Riemannian geometry of $\mathbb{CP}^n$ suggests a natural choice: the *horizontal lift* of $\hat{\mathbf{x}}$. Given a starting point $\mathbf{x}^{(0)} \in S^{2n+1}$ representing $\hat{\mathbf{x}}(0) \in \mathbb{CP}^n$, the *horizontal lift* $\mathbf{x} : [0, 1] \to S^{2n+1}$ is the unique smoothly parametrized curve

$$\begin{aligned} \mathbf{x}(0) &= \mathbf{x}^{(0)} \\ \dot{\mathbf{x}}(t) &\in \mathcal{H}_{\mathbf{x}(t)} \\ D_\mathbf{x} \hat{H}(\mathbf{x}(t), t) \dot{\mathbf{x}}(t) &= -D_t \hat{H}(\mathbf{x}(t), t) \end{aligned} \tag{14.9}$$

This is the projective analog of the DAVIDENKO differential equation.

Intuitively, this choice is a representation whose tangent vector is always orthogonal to the fiber direction. Concerning Riemannian geometry, this choice is indeed a natural one, because the submersion π acts as an isometry along such a curve. In addition, there are three more properties that justify this choice: First, over each infinitesimal t-interval, the horizontal lift has the minimum length among all smooth representations of $\hat{\gamma}$ in S^{2n+1}, which is certainly a desirable property. Second, when the Fubini-Study metric is used, the horizontal lift has exactly the same length as $\hat{\gamma}$. Hence this choice of representation does not artificially stretch the curve in length. Finally, as an arguably more important benefit for numerical algorithms, this choice has the best numerical condition among all the representations.

Using $\mathbb{C}^{n+1}$ as the ambient space, at each fixed $\mathbf{x} \in S^{2n+1} \subset \mathbb{C}^{n+1}$, the horizontal space $\mathcal{H}_{\mathbf{x}}$ has a simple numerical description: Via the isomorphism $T_{\mathbf{x}}\mathbb{C}^{n+1} \cong \mathbb{C}^{n+1}$, $\mathcal{H}_{\mathbf{x}}$ is given by the subspace

$$\mathcal{H}_{\mathbf{x}} = \{\mathbf{v} \in \mathbb{C}^{n+1} \mid \langle \mathbf{x}, \mathbf{v} \rangle_{\mathbb{C}} = \mathbf{x}^H \mathbf{v} = 0\}$$

where $\mathbf{x}^H$ is the conjugate transpose of vector $\mathbf{x}$. Notice that this characterization of $\mathcal{H}_{\mathbf{x}}$ is invariant under the group action of S^1, since if $\langle \mathbf{x}, \mathbf{v} \rangle_{\mathbb{C}} = 0$, then $\langle e^{i\theta}\mathbf{x}, \mathbf{v} \rangle_{\mathbb{C}} = 0$ for any $e^{i\theta} \in S^1$. With this formulation, the projective Davidenko differential equation (14.9) can be expressed in coordinate as

$$\begin{pmatrix} D_{\mathbf{x}}\hat{H}(\mathbf{x}, t) \\ \mathbf{x}^H \end{pmatrix} \cdot \dot{\mathbf{x}} = \begin{pmatrix} -D_t\hat{H}(\mathbf{x}, t) \\ 0 \end{pmatrix}. \tag{14.10}$$

It is clear that under the smoothness condition of the homotopy $\hat{H}$, the above system of ODE uniquely determines the tangent vector $\dot{\mathbf{x}}$ at each point along the curve $\mathbf{x}(t)$. So the projective path tracking can be reduced to the initial value problem given by (14.10) on the Riemannian manifold S^{2n+1}. This forms the foundation to establish the projective path tracking algorithm. In the following subsections we will outline the basic building blocks of the algorithm.

14.3.1. Spherical projective Euler's predictor. Given a point $\mathbf{x} = \mathbf{x}(t_0) \in S^{2n+1}$ on (or close to) a horizontal lift of a projective path and a step size Δt, the task of a predictor is to produce an approximation of the point on the path at $t = t_0 + \Delta t$. In light of Equation (14.10), with the ability to compute tangent vectors, almost any curve fitting or extrapolation scheme on the sphere S^{2n+1} can be used as predictors. For simplicity, we shall focus on the generalization of Euler's method.

A geometric interpretation of Euler's method in (4.4) is the movement of a point along the straight line defined by the tangent vector by certain step length. The analogue in the context of Riemannian geometry is the exponential map $\mathrm{Exp} : TS^{2n+1} \to S^{2n+1}$

$$\mathrm{Exp}(\mathbf{x}, \mathbf{v}) := \gamma_{\mathbf{v}}(1)$$

where $\gamma_{\mathbf{v}} : \mathbb{R} \to S^{2n+1}$ is a Riemannian geodesic such that $\gamma_{\mathbf{v}}(0) = \mathbf{x}$ and $\dot{\gamma}_{\mathbf{v}}(0) = \mathbf{v}$. It moves a point $\mathbf{x} \in S^{2n+1}$ along a Riemannian geodesic passing though that point with the given initial tangent vector $\mathbf{v} \in T_{\mathbf{x}}S^{2n+1}$ for a step of unit length within the confine of S^{2n+1}. On S^{2n+1}, one can verify that the geodesic with initial tangent vector $\mathbf{v}$ is simply given by

$$\gamma_{\mathbf{v}}(t) = \cos(\|\mathbf{v}\|_2 t)\mathbf{x} + \sin(\|\mathbf{v}\|_2 t)\mathbf{v}/\|\mathbf{v}\|_2.$$

Therefore, in this context, the exponential map is given by

$$\mathrm{Exp}(\mathbf{x}, \mathbf{v}) = \cos(\|\mathbf{v}\|_2)\mathbf{x} + \sin(\|\mathbf{v}\|_2)\mathbf{v}/\|\mathbf{v}\|_2.$$

One can construct the generalized Euler's method out of a scaled version of the exponential map: define **spherical projective Euler's prediction** $\mathcal{E}_{\mathrm{Exp}} : S^{2n+1} \times \mathbb{R} \to S^{2n+1}$ by

$$\mathcal{E}_{\mathrm{Exp}}(\mathbf{x}, \Delta t) := \cos(\|\dot{\mathbf{x}}\|_2 \Delta t)\mathbf{x} + \sin(\|\dot{\mathbf{x}}\|_2 \Delta t)\dot{\mathbf{x}}/\|\dot{\mathbf{x}}\|_2 \tag{14.11}$$

where Δt is the step size. It is easy to verify that $\mathcal{E}_{\mathrm{Exp}}(\mathbf{x}, 0) = \mathbf{x}$, $\mathcal{E}_{\mathrm{Exp}}(\mathbf{x}, \Delta t) \in S^{2n+1}$, and the Riemannian distance between $\mathbf{x}$ and $\mathcal{E}_{\mathrm{Exp}}(\mathbf{x}, \Delta t)$ is exactly $\|\dot{\mathbf{x}}\|_2 \cdot \Delta t$ for any $\Delta t \geq 0$, agreeing with our intuition.

14.3.2. Spherical projective Newton's corrector. The prediction $(\mathbf{x}', t_0 + \Delta t)$ produced by projective Euler's predictor may not be exactly on or even very close to the projective path defined by $\hat{H} = \mathbf{0}$. If the next prediction step is to start from such an approximation, the error can quickly build up to an unacceptable level. To curb such error accumulation, a *corrector* is needed to produce a refinement $\mathbf{x}''$ of the approximate solution $\mathbf{x}'$ of $\hat{H} = \mathbf{0}$ at $t_1 = t_0 + \Delta t$. When a corrector fails to bring the prediction back to the path quickly and reliably, the prediction should be performed again with a smaller step size.

A natural choice of the corrector is an extension of Newton's iteration to the sphere in the same way the spherical Euler's method is constructed via the exponential map. Starting from the prediction $\mathbf{x}^{(1)} = \mathbf{x}'$ provided by the

spherical Euler's method, a sequence of points $\mathbf{x}^{(2)}, \mathbf{x}^{(3)}, \ldots$ will be produced iteratively, we wish they will converge to some approximated solution $\mathbf{x}''$ of $\hat{H} = \mathbf{0}$ at $t = t_1$. For the k-th iteration, the Newton direction $\Delta\mathbf{x}^{(k)}$ is given via the linear system

$$\begin{pmatrix} \hat{H}_{\mathbf{x}}(\mathbf{x}^{(k-1)}, t_1) \\ (\mathbf{x}^{(k-1)})^H \end{pmatrix} \cdot \Delta\mathbf{x}^{(k)} = \begin{pmatrix} -\hat{H}(\mathbf{x}^{(k-1)}, t_1) \\ 0 \end{pmatrix}$$

which came from the "projective Newton's method" developed in [96]. Considering the vector $\Delta\mathbf{x}^{(k)}$ as a horizontal tangent vector in $\mathcal{H}_{\mathbf{x}}$, the spherical Newton's iteration is defined as

$$\begin{aligned} \mathcal{N}_{\mathrm{Exp}}(\mathbf{x}^{(k-1)}) := \cos(\|\Delta\mathbf{x}^{(k)}\|_2)\mathbf{x}^{(k-1)} \\ + \sin(\|\Delta\mathbf{x}^{(k)}\|_2)\Delta\mathbf{x}^{(k)}/\|\Delta\mathbf{x}^{(k)}\|_2. \end{aligned} \tag{14.12}$$

Using this map, we can produce points

$$\mathbf{x}^{(k)} = \mathcal{N}_{\mathrm{Exp}}(\mathbf{x}^{(k-1)})$$

for $k = 1, 2, \ldots$ until certain convergence criteria are met. The exact convergence criteria are implementation dependent. The Riemannian distance $d_{S^{2n+1}}(\mathbf{x}^{(k)}, \mathbf{x}^{(k-1)})$ between consecutive points $\mathbf{x}^{(k)}$ and $\mathbf{x}^{(k-1)}$ or, in general, $d_{S^{2n+1}}(\mathbf{x}^{(k)}, \mathbf{x}^{(k-j)})$ for some $j \in \mathbb{N}$ serve as useful stopping criteria, since the shrinking of these distances is usually a good indication of convergence. Here we refer to [53] for a list of the stopping criteria as well as their detailed descriptions.

Remark 14.6. Note that the spherical projective Newton's method proposed here is quite different from the "Projective Newton's method" introduced in [13] and [96]. In the first place, the spherical projective Newton's method uses the exponential map. Secondly, while the spherical projective Newton's method is used as the corrector in the predictor-corrector scheme here, [13] and [96] uses Projective Newton's method alone to track the paths.

15. Parallel mixed cells enumeration

Modern scientific computing is marked by the advent of vector and parallel computers and search for algorithms that are to a large extend parallel in nature. A great advantage of the homotopy continuation algorithm for solving polynomial systems is, it is to a large degree parallel in the sense that each isolated zero can be computed independently. In this respect, it stands

in contrast to the highly serial algebraic elimination methods, which use resultants or Gröbner bases. On the other hand, to attain more computing resources for solving larger polynomial systems, the parallelization of the homotopy method becomes inevitably essential.

The landscape of computation hardware has seen extremely active developments in recent years making available a wide spectrum of exciting new technologies. First, developments in new processor design and network technology have allowed supercomputers and computer clusters to grow larger and faster than ever. Second, new ideas such as cycle-scavenging and grid computing has led to the creation of virtual supercomputers out of large numbers of individual computers around the globe. Another exciting development is the advent of parallel computing on GPUs (Graphical Processing Units). While originally designed to handle 2D and 3D graphics rendering only, over the years GPUs have become sufficiently sophisticated to handle a much wider range of problems. Highly parallel by design, GPUs are more efficient than general purpose CPUs in carrying out a range of complex algorithms. Living in such interesting times is exciting and daunting. We must rise up to the challenge, fully incorporate all these cutting-edge parallel computing technology, and solve larger and larger polynomial systems.

As mentioned above, the "path tracking" part of the homotopy continuation method is *pleasantly parallel*, since each path can be tracked independent from one another. In the context of polyhedral homotopy (§7), however, the main preprocessing step of "mixed cell enumeration", detailed in §8, appears to be quite serial and is potentially a major bottleneck for the parallel scalability. Based on the idea of reformulating the problem into a graph-theoretic search problem, a fully parallel mixed cell enumeration algorithm that is efficient, robust, and highly scalable has been developed in Hom4PS-3 [15]. In this section, we briefly explain this algorithm.

In the main algorithm for mixed cell enumeration described in §8: while some of the one-point tests are closely related, most of the other one-point tests are independent from one another. Based on this observation, the mixed cell enumeration algorithm have since been modified to a parallel algorithm developed in [16] rooted from classical algorithms in graph theory.

In the reformulation, related one-point tests can be group together to form "tasks": a **task** is a series of one-point tests (8.4) originated from the same subface. Namely, they are the sets of one-point tests of the form $LP(F,*) := \{LP(F,\mathbf{b}) : \mathbf{b} \in \hat{S}_j\}$. For instance, all one-point tests originated from the subface $(\{\hat{\mathbf{a}}, \hat{\mathbf{a}}'\}, \{\hat{\mathbf{b}}, \hat{\mathbf{b}}'\})$ will be grouped together to form a task denoted by $LP((\{\hat{\mathbf{a}}, \hat{\mathbf{a}}'\}, \{\hat{\mathbf{b}}, \hat{\mathbf{b}}'\}), *)$. Such tasks will be our smallest units of computation around which the parallel algorithm is designed.

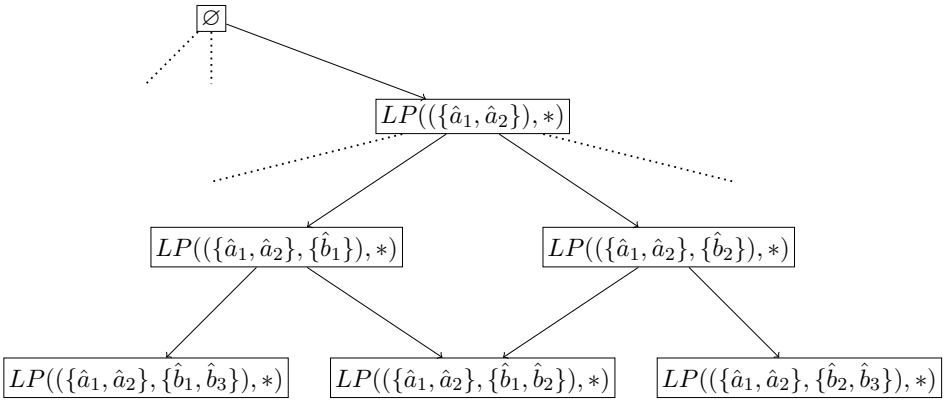


Figure 26. A direct acyclic graph containing tasks

Tasks are interconnected in such a way that they form a *direct acyclic graph* or DAG, whose vertices are the tasks and edges between vertices are given by the natural *extension* relation between subfaces of the tasks as elaborated in §8. Figure 26 depicts an example of such graphs. In this way, a graph representation of the totality of the one-point tests is produced. Of course, some of the one-point tests in the graph will be infeasible. With this connection, the mixed cell enumeration process via one-point tests can be understood as a special case of the *graph traverse problem* (the problem of visiting vertices of a graph by walking along edges connecting them). One important distinction is that in the mixed cell enumeration process there is no need to visit every vertex. Recall that by advanced techniques discussed in §8.3.3, §8.4, §8.5, and §8.3, results of many one point tests can quickly be attained. As a consequence, only a small fraction of one-point tests need to be solved formally. This specialized graph traverse problem can be handled by *graph traversal* algorithms following a "discover-explore" procedure with a proper book keeping. The theory behind such graph traverse algorithms can be found in standard textbooks such as [97].

In order to keep track of the progress of the exploration and coordinate multiple threads,[5] the collection of discovered (but not yet completely explored) vertices are stored in a data structure called "task pool". Though a number of data structures can be used, Hom4PS-3 chooses to use a *priority queue* to maintain the task pool which provides fine-grained control of the

[5]Here a "thread" refers to the smallest unit of a sequence of instructions that can be executed independently by the processor.

exploration process, e.g., *depth-first-search* versus *breadth-first-search*. We refer to [16] for details.

Multiple threads will operate on the task pool and perform one point tests simultaneously: Each thread repeatedly fetches a single task from the pool and explores it by performing a series of one-point tests. This "fetch-and-explore" procedure continues until the task pool is empty and there is no tasks that are currently being explored. At this point the feasible one point tests in the DAG are completely explored and all the mixed cells have been obtained. The algorithm then terminates. Since multiple threads will access the priority queue concurrently, operations on the queue must be made thread-safe to prevent *race condition* [41] (a condition in which multiple threads access the same data structure resulting in catastrophic data corruption). In [16], "mutex" ("mutual exclusion", a standard mechanism commonly used to ensure only one thread has access to a data structure) was proposed to guard the task pool and prevent race conditions. A much more efficient and scalable solution using "concurrent data structure" was later adopted in Hom4PS-3.

On multi-core systems, the implementation of this algorithm, based on Intel TBB (and optionally OpenMP), in Hom4PS-3 has achieved remarkable efficiency and scalability. Nearly n-fold linear speedups scalable up to 64 processor cores have been exhibited in experiments on standard test suite problems. Figure 27 shows the speedup ratio on the standard benchmark problem cyclic-15 [12].

This general parallel algorithm for mixed cell enumeration can be further modified to adapt to other parallel architectures including *NUMA systems*, *computer clusters*, *distributed environments*, and *GPU devices* (See. [16]).

15.1. On-the-fly NUMA optimization

Modern shared-memory systems with a large number of processor cores usually adopt a Non-Uniform Memory Access [41], or NUMA. In this architecture each processor core can access all the available memory with different speeds, depending on the relative closeness between the core and memory. Developed in the 1990s as an answer to the scalability limitation in the traditional SMP (symmetric multiprocessing) architectures, it has gained a great popularity in the world of high performance computing especially when AMD and Intel adopted the technology under the names HyperTransport (2003) and QPI (2007) respectively.

Figure 28 shows the "memory-processor topology" of a NUMA system that consists of 8 nodes. Each node contains 4 processor cores as well as

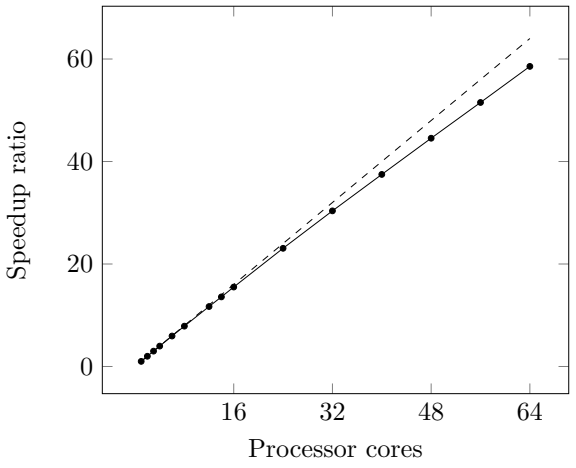


Figure 27. Speedup ratio achieved on a 64 core system (AMD Opteron with 512GB memory) for the cyclic-15 [12] problem showing close to n-fold linear speedups for up to 64 cores. The speedup is computed in comparison with the fastest serial implementations published: MixedVol-2.0 [52] and DEMiCs [77, 78].

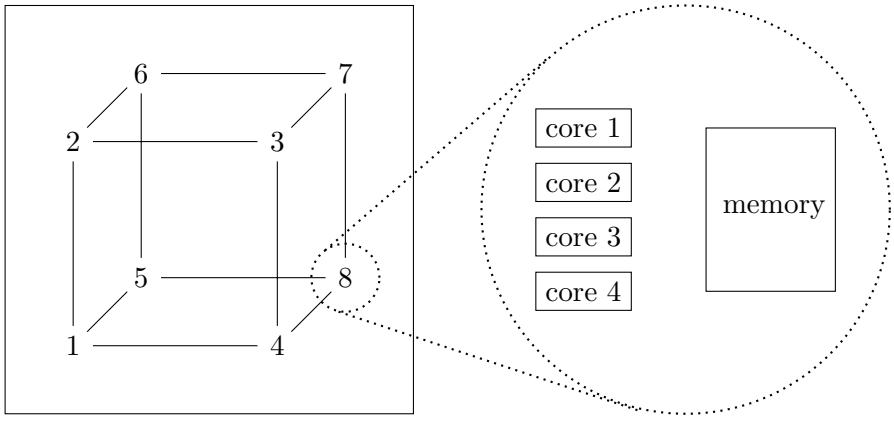


Figure 28. An example of a NUMA node structure

their "local" memory which they can access at full speed. The edges between nodes indicate the direct connectedness between nodes and determine the speed at which processor cores on one node can access memory on other

nodes. For instance a processor on node 1 can access the memory on node 2 at a slower rate than it could access its local memory on node 1. The same core can access memory on node 3 at a even slower rate due to the minimum two jumps required (through node 2 or 4). Similarly there are at least three jumps between node 1 and node 7. Consequently, that processor core would have the slowest memory access to memory on node 7.

Recall that most of the data required by our algorithm for mixed cell enumeration reside in the task pool. On the NUMA system, it is therefore crucial to split a single task pool into several task pools shared by the threads in an optimized pattern that ideally matches the underlying memory-processor topology. The planning of this pattern is governed by two conflicting constraints:

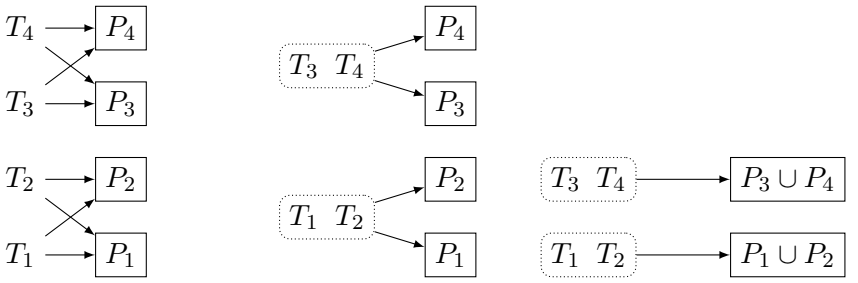
1) Each thread should access a task pool that is placed as close as possible in terms of memory-processor topology to optimize the memory access time.

2) Each task pool should be shared by as many threads as possible to avoid load balancing issues (to be discussed in detail in §15.2).

Unfortunately there is no standardized method currently available to determine precisely the memory-processor topology [41]. In particular, while one could inquire from the operating system which memory access patterns are slow, but not how slow. Coupled with the fact that the operating system can, at any time, migrate a running thread from one processor core to another, a dynamic and on-the-fly planning of the task pool placement and sharing pattern is therefore a necessity. Here we briefly outline the procedure.

At the beginning of the extension process, the program starts with an initial "evaluation phase" in which each thread is spawned with its own task pool in the memory local to the processor core that the thread runs on.[6] During the extension process, each thread will access tasks from *all* task pools. The average time for accessing each task pool is monitored, and the resulting data is used to construct a list of "preferred" task pools for having best access time. (see for example Figure 29a). Then threads that prefer the same task pool are grouped into "thread clusters" (see Figure 29b for the formation of clusters). Conversely task pools that are preferred by the same cluster of threads are merged (see Figure 29c for the merger of task pools).

[6]On Linux, this is done via the standard library libnuma provided by most Linux distributions. On Unix this step requires the correct configuration to be set by the user.



(a) Four threads T_1,T_2,T_3,T_4 with their preferred task pools P_1, P_2, P_3, P_4.

(b) Two clusters of threads formed based on common preference on task pools.

(c) Task pools preferred by the same clusters are merged together.

Figure 29. Dynamic optimization of the task pool sharing pattern on NUMA architectures

This process is repeated until the threads' preference of task pools stabilizes. At this point each cluster of threads has a single preferred task pool that has the best access time and this "evaluation phase" is terminated. Afterwards, this task pool sharing pattern is fixed and each thread only access its own preferred task pool. This optimization procedure can be performed again whenever threads have migrated to different processor cores or certain task pool becomes empty before others.

Incurring minimum additional computational cost at the initial "evaluation phase", this technique substantially improves the memory access time on NUMA systems. As shown in Table 3, in experiments on standard benchmark problems: the cyclic family [12], the five-body central configuration problem (fivebody) [3, 37, 54], and the 6-vortex problem (vortexAC6) [38, 111], approximately 1.5× to 20× speedup (150% to 2000%) in memory access time[7] have been observed which resulted in 5% to 35% overall speedups in the extension process.

Remark 15.1. The sensitivity in the overall run time to memory access pattern exhibited in Table 3 highlights the possibility that for sufficiently large systems, the mixed cell enumeration problem will become memory-bound. That is, the run time will no longer be dominated by the number of floating point operations but will instead be dominated by the memory

[7]The memory access time is approximated by using the "memory access latency" provided by the Intel VTune software which closely correlates to the actual memory access time that is generally difficult to measure. For the best accuracy, all CPU caches were disabled when measuring memory access latency.

access latency, an important factor that is often ignored in complexity analysis.

System	Mem. acc. speedup	Overall reduction
cyclic-14*	1.40	4.9%
cyclic-15*	2.40	8.2%
cyclic-16*	4.50	9.5%
fivebody	17.55	33.2%
vortex-6	19.95	34.5%

Table 3. The memory access speedup (with errors within ± 0.05) and overall reductions in run time in the extension process over the basic algorithm observed in experiments with a few large systems in standard test suites on a NUMA system consisting of 8 node each having 8 quad-core AMD Opteron processor with a total of 256 cores. Experiments marked by '*' only used 4 out of 8 nodes (128 of the 256 cores) due to smaller size. The time spent on the computation of cell volume (6.13) and the accumulation of the mixed volume are not included since they are not affect by the memory access pattern.

15.2. Extending to computer clusters

While the above referenced NUMA architecture allows shared-memory systems to scale to tens or even more than 100 processor cores, their scalability is still limited by the inherently high cost. Larger systems that contain several hundreds or even thousands of cores generally take the form of distributed-memory systems in which nodes, connected by some network, do not directly share memory spaces but communicate with one another by passing messages instead. The parallel algorithm described above can be extended to distributed-memory systems including *computer clusters* in which nodes are connected by dedicated high speed network.

In such distributed-memory systems, a *master-worker* model is chosen to extend the parallel algorithm described above. In this model, the "master" runs on a single node in the system. It first populates its own task pool with an initial set of subfaces. The number of initial subfaces is determined based on the number of nodes available within the system (a prescribed multiple of the number of nodes). This initial task pool is then divided into equal

portions and sent to each of the remaining nodes as seeds for exploration. Each worker executes the EXPLORE algorithm described in §8 and explores the subgraph accessible from the initial set of nodes sent by the master until its task pool becomes empty. At the end each worker would have a collection of mutually exclusive mixed cells. These are then passed back to the master to form a final set of mixed cells. This basic scheme was proposed in [66] and significantly improved in [16]. Figure 30 shows a typical setup in which arrows indicate the passing of tasks between nodes.

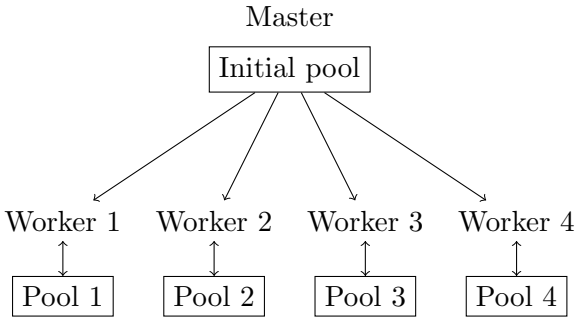


Figure 30. A master-worker setup for performing parallel mixed cell enumeration on a computer cluster with 5 nodes. One node act as the master and the remaining nodes act as workers each having their own task pools. The initial tasks are passed to each worker for exploration.

The Message Passing Interface, or MPI, is a specification that allows nodes to communicate with one another in a cluster. Though not sanctioned by any major standards body, MPI has became a *de facto* standard for scientific computation on computer clusters. In Hom4PS-3, this protocol is used for the communication between the master and workers.

An implementation based on this master-worker model would not be scalable without *load balancing* mechanisms. In exploring the spanning tree of the feasible subgraph, certain branches may require significantly more CPU time than other branches. Such imbalances generally cannot be detected easily ahead of time, a dynamic load balancing mechanism that actively shifts tasks from one worker to another is therefore critically essential to the overall efficiency and scalability of this algorithm. In [16], this problem is resolved by requiring each worker to request more tasks from the master when it exhausted its own task pool. However, the waiting spent on message passing incurs a measurable and sometime significant cost. Indeed, in large clusters, experiments suggest that the waiting time often dominate the

overall run time as the single master node can be easily overwhelmed by the large number of worker nodes.

A major improvement Hom4PS-3 provides over [16] is the use of asynchronous message passing and buffering to further improve the load balancing mechanism: In addition to the task pool, each worker also maintains an "overflow buffer" which is filled with newly discovered tasks whenever the number of tasks in the task pool exceeds a prescribed threshold. The number of tasks in the buffer is periodically reported back to the master which maintains a dynamic tally of the imbalance of buffers among the workers. The master periodically broadcasts to all workers which buffer has the lowest number of tasks. Upon receiving the notice, each worker whose buffer has higher number of tasks then passes certain number of tasks to the buffers with lowest number of tasks. Here the passing of tasks from one buffer to another is performed via efficient *asynchronous message passing* provided by recent revisions of the MPI standard that are asynchronous in the sense that they do not interrupt the threads from their main computational intensive task. That is, the transmission of data takes place in the background and the workers have no need to wait for the sending or receiving. See Figure 31 for the illustration. When a worker exhausted its own task pool, it moves tasks from its own buffer, which resides in the worker's local memory space, to its task pool so that the task pool reaches the original prescribed capacity. The worker then continues its exploration.

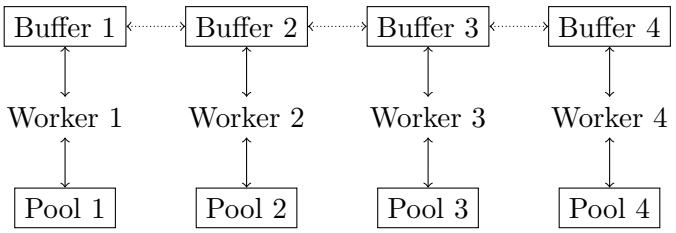


Figure 31. Buffering and load balancing mechanisms among workers. Tasks are moved from one buffer to another in the background. In contrast to the simple master-worker setup illustrated in Figure 30, this model relies mostly on direct (peer-to-peer) communication among worker nodes.

The asynchronous load balancing substantially improved the efficiency and scalability of the basic scheme developed in [16]. The implementation exhibits great scalability on clusters having between 32 and 200 nodes. It is expected that the speedup ratio cannot get close to those achieved on a multi-core system (as shown in Figure 27) due to the inherently higher cost

in communication. However it is possible to scale to many more processors cores than on multi-core or NUMA system. For example, the speedup ratios achieved using multiple nodes in a cluster for the fivebody (five body central configuration) problem [3, 37, 54] is shown in Figure 32.

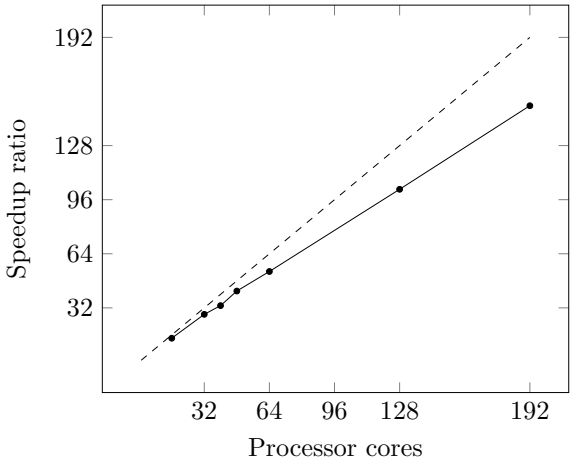


Figure 32. Speedup ratios achieved by the distributed-memory variation of the parallel algorithm for mixed cell enumeration over the fastest serial implementations MixedVol-2.0 [52] and DEMiCs [77]. Measurements are done in a cluster containing up to 192 processor cores.

15.3. GPU accelerated mixed cell enumeration algorithm

An exciting development in the world of computing is the advent of general purpose parallel computation on GPUs. Highly parallel by design, GPUs are more efficient than traditional CPUs in performing a variety of complicated tasks [85]. In terms of raw computational power, GPU devices have now surpassed the fastest CPU available [85]. However, hurdles still exist along the path to fully employing GPU computing. In particular, both the memory layout and thread organization are very different from their counterparts in traditional CPUs. In NVidia's CUDA architecture, for example, threads are always organized in groups of 32 threads called "warps" [85] which are the basic scheduling units in CUDA GPUs with all threads in a warp always perform the same instruction at the same time.

In an attempt to take advantage of GPU devices, Hom4PS-3 adopts the approach of GPU *accelerated* mixed cell enumeration algorithm where the

CPUs still perform the main algorithm, and GPU devices provide assistance in computation intensive tasks that are best suited for GPUs. Though still an experimental part of the software with active development currently underway, the remarkable speedups achieved definitely merit further investigation on the idea. We therefore present the preliminary results here.

The GPU accelerated mixed cell enumeration algorithm explores the parallelism inside individual "tasks" which is not directly utilized in the approaches discussed above. It comes in the form of a separated module that performs a specific set of operations inside the one-point test problems (8.4). As we mentioned in §8, in the mixed cell enumeration algorithm included in Hom4PS-3, the simplex method is preferred due to the great amount of additional information it generates which can be used to substantially accelerate the mixed cell enumeration process (see §8.3.3).

The simplex method is an iterative method for solving the linear programming problems of the form (8.4). Leaving aside the technical details, the key property being exploited here is that each iteration in the simplex method involves the manipulation of a fixed size matrix. In particular, the part that dominates the overall computational cost takes the form of a matrix-vector multiplication

$$\mathbf{y} \leftarrow A \cdot \mathbf{x}$$

where A is a fixed matrix with real entries whose number of rows is much greater than the number of columns. The potential parallelism in this operation is immediate: in principle, every scalar-scalar multiplication involved can be computed independently.

A straightforward approach would be to utilize the standard NVidia cuBLAS library [85] which has built-in functions designed specifically for handling this task. Unfortunately, the cuBLAS library incurs a small but measurable amount of additional cost upon each invocation. Recall that the enumeration of mixed cells involves a large number of one-point tests, the cumulative costs associated with cuBLAS often outweighs its benefits, as our experiments suggest.

A direct programming approach is therefore in place. The computation is divided into two steps. First, all the scalar-scalar products $\{a_{i,j} \cdot x_j\}$ are computed in parallel where $a_{i,j}$ and x_j are the entries of A and $\mathbf{x}$ respectively. GPU devices are generally capable of running hundreds or even thousands of threads simultaneously. Conforming to the organization of threads on GPU, this step is done in block of 16×16 threads. That is, the (i, j) block of

System (dimension)	Avg. speedup in $\mathbf{y} \leftarrow A\mathbf{x}$
cyclic15 (15)	3.55
cyclic19 (19)	3.95
cyclic23 (23)	8.09
cyclic27 (27)	9.00
cyclic31 (31)	31.54
cyclic47 (47)	39.90

Table 4. The average speedup ratio in computing $\mathbf{y} \leftarrow A\mathbf{x}$ and solving one-point test problems respectively observed in some standard test suite problems. The data reflect the average of 10000 one-point tests for each problem.

16×16 threads computes the array of products

$$\begin{bmatrix} a_{16i,16j} \cdot x_{16j} & a_{16i,16j+1} \cdot x_{16j+1} & \cdots & a_{16i,16j+15} \cdot x_{16j+15} \\ a_{16i+1,16j} \cdot x_{16j} & a_{16i+1,16j+1} \cdot x_{16j+1} & \cdots & a_{16i+1,16j+15} \cdot x_{16j+15} \\ \vdots & \vdots & & \vdots \\ a_{16i+15,16j} \cdot x_{16j} & a_{16i+15,16j+1} \cdot x_{16j+1} & \cdots & a_{16i+15,16j+15} \cdot x_{16j+15} \end{bmatrix}$$

with one thread computing each entry. Once the scalar-scalar products $\{a_{i,j} \cdot x_j\}$ are all computed, the standard "parallel reduction" algorithm is then applied to compute the row sums among blocks of the above form. Table 4 shows the speedup results of this algorithm observed on some standard test suite problems using a NVidia GTX 970 graphic card. Measuring this operation of computing $\mathbf{y} \leftarrow A\mathbf{x}$ alone, approximately 3.5x to 40x speedup ratio have been achieved on sufficiently large systems.

Though still limited in its functionality and portability, on sufficiently large systems the GPU accelerated part shows remarkably promising results. Developments in applying GPU to more operations in the mixed cell enumeration algorithm are currently underway.

References

[1] R. H. Abraham and J. W. Robbin, *Transversal mappings and flows.* Benjamin, 1967.

[2] S. Ackermann and W. Kliesch, *Computation of stationary points via a homotopy method.* Theoretical Chemistry Accounts, **99**(4):255–264, July 1998.

[3] A. Albouy and V. Kaloshin, *Finiteness of central configurations of five bodies in the plane*. Ann. Math, **176**(1):535–588, 2012.

[4] E. Allgower and K. Georg, *Introduction to numerical continuation methods*, volume 45. Society for Industrial and Applied Mathematics, 2003.

[5] E. L. Allgower, *Bifurcations Arising in the Calculation of Critical Points via Homotopy Methods*. In: Numerical Methods for Bifurcation Problems (T. Küpper, H. D. Mittelmann, and H. Weber, eds.), International Series of Numerical Mathematics, no. 70, pp. 15–28, Birkhäuser Basel, Jan. 1984.

[6] E. L. Allgower and K. Georg, *Homotopy methods for approximating several solutions to nonlinear systems of equations*. 1979.

[7] G. Attardi and C. Traverso, *The PoSSo library for polynomial system solving*. Proc. of AIHENP95, 1995.

[8] D. J. Bates, J. D. Hauenstein, A. J. Sommese and C. W. Wampler, *Numerically Solving Polynomial Systems with Bertini*. Society for Industrial and Applied Mathematics, 2013.

[9] D. N. Bernshtein, *The number of roots of a system of equations*. Functional Analysis and its Applications, **9**(3):183–185, 1975.

[10] M. Best and R. Klaus, *Linear programming: Active set analysis and computer programs*. Prentice-Hall, Inc., 1985.

[11] U. Betke, *Mixed volumes of polytopes*. Archiv der Mathematik, **58**(4): 388–391, Apr. 1992.

[12] G. Björck and R. Fröberg, *A faster way to count the solutions of inhomogeneous systems of algebraic equations, with applications to cyclic n-roots*. Journal of Symbolic Computation, **12**(3):329–336, 1991.

[13] L. Blum, F. Cucker, M. Shub and S. Smale, *Complexity and real computation*. Springer-Verlag, 1998.

[14] J. Canny and J. M. Rojas, *An optimal condition for determining the exact number of roots of a polynomial system*. In: Proceedings of the 1991 International Symposium on Symbolic and Algebraic Computation, ISSAC '91, pp. 96–102, ACM, New York, NY, USA, 1991.

[15] T. Chen, T.-L. Lee and T.-Y. Li, *Hom4PS-3: A Parallel Numerical Solver for Systems of Polynomial Equations Based on Polyhedral*

Homotopy Continuation Methods. In: Mathematical Software — ICMS 2014 (H. Hong and C. Yap, eds.), Lecture Notes in Computer Science, no. 8592, pp. 183–190, Springer Berlin Heidelberg, Jan. 2014.

[16] T. Chen, T.-L. Lee and T.-Y. Li, *Mixed volume computation in parallel*. Taiwanese Journal of Mathematics, **18**(1):93–114, 2014.

[17] T. Chen and T.-Y. Li, *Solutions to systems of binomial equations*. Annales Mathematicae Silesianae, **28**:7–34, 2014.

[18] T. Chen, T.-Y. Li and X. Wang, *Theoretical aspects of mixed volume computation via mixed subdivision*. Communications in Information and Systems, **14**(4):213–242, 2014.

[19] T. Chen and D. Mehta, *An index-resolved fixed-point homotopy and potential energy landscapes*. `arXiv:1504.06622 [cond-mat]`, Apr. 2015.

[20] T. Chen and D. Mehta, *Parallel degree computation for binomial systems*. Numerical Algebraic Geometry: Special Issue of Journal of Symbolic Computation, (to appear).

[21] S.-N. Chow, J. Mallet-Paret and J. A. Yorke, *A homotopy method for locating all zeros of a system of polynomials*. In: Functional Differential Equations and Approximation of Fixed Points (H.-O. Peitgen and H.-O. Walther, eds.), Lecture Notes in Mathematics, no. 730, pp. 77–88, Springer Berlin Heidelberg, Jan. 1979.

[22] D. A. Cox, J. B. Little and D. O'shea, *Using algebraic geometry*. 2005.

[23] D. A. Cox, J. B. Little and H. K. Schenck, *Toric varieties*. American Mathematical Soc., 2011.

[24] D. F. Davidenko, *On a new method of numerical solution of systems of nonlinear equations*. In: Dokl. Akad. Nauk SSSR, volume 88, pp. 601–602, 1953.

[25] F.-J. Drexler, *Eine methode zur Berechnung sämtlicher Lösungen von Polynomgleichungssystemen*. Numerische Mathematik, **29**(1):45–58, Mar. 1977.

[26] D. Eisenbud and B. Sturmfels, *Binomial ideals*. Duke Mathematical Journal, **84**:1–46, July 1996.

[27] W. Fulton, *Introduction to toric varieties*, no. 131. Princeton University Press, 1993.

[28] W. Fulton, *Intersection Theory*. Springer New York, Jan. 1998.

[29] T. Gao and T.-Y. Li, *Mixed volume computation via linear programming*. Taiwanese Journal of Mathematics, **4**(4):599–619, Jan. 2000.

[30] T. Gao and T.-Y. Li, *Mixed volume computation for semi-mixed systems*. Discrete & Computational Geometry, **29**(2):257–277, Jan. 2003.

[31] T. Gao, T.-Y. Li and X. Wang, *Finding all isolated zeros of polynomial systems in Cn via stable mixed volumes*. Journal of Symbolic Computation, **28**(1–2):187–212, July 1999.

[32] C. B. Garcia and W. I. Zangwill, *An Approach to Homotopy and Degree Theory*. Mathematics of Operations Research, **4**(4):390–405, Nov. 1979.

[33] C. B. Garcia and W. I. Zangwill, *Finding all solutions to polynomial systems and other systems of equations*. Mathematical Programming, **16**(1):159–176, Dec. 1979.

[34] M. S. Gockenbach, *Finite-dimensional linear algebra*. CRC Press, June 2011.

[35] A. Griewank and M. Osborne, *Analysis of Newton's Method at Irregular Singularities*. SIAM Journal on Numerical Analysis, **20**(4):747–773, Aug. 1983.

[36] R. Gunning and H. Rossi, *Analytic functions of several complex variables*. AMS Chelsea Publishing, 2009.

[37] M. Hampton and A. Jensen, *Finiteness of spatial central configurations in the five-body problem*. Celestial Mechanics and Dynamical Astronomy, **109**(4):321–332, 2011.

[38] M. Hampton and R. Moeckel, *Finiteness of stationary configurations of the four-vortex problem*. Transactions of the American Mathematical Society, **361**(3):1317–1332, 2009.

[39] R. Hartshorne, *Algebraic geometry*, no. 52. Springer, 1977.

[40] J. Hauenstein, A. Sommese and C. Wampler, *Regeneration homotopies for solving systems of polynomials*. Mathematics of Computation, **80**(273):345–377, 2011.

[41] M. Herlihy and N. Shavit, *The art of multiprocessor programming*. Elsevier, June 2012.

[42] B. Huber and B. Sturmfels, *A polyhedral method for solving sparse polynomial systems*. Math. of computation, **64**(212):1541–1555, 1995.

[43] B. Huber and B. Sturmfels, *Bernstein's theorem in affine space*. Discrete & Computational Geometry, **17**(2):137–141, Mar. 1997.

[44] B. Huber and J. Verschelde, *Polyhedral end games for polynomial continuation*. Numerical Algorithms, **18**(1):91–108, 1998.

[45] T. Kahle, *Decompositions of binomial ideals*. Annals of the Institute of Statistical Mathematics, **62**(4):727–745, 2010.

[46] T. Kahle and E. Miller, *Decompositions of commutative monoid congruences and binomial ideals*. Algebra & Number Theory, **8**(6):1297–1364, Oct. 2014.

[47] L. Kantorovich, *On Newton's method for functional equations*. In: Dokl. Akad. Nauk SSSR, volume 59, pp. 1237–1240, 1948.

[48] A. Khovanskii, *Newton polyhedra and the genus of complete intersections*. Functional Analysis and Its Applications, **12**(1):38–46, 1978.

[49] Y. C. Kuo and T.-Y. Li, *Determining dimension of the solution component that contains a computed zero of a polynomial system*. Journal of Mathematical Analysis and Applications, **338**(2):840–851, Feb. 2008.

[50] A. G. Kushnirenko, *Newton polytopes and the Bezout theorem*. Functional Analysis and Its Applications, **10**(3):233–235, July 1976.

[51] C. Lee, *Regular triangulations of convex polytopes*. Applied Geometry and Discrete Mathematics: The Victor Klee Festschrift, **4**, 1991.

[52] T.-L. Lee and T.-Y. Li, *Mixed volume computation in solving polynomial systems*. Contemp. Math., **556**:97–112, 2011.

[53] T.-L. Lee, T.-Y. Li and C.-H. Tsai, *HOM4PS-2.0: a software package for solving polynomial systems by the polyhedral homotopy continuation method*. Computing, **83**(2):109–133, 2008.

[54] T.-L. Lee and M. Santoprete, *Central configurations of the five-body problem with equal masses*. Celestial Mechanics and Dynamical Astronomy, **104**(4):369–381, 2009.

[55] A. Leykin, J. Verschelde and A. Zhao, *Newton's method with deflation for isolated singularities of polynomial systems*. Theoretical Computer Science, **359**(1–3):111–122, Aug. 2006.

[56] T. Li and Z. Zeng, *A Rank-Revealing Method with Updating, Downdating, and Applications*. SIAM Journal on Matrix Analysis and Applications, **26**(4):918–946, Jan. 2005.

[57] T.-Y. Li, *On Chow, Mallet-Paret and Yorke homotopy for solving systems of polynomials*. Bulletin of the Institute of Mathematics. Acad. Sinica, **11**(3):433–437, 1983.

[58] T.-Y. Li, *Numerical solution of polynomial systems by homotopy continuation methods*. In: Handbook of Numerical Analysis (P. G. Ciarlet, ed.), volume 11, pp. 209–304, North-Holland, 2003.

[59] T.-Y. Li and X. Li, *Finding mixed cells in the mixed volume computation*. Foundations of Computational Mathematics, **1**(2):161–181, Jan. 2001.

[60] T.-Y. Li and T. Sauer, *A simple homotopy for solving deficient polynomial systems*. Japan Journal of Applied Mathematics, **6**(3):409–419, Oct. 1989.

[61] T.-Y. Li, T. Sauer and J. A. Yorke, *Numerical solution of a class of deficient polynomial systems*. SIAM Journal on Numerical Analysis, **24**(2):435–451, Apr. 1987.

[62] T.-Y. Li, T. Sauer and J. A. Yorke, *The random product homotopy and deficient polynomial systems*. Numerische Mathematik, **51**(5):481–500, 1987.

[63] T.-Y. Li, T. Sauer and J. A. Yorke, *Numerically determining solutions of systems of polynomial equations*. Bulletin of the American Mathematical Society, **18**(2):173–177, 1988.

[64] T.-Y. Li, T. Sauer and J. A. Yorke, *The cheater's homotopy: an efficient procedure for solving systems of polynomial equations*. SIAM Journal on Numerical Analysis, pp. 1241–1251, 1989.

[65] T.-Y. Li, T. Wang and X. Wang, *Random product homotopy with minimal BKK bound*. The Mathematics of Numerical Analysis, **32**, 1996.

[66] T.-Y. Li and C.-H. Tsai, *HOM4PS-2.0para: Parallelization of HOM4PS-2.0 for solving polynomial systems*. Parallel Computing, **35**(4):226–238, 2009.

[67] T.-Y. Li and X. Wang, *A homotopy for solving the kinematics of the most general six-and-five-degree of freedom manipulators*. Proc. of ASME Conference on Mechanisms, **25** (1990), pp. 249–252.

[68] T.-Y. Li and X. Wang, *Solving deficient polynomial systems with homotopies which keep the subschemes at infinity invariant*. Mathematics of Computation, **56**(194):693–710, 1991.

[69] T.-Y. Li and X. Wang, *Nonlinear Homotopies for Solving Deficient Polynomial Systems with Parameters*. SIAM Journal on Numerical Analysis, **29**(4):1104–1118, Aug. 1992.

[70] T.-Y. Li and X. Wang, *The BKK root count in Cn*. Mathematics of Computation of the American Mathematical Society, **65**(216):1477–1484, 1996.

[71] G. Malajovich, *Computing mixed volume and all mixed cells in quermassintegral time*. `arXiv:1412.0480 [math]`, Dec. 2014.

[72] D. Mehta, T. Chen, J. D. Hauenstein and D. J. Wales, *Newton homotopies for sampling stationary points of potential energy landscapes*. The Journal of Chemical Physics, **141**(12):121104, Sept. 2014.

[73] D. Mehta, T. Chen, J. W. R. Morgan and D. J. Wales, *Exploring the potential energy landscape of the Thomson problem via Newton homotopies*. The Journal of Chemical Physics, **142**(19):194113, May 2015.

[74] D. Mehta, J. D. Hauenstein and M. Kastner, *Energy-landscape analysis of the two-dimensional nearest-neighbor ϕ^4 model*. Physical Review E, **85**(6):061103, June 2012.

[75] E. Miller and B. Sturmfels, *Combinatorial commutative algebra*. volume 227, Springer, 2005.

[76] H. Minkowski, *Theorie der Konvexen Körper, insbesondere Begrundung ihres Oberflachenbegriffs*. Gesammelte Abhandlungen von Hermann Minkowski, **2**:131–229, 1911.

[77] T. Mizutani and A. Takeda, *DEMiCs: A Software Package for Computing the Mixed Volume Via Dynamic Enumeration of all Mixed Cells*. In: Software for Algebraic Geometry (M. Stillman, J. Verschelde and N. Takayama eds.), The IMA Volumes in Mathematics and its Applications, no. 148, pp. 59–79, Springer, Jan. 2008.

[78] T. Mizutani, A. Takeda and M. Kojima, *Dynamic enumeration of all mixed cells*. Discrete & Computational Geometry, **37**(3):351–367, Mar. 2007.

[79] A. P. Morgan, *A homotopy for solving polynomial systems*. Applied Mathematics and Computation, **18**(1):87–92, Jan. 1989.

[80] A. P. Morgan, *Solving polynomial systems using continuation for engineering and scientific problems*. Classics in Applied Mathematics, volume 57, Society for Industrial and Applied Mathematics, 2009.

[81] A. P. Morgan and A. J. Sommese, *A homotopy for solving general polynomial systems that respects m-homogeneous structures*. Applied Mathematics and Computation, **24**(2):101–113, 1987.

[82] A. P. Morgan and A. J. Sommese, *Coefficient-parameter polynomial continuation*. Applied Math. & Computation, **29**(2):123–160, 1989.

[83] A. P. Morgan, A. J. Sommese and C. W. Wampler, *A power series method for computing singular solutions to nonlinear analytic systems*. Numerische Mathematik, **63**(1):391–409, Dec. 1992.

[84] A. P. Morgan, A. J. Sommese and C. W. Wampler, *Computing singular solutions to polynomial systems*. Advances in Applied Mathematics, **13**(3):305–327, Sept. 1992.

[85] NVIDIA Corporation, *NVIDIA CUDA C Programming Guide*. Tech. rep., July 2011.

[86] T. Ojika, *Modified deflation algorithm for the solution of singular problems. I. A system of nonlinear algebraic equations*. Journal of Mathematical Analysis and Applications, **123**(1):199–221, Apr. 1987.

[87] T. Ojika, S. Watanabe and T. Mitsui, *Deflation algorithm for the multiple roots of a system of nonlinear equations*. Journal of mathematical analysis and applications, **96**(2):463–479, 1983.

[88] J. M. Ortega, *The Newton-Kantorovich Theorem.*The American Mathematical Monthly, **75**(6):658, June 1968.

[89] J. M. Ortega and W. C. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables*. SIAM, Jan. 2000.

[90] P. Percell, *Note on a global homotopy*. Numerical Functional Analysis and Optimization, **2**(1):99–106, Jan. 1980.

[91] J. M. Rojas, *A convex geometric approach to counting the roots of a polynomial system*. Theoretical Computer Science, **133**(1):105–140, Oct. 1994.

[92] J. M. Rojas, *Toric intersection theory for affine root counting*. Journal of Pure and Applied Algebra, **136**(1):67–100, Mar. 1999.

[93] S. M. Selby ed., *CRC Standard Mathematical Tables*. The Chemical Rubber Company, Cleveland, Ohio, 1971.

[94] A. Sard, *The measure of the critical values of differentiable maps*. Bulletin of the American Mathematical Society, **48**(12):883–890, 1942.

[95] I. R. Shafarevich, *Basic Algebraic Geometry 1*. Springer Berlin Heidelberg, Jan. 2013.

[96] M. Shub and S. Smale, *Complexity of Bézout's Theorem I: Geometric aspects*. Journal of the AMS, **6**(2):459–501, 1993.

[97] S. S. Skiena, *The algorithm design manual*. Springer Science & Business Media, Apr. 2009.

[98] S. Smale, *A convergent process of price adjustment and global newton methods*. Journal of Mathematical Economics, **3**(2):107–120, July 1979.

[99] A. J. Sommese, J. Verschelde and C. W. Wampler, *Using monodromy to decompose solution sets of polynomial systems into irreducible components*. In: Applications of Algebraic Geometry to Coding Theory, Physics and Computation (C. Ciliberto, F. Hirzebruch, R. Miranda and M. Teicher eds.), NATO Science Series, no. 36, pp. 297–315, Springer Netherlands, 2001.

[100] A. J. Sommese, J. Verschelde and C. W. Wampler, *Symmetric functions applied to decomposing solution sets of polynomial systems*. SIAM Journal on Numerical Analysis, **40**(6):2026–2046, 2002.

[101] A. J. Sommese, J. Verschelde and C. W. Wampler, *Solving polynomial systems equation by equation*. In: Algorithms in algebraic geometry, The IMA Volumes in Mathematics and its Applications, no. 146, pp. 133–152, Springer, 2008.

[102] A. J. Sommese and C. W. Wampler, *Numerical algebraic geometry*. In: The Mathematics of Numerical Analysis, Lectures in Applied Mathematics, volume 32, pp. 749–763, AMS, 1996.

[103] A. J. Sommese and C. W. Wampler, *The numerical solution of systems of polynomials arising in engineering and science*. World Scientific Pub Co Inc, 2005.

[104] J. Stoer and R. Bulirsch, *Introduction to numerical analysis*. volume 12, Springer-Verlag, 2002.

[105] B. Sturmfels, *Equations defining toric varieties*. In PROC. SYMPOSIA IN PURE, Citeseer, 1997.

[106] A. Takeda, M. Kojima and K. Fujisawa, *Enumeration of all solutions of a combinatorial linear inequality system arising from the polyhedral homotopy continuation method*. Journal of the Operations Research Society of Japan-Keiei Kagaku, **45**(1):64–82, 2002.

[107] Traverso, C, *The PoSSo test suite examples*. 1993.

[108] L.-W. Tsai and A. P. Morgan, *Solving the kinematics of the most general six- and five-degree-of-freedom manipulators by continuation methods*. Journal of Mechanical Design, **107**(2):189–200, June 1985.

[109] J. Verschelde, K. Gatermann and R. Cools, *Mixed-volume computation by dynamic lifting applied to polynomial system solving*. Discrete & Computational Geometry, **16**(1):69–112, Jan. 1996.

[110] J. Verschelde and A. Haegemans, *The GBQ-Algorithm for Constructing Start Systems of Homotopies for Polynomial Systems*. SIAM Journal on Numerical Analysis, **30**(2):583–594, Apr. 1993.

[111] H. von Helmholtz, *Uber integrale der hydrodynamischen gleichungen welche den wirbelbewegungen entsprechen*. Crelle's Journal fur Mathematic, **55**:25–55, 1858.

[112] C. W. Wampler, *Bezout number calculations for multi-homogeneous polynomial systems*. Applied Mathematics and Computation, **51**(2–3): 143–157, Oct. 1992.

[113] C. W. Wampler, *An efficient start system for multi-homogeneous polynomial continuation*. Numerische Mathematik, **66**(1):517–523, Dec. 1993.

[114] C. W. Wampler, A. P. Morgan and A. J. Sommese, *Complete solution of the nine-point path synthesis problem for four-bar linkages*. Journal of Mechanical Design, **114**(1):153–159, 1992.

[115] A. H. Wright, *Finding all solutions to a system of polynomial equations*. Mathematics of Computation, **44**(169):125–133, 1985.

[116] S. T. Yau, personal communication.

[117] W. Zulehner, *A simple homotopy method for determining all isolated solutions to polynomial systems*. Mathematics of Computation, **50** (181):167–177, 1988.

Department of Mathematics, Michigan State University
619 Red Cedar Road, East Lansing, MI 48824, USA
E-mail address: chentia1@msu.edu

Department of Mathematics, Michigan State University
619 Red Cedar Road, East Lansing, MI 48824, USA
E-mail address: li@math.msu.edu

Received September 6, 2015
Accepted October 2, 2015