Numerical Algorithms
https://doi.org/10.1007/s11075-023-01590-1

ORIGINAL PAPER

u-generation: solving systems of polynomials equation-by-equation

Timothy Duff[1] · **Anton Leykin**[2] · **Jose Israel Rodriguez**[3]

Received: 7 December 2022 / Accepted: 27 May 2023

Abstract
We develop a new method that improves the efficiency of equation-by-equation homotopy continuation methods for solving polynomial systems. Our method is based on a novel geometric construction and reduces the total number of homotopy paths that must be numerically continued. These improvements may be applied to the basic algorithms of numerical algebraic geometry in the settings of both projective and multiprojective varieties. Our computational experiments demonstrate significant savings obtained on several benchmark systems. We also present an extended case study on maximum likelihood estimation for rank-constrained symmetric $n \times n$ matrices, in which multiprojective u-generation allows us to complete the list of ML degrees for $n \leq 6$.

Keywords Solving polynomials · Numerical algebraic geometry · Regeneration

All authors thank the Institute for Mathematics and its Applications in Minneapolis for hosting them for a series of "SageMath and Macaulay2" meetings in 2019-2020 where the ideas of this work were born.

Timothy Duff: https://timduff35.github.io/timduff35/
Anton Leykin: https://antonleykin.math.gatech.edu/index.html
Jose Israel Rodriguez: https://sites.google.com/wisc.edu/jose/.

✉ Timothy Duff
timduff@uw.edu

Anton Leykin
leykin@math.gatech.edu

Jose Israel Rodriguez
jose@math.wisc.edu

1 Department of Mathematics, University of Washington, Box 354350, Seattle, WA 98195–4350, USA

2 School of Mathematics, Georgia Institute of Technology, 686 Cherry Street NW, Atlanta, GA 30308, USA

3 Department of Mathematics, University of Wisconsin-Madison, 480 Lincoln Drive, Madison, WI 30308, USA

Published online: 29 June 2023

Mathematics Subject Classification (2010) 65H20 · 14Q65 · 68W30

1 Introduction

1.1 History of equation-by-equation solvers

Variants of the equation-by-equation approach to polynomial system solving have been developed in the context of both symbolic computation (see, e.g., [10]) and numerical homotopy methods. It is far beyond the scope of this paper to evaluate relative strengths and weaknesses when comparing these two broad categories. Our sole aim is to improve one of the existing *numerical* methods, and we briefly review the most relevant prior work.

The approach of [28] modifies polynomials in the system by adding linear terms in a set of new *slack variables*, producing *embedded systems*. The resulting *cascade of homotopies* may be considered the first practical equation-by-equation solver in the framework of numerical algebraic geometry.

The algorithm of [30] also introduces new variables at each step of a different equation-by-equation cascade based on *diagonal homotopies*. The number of additional variables in both this method and the method of [28] equals at least the dimension of the solution set.

The ingenuity of *regeneration* [16–18] stems from a realization that at each step of the cascade, when considering a new polynomial of degree d, one can "replace" it with a product of random d linear forms. This results in a **two-stage** procedure that, first, precomputes d copies of witness sets corresponding to the linear factors and, second, deforms the union of d hyperplanes into a hypersurface given by the original polynomial.

The homotopy realizing this deformation describes a family of 0-dimensional varieties in the projective space $\mathbb{P}^n$ — no new variables are introduced. For an overview of regeneration, as implemented in the software package Bertini [2], we refer to [3, §5.4, 9.4], which also discusses computational examples highlighting its relative strengths. An extension of this method to *multiprojective* varieties in a product of projective spaces $\mathbb{P}^{n_1} \times \cdots \times \mathbb{P}^{n_k}$ was given in [14].

Remark 1.1 We assume familiarity with basic notions from algebraic geometry. For readers who are less comfortable with these notions, we suggest looking at Example 2.2 to understand our proposed method in its most basic form and for an explicit example where the homotopy is written out completely in coordinates. The discussion at the beginning of Section 3, up to and including Example 3.1, should provide motivation and intuition for the more-complicated multiprojective homotopy.

For theoretical reasons, it is convenient to work with (multi)projective varieties defined by (multi)homogeneous equations. For affine varieties defined by general polynomial systems, various strategies enable a reduction to the homogeneous case. For example, when computing witness sets as in Algorithm 3, points representing these extraneous components at infinity may be discarded at any stage if only finite solutions are of interest.

1.2 Contributions and outline

We develop a new **one-stage** procedure dubbed *u-generation* for a step in the equation-by-equation cascade. The core is a geometric construction that relies on a homotopy in $\mathbb{P}^{n+1}$, thus introducing one new variable u. In fact, this new variable can be eliminated when implementing the method (see Remark 2.6). With this modification, u-generation performs similarly to the second stage of regeneration, thus saving the cost of performing the first stage.

The rest of the paper is outlined as follows. Section 2 provides necessary background on witness sets, introduces u-generation, and — for completeness — outlines a simple algorithm for an equation-by-equation cascade. In Section 3, we extend our approach, albeit in a nontrivial way, to multihomogeneous systems: homotopies are transplanted from $\mathbb{P}^{n_1} \times \cdots \times \mathbb{P}^{n_k}$ to $\mathbb{P}^{n_1+1} \times \cdots \times \mathbb{P}^{n_k+1}$. Section 4 describes the results of several computational experiments, comparing u-generation to regeneration. In Section 4.1, we apply the methods of Section 2 to several benchmark problems, demonstrating potential savings brought by u-generation. In Section 4.2, we demonstrate how u-generation in the multiprojective setting may be applied to solve nontrivial problems in maximum-likelihood estimation. Section 5 provides a short conclusion.

Remark 1.2 One seeming complication for u-generation in a multiprojective case is that, unlike for multiprojective regeneration in [14], the continuation paths may emanate from a singular point. However, it turns out that, should there be a singularity at the value of continuation parameter $t = 0$, one can analyze the behavior of the path in its neighborhood and ensure a robust numerical tracking from a regular starting point at $t = \varepsilon$ as explained in Section 3.2.

2 *u*-generation in $\mathbb{P}^n$

2.1 The homotopy

Let X be a closed subvariety of complex projective space $\mathbb{P}^n$. A straight-line homotopy on X has the form

$$H_t = (1-t)G_0 + tG_1, \ t \in [0,1], \tag{1}$$

where G_0 and G_1 consist of r-many homogeneous polynomials of matching degrees in indeterminates $[x_0 : x_1 : \cdots : x_n]$. We abbreviate the straight-line homotopy (1) by $G_0 \rightsquigarrow G_1$. Given that $\mathbb{V}(H_t) \cap X$ is a finite set, for $t \in [0,1]$, one may consider homotopy paths emanating from *start points* $\mathbb{V}(G_0) \cap X$. A typical application of numerical homotopy continuation is to track these paths in an attempt to compute the *endpoints* $\mathbb{V}(G_1) \cap X$.[1]

For traditional equation-by-equation homotopies, the systems G_0 and G_1 appearing in (1) differ by exactly one equation, and the remaining equations define a curve in some neighborhood of any given start point. Hence, we may assume that X is a curve

[1] In this paper, we shall gloss over many details of how homotopy tracking is accomplished in practice and issues arising from the need to use approximations of points. For this, we refer the reader to introductory chapters of [31].

which is a component or a union of one-dimensional components of $\mathbb{V}(F) \subset \mathbb{P}^n$. Consider the cone $\tilde{X} \subset \mathbb{P}^{n+1}$ with coordinates $[u : \boldsymbol{x}] = [u : x_0 : x_1 : \cdots : x_n]$. In the chart $u = 1$, this is the affine cone over X, and

$$\tilde{X} = \overline{\{[u : \boldsymbol{x}] \mid \boldsymbol{x} \in X\}} = \{[u : \boldsymbol{x}] \mid \boldsymbol{x} \in X\} \cup \{[1 : 0 : \cdots : 0]\}. \tag{2}$$

Given $g_1 \in \mathbb{C}[\boldsymbol{x}]_d$ (homogeneous of degree d), we consider a homotopy

$$\tilde{H}_t : (g_0, \ell_0) \rightsquigarrow (g_1, u) \quad \text{on } \tilde{X}, \tag{3}$$

where $\ell_0 \in \mathbb{C}[\boldsymbol{x}]_1$ and $g_0 \in \mathbb{C}[u, \boldsymbol{x}]_d$.

Proposition 2.1 *For generic $g_0 \in \mathbb{C}[u, \boldsymbol{x}]_d$ and $\ell_0 \in \mathbb{C}[\boldsymbol{x}]_1$, the cardinality of points on $\tilde{X}$ satisfying $\tilde{H}_t$ is equal to $d \cdot \deg X$ for $t \in [0, 1)$, where $\deg X$ denotes the degree of the projective variety X.*

The start points of the homotopy $\tilde{H}_t$ are

$$\mathbb{V}(\tilde{H}_0) \cap \tilde{X} = \{[u : \boldsymbol{x}] \mid [\boldsymbol{x}] \in X \cap \mathbb{V}(\ell_0), \text{ and } u \text{ satisfies } g_0(u, \boldsymbol{x}) = 0\}.$$

The endpoints of $\tilde{H}_t$ lie in the set

$$\{[0 : \boldsymbol{x}] \mid [\boldsymbol{x}] \in X \cap \mathbb{V}(g_1)\}.$$

In the case when this set is finite, every point is reached.

Proof Consider the exceptional set

$$\Sigma = \{(g, \ell) \in \mathbb{C}[u, \boldsymbol{x}]_d \times \mathbb{C}[u, \boldsymbol{x}]_1 \ \mid \ |\tilde{X} \cap \mathbb{V}(g, \ell)| \neq d \cdot \deg X\}$$

in the affine space of all coefficients of pairs of polynomials in $\mathbb{C}[u, \boldsymbol{x}]_d \times \mathbb{C}[u, \boldsymbol{x}]_1$. Genericity of ℓ_0 implies $X \cap \mathbb{V}(\ell_0)$ contains $\deg X$ points, and so long as the coefficients of g_0 lie outside of $(\deg X)$-many hypersurfaces in $\mathbb{C}^N$, we have $(g_0, \ell_0) \notin \Sigma$. It follows by the usual argument [31, Lemma 7.12] that the real segment $(1-t)(g_0, \ell_0) + t(g_1, u)$ for all $t \in [0, 1)$ is also disjoint from Σ. The description of the start points and where the endpoints lie is clear from the definition of $\tilde{H}_t$.

Our last claim follows from the parameter continuation theorem [31, Theorem 7.1.6]. We give an alternative elementary self-contained proof below.

Consider the incidence correspondence

$$\Gamma = \{((g, \ell), [\tilde{u} : \tilde{\boldsymbol{x}}]) \subset \mathbb{C}[u, \boldsymbol{x}]_d \times \mathbb{C}[u, \boldsymbol{x}]_1 \times \tilde{X} \mid g([\tilde{u} : \tilde{\boldsymbol{x}}]) = \ell([\tilde{u} : \tilde{\boldsymbol{x}}]) = 0\},$$

equipped with the projection $\pi : \Gamma \to \mathbb{C}[u, \boldsymbol{x}]_d \times \mathbb{C}[u, \boldsymbol{x}]_1$. The map π is a branched covering: restricted to the preimage of the complement of Σ, it is a topological covering map both in the usual (complex) and Zariski topology.

Suppose the fiber $\pi^{-1}(g_1, u)$ is finite. Consider a point

$$p = ((g_1, u), [0 : \boldsymbol{x}]) \in \pi^{-1}(g_1, u)$$

and let V be an open neighborhood of p in Γ (with the usual topology) containing no other points of $\pi^{-1}(g_1, u)$ in its closure. Notice that

$$\dim V = \dim \Gamma = \dim \left(\mathbb{C}[u, \boldsymbol{x}]_d \times \mathbb{C}[u, \boldsymbol{x}]_1\right)$$

and $U = \pi(V) \setminus \Sigma$ is a nonempty open subset of $\mathbb{C}[u, \boldsymbol{x}]_d \times \mathbb{C}[u, \boldsymbol{x}]_1$ with $\pi(p)$ in the interior of $\overline{U}$. The map π restricted to $\pi^{-1}(U) \cap V$ is a biholomorphism onto U..[2] Since $\pi(p) = (g_1, u)$ is in the interior of $\overline{U}$, the segment $(1-t)(g_0, \ell_0) + t(g_1, u)$ intersects U for values of t arbitrarily close to 1. Let $(g_t, \ell_t) \to (g_1, u)$ along the set of points where this segment intersects U. Lifting to $\pi^{-1}(U) \cap V$, we obtain $((g_t, \ell_t), [\tilde{u}_t : \tilde{\boldsymbol{x}}_t]) \to ((g_1, u), [\tilde{u} : \tilde{\boldsymbol{x}}]) \in \pi^{-1}(g_1, u)$. Since $[\tilde{u} : \tilde{\boldsymbol{x}}]$ is an endpoint of $\tilde{H}_t$, we must have $\tilde{u} = 0$. Moreover, since $\pi^{-1}(g_1, u) \cap \overline{V} = \{p\}$, we must have $[\tilde{u} : \tilde{\boldsymbol{x}}] = [0 : \boldsymbol{x}]$.

Example 2.2 Fig. 1 illustrates the homotopy $\tilde{H}_t$ (3) for a simple case: intersecting two parabolas in the plane. We intersect $X = \mathbb{V}(F) \subset \mathbb{P}^2$, where

$$F(\boldsymbol{x}) = x_1^2 - x_0 x_2 - 2x_0^2,$$

with $\mathbb{V}(g_1) \subset \mathbb{P}^2$, where

$$g_1(\boldsymbol{x}) = 2x_0^2 + x_1 x_0 - x_2^2.$$

Choosing $\ell_0(\boldsymbol{x}) = x_2$ and $g_0(u, \boldsymbol{x}) = x_0^2 - u^2$ (cf. Remark 2.5), the genericity conditions required in Proposition 2.1 are satisfied. The start points are the four points in $\mathbb{V}(F) \cap \mathbb{V}(g_0) \cap \mathbb{V}(\ell_0) = \{[\pm 1 : 1 : \pm 2 : 0]\} \subset \mathbb{P}^3$. The four endpoints of $\tilde{H}_t$ in $\mathbb{P}^3$ are given in homogeneous coordinates $[u : x_0 : x_1 : x_2]$ as below.

u	x_0	x_1	x_2
0	1	-1	-1
0	1	2	2
0	1	$-\phi$	$\phi - 1$
0	1	$\phi - 1$	$-\phi$

(Here, $\phi = (1 + \sqrt{5})/2$.) The coordinates $[x_0 : x_1 : x_2]$ in the table above give us the four points of intersection in $\mathbb{V}(F) \cap \mathbb{V}(g_1)$.

In practice, we may write down a concrete realization of the geometrically defined homotopy $\tilde{H}_t$ by working in coordinates on the affine cone over the parabola X. Specifically, we may seek implicit functions $(\mathbf{x}(t), u(t)) \in \mathbb{C}^3 \times \mathbb{C}$ of $t \in [0, 1]$ satisfying

$$(1-t)\begin{bmatrix} F(\mathbf{x}) \\ g_0(u, \mathbf{x}) \\ \ell_0(\mathbf{x}) \\ \ell_{\mathbf{aff}}(\mathbf{x}) \end{bmatrix} + t\begin{bmatrix} F(\mathbf{x}) \\ g_1(\mathbf{x}) \\ u \\ \ell_{\mathbf{aff}}(\mathbf{x}) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \tag{4}$$

where

$$\ell_{\mathbf{aff}}(\mathbf{x}) = a_0 x_0 + a_1 x_1 + a_2 x_3 + a_3 = 0, \quad a_3 \neq 0 \tag{5}$$

[2] Here, we use the fact that π is an open map, which follows from [27, 3.10].

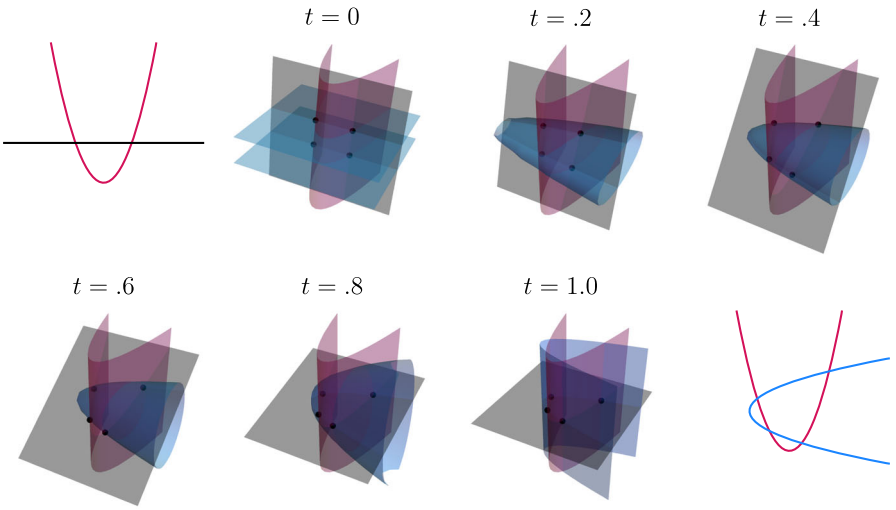

Fig. 1 Illustration of Example 2.2 in the affine chart $x_0 = 1$. The homotopy $\tilde{H}_t$ deforms a union of planes $\mathbb{V}(g_0) \subset \mathbb{P}^3$ into the cone over the target hypersurface g_1, and the plane $\mathbb{V}(\ell_1)$ into $\mathbb{V}(u)$. In the plane $\mathbb{P}^2$, this allows us to obtain $\mathbb{V}(F) \cap \mathbb{V}(g_1)$ from $\mathbb{V}(F) \cap \mathbb{V}(\ell_0)$

defines a sufficiently generic affine chart on $\mathbb{P}^2$. Intuitively, (5) removes the ambiguity of scale, since $[x_0 : x_1 : x_2] = [\lambda x_0 : \lambda x_1 : \lambda x_2]$ in $\mathbb{P}^2$ for all nonzero scalars λ. For this particular example, the choice $\ell_{\mathbf{aff}}(\mathbf{x}) = x_0 - 1$ turns out to be sufficiently generic. With this choice, there exist four solution curves $(\mathbf{x}(t), u(t))$ defined for all $t \in [0, 1]$, which are uniquely determined by the initial conditions $(u(0), \mathbf{x}(0))) = (\pm 1, (1, \pm 2, 0))$. Computing numerical approximations of these solution curves over a discretization of $[0, 1]$ now reduces to numerical integration—see [3, Ch. 1]. This is the fundamental operation performed by any numerical continuation software.

For more complicated examples, to ensure the genericity assumptions are satisfied, the linear polynomials ℓ_0, $\ell_{\mathbf{aff}}$ appearing in the homotopy (4) should be chosen generically, from which we may determine a suitable g_0 by Remark 2.5. Additional randomization may also be useful in practice, e.g., to avoid nearly-singular situations.

2.2 Witness sets

In numerical algebraic geometry, algebraic varieties are represented by witness sets. In this subsection, we review witness sets and show how they relate to the start and endpoints of our homotopy (3).

An equidimensional variety $X \subset \mathbb{P}^n$ is a finite union of irreducible varieties of the same dimension. It is represented by a *witness set* $w := (F, L, W)$, a triple consisting of

- polynomials F defining X such that it is a union of irreducible components of $\mathbb{V}(F)$;

- general [3] linear polynomials L defining a codimension $\dim(X)$ linear space, informally called a slice of $\mathbb{P}^n$;
- the set of points $W = X \cap \mathbb{V}(L)$.

Witness sets may be used to test if a point is in an irreducible component of a variety [31, Chapter 15.1] and describe a wide class of varieties including closures of images of a rational maps [14] and subvarieties of products of projective spaces [13, 15] or Grassmannians [32].

Example 2.3 Suppose $X \subset \mathbb{P}^n$ is a finite set of points. A witness set for X has the form $w = (F, \emptyset, X)$ where each point of X is an isolated point in $\mathbb{V}(F)$.

When X is a finite set of points (i.e., $\dim X = 0$), computing the intersection of X with a hypersurface is straightforward: $X \cap V(h) = \{x \in X : h(x) = 0\}$. For X of arbitrary dimension, Algorithm 2 shows how to obtain a witness set for the intersection of X with a hypersurface from a witness set for X. This can be done by reduction to the case of a curve and applying Proposition 2.1. First, we interpret the homotopy $\tilde{H}_t$ in the language of witness sets in Example 2.4.

Example 2.4 Recall from equation (3) that X is a curve, $g_0 \in \mathbb{C}[\boldsymbol{x}]_d$ is general, and the intersection $\tilde{X} \cap V(g_0)$ is a degree $\deg(X) \cdot d$ curve. Let w_0 denote a witness set for $\tilde{X} \cap V(g_0)$. Then, the witness points of w_0 are a set of start points for our homotopy (3). Similarly, the isolated endpoints of the homotopy $\tilde{H}_t$ are witness points for $\tilde{X} \cap \mathbb{V}(g_1)$, and projection from u gives witness points for $X \cap \mathbb{V}(g_1)$.

To compute w_0 and thereby the start points of $\tilde{H}_t$, assume we are given a witness set $w = (F, \ell, W)$ for X. For each point $\boldsymbol{x}^*$ in W, the set

$$\{[u : \boldsymbol{x}^*] \in \mathbb{P}^{n+1} : g_0(u, \boldsymbol{x}^*) = 0\}$$

has $\deg(g_0)$ points in $\tilde{X} \cap \mathbb{V}(g_0)$. All together, by solving $\deg(X)$ univariate degree $\deg(g_0)$ polynomials, we have

$$w_0 = (F \cup \{g_0\}),\ \ell,\ \{[u : \boldsymbol{x}^*] \in \mathbb{P}^{n+1} : g_0(u, \boldsymbol{x}^*) = 0,\ \boldsymbol{x}^* \in W\}.$$

We summarize this process in Algorithm 1.

One important fact to recall about intersecting an irreducible projective variety X with a hypersurface $\mathbb{V}(h)$ is that it leads to one of two cases: either $X \cap \mathbb{V}(h) = X$ or $X \cap \mathbb{V}(h)$ is equidimensional with dimension $\dim(X) - 1$. The former case only occurs when X is contained in the hypersurface. In the latter case, the dimension decreases by one. Moreover, the degree of the intersection is bounded above by $\deg(h) \cdot \deg(X)$.

More generally, if we drop the irreducibility hypothesis and only assume X is equidimensional, then the intersection $X \cap \mathbb{V}(h)$ consists of two equidimensional

[3] We use "general" throughout this article to mean "avoiding some proper Zariski closed set." Here, this exceptional set has a simple description: $\mathbb{V}(L)$ needs to be outside the set of subspaces in the Grassmannian $\mathrm{Gr}(n-c, n)$ that do not intersect X regularly. However, the randomized methods of homotopy continuation do not rely on knowing the exceptional set description: we use "general" without aiming to provide such a description later on (e.g., in Remark 2.5).

Algorithm 1 `uStartPoints`(w, g_0)

Input : A witness set $w = (F, \ell, W)$ for a curve $X \subset \mathbb{P}^n$;
a general homogeneous polynomial $g_0 \in \mathbb{C}[u, \boldsymbol{x}]$ of degree d.
Output : A witness set w_0 representing $\tilde{X} \cap \mathbb{V}(g_0) \subset \mathbb{P}^{n+1}$.

1 Compute

$$S_0 \leftarrow \{[u : \boldsymbol{x}] \mid \boldsymbol{x} \in W \text{ and } g_0(u, \boldsymbol{x}) = 0\},$$

which requires solving a univariate polynomial equation for each $\boldsymbol{x} \in W$.

2 Let $w_0 \leftarrow \big(F \cup \{g_0\},\ \{\ell\},\ S_0\big)$. // Here F and ℓ (which do not depend on u) should be seen as elements of $\mathbb{C}[u, \boldsymbol{x}]$.

components: the union of all irreducible components of X that are also contained $\mathbb{V}(h)$, and $\overline{(X \setminus \mathbb{V}(h)) \cap \mathbb{V}(h)}$. The first equidimensional component has the same dimension as X, and the second has dimension $\dim(X) - 1$.

As an immediate consequence of these facts, the results in the previous sections, and these examples, we have Algorithm 2.

Algorithm 2 `IntersectionWithHypersurface`(w, g_1)

Input : A homogeneous polynomial $g_1 \in \mathbb{C}[\boldsymbol{x}]$, and a witness set $w = (F, L, W)$ for a pure r-dimensional variety $X \subset \mathbb{P}^n$.
Output : Witness sets for the equidimensional components of $X \cap \mathbb{V}(g_1)$.

1 Record the witness set

$$w^{(r)} \leftarrow (F, L, W \cap V(g_1))$$

for the union of irreducible components of X contained in $V(g_1)$.

2 **if** $r \geq 1$ **then**

3 Let ℓ denote a linear polynomial in L.

4 For the curve $Z := \overline{(X \setminus \mathbb{V}(g_1))} \cap V(L \setminus \{\ell\})$, form the witness set

$$w_Z \leftarrow (F \cup L \setminus \{\ell\},\ \{\ell\},\ W \setminus \mathbb{V}(g_1)).$$

5 Choose $g_0 \in \mathbb{C}[u, \boldsymbol{x}]_{\deg g_1}$ (e.g., as in Remark 2.5) to form the homotopy

$$\tilde{H}_t : (g_0, \ell) \rightsquigarrow (g_1, u) \quad \text{on } \tilde{Z}. \tag{6}$$

6 Follow $|W \setminus \mathbb{V}(g_1)| \cdot \deg(g_1)$ homotopy paths starting (at $t = 0$) from the witness set for $\tilde{Z} \cap \mathbb{V}(g_0)$ returned by `uStartPoints`(w_Z, g_0) to obtain endpoints $E = \tilde{Z} \cap \mathbb{V}(g_1, u)$ at $t = 1$.

7 Record the witness set

$$w^{(r-1)} \leftarrow (F \cup \{g_1\},\ L \setminus \{\ell\},\ \pi_{\boldsymbol{x}}(E)),$$

where $\pi_{\boldsymbol{x}}$ is the projection that drops the u-coordinate, which represents $\overline{(X \setminus \mathbb{V}(g_1))} \cap \mathbb{V}(g_1)$.

8 **return** $\{w^{(r)}, w^{(r-1)}\}$

9 **else**

10 **return** $\{w^{(r)}\}$

Remark 2.5 When implementing Algorithm 2 and subsequent algorithms, the polynomial g_0 should be chosen in a sufficiently random fashion. At the same time, it

is desirable to have a low evaluation cost for both the values of g_0 and the roots of $g_0(u, \boldsymbol{x}) = 0$ when $\boldsymbol{x}$ is fixed.

One good candidate is $g_0 = \gamma(u^d - \ell_0^d)$ where $d = \deg(g_0)$, the linear form $\ell_0 \in \mathbb{C}[\boldsymbol{x}]_1$ does not vanish on the start points, and $\gamma \in \mathbb{C}$ is generic[4]. This is akin to the so-called γ*-trick* [31, pp. 94–95, 120–122]. Often, for practical purposes, γ is chosen randomly on the unit circle. In this case, only finitely many points on the circle are exceptional.

One other important detail for a reader who intends to implement Algorithm 2 is in the following remark.

Remark 2.6 A typical implementation of a homotopy tracking algorithm would introduce an affine chart on $\mathbb{P}^{n+1}$ by imposing an additional affine linear equation in u and $\boldsymbol{x}$. One can eliminate u in two ways: (a) using the chart equation or (b) using the equation $(1-t)\ell + tu = 0$ from homotopy (3). This results in going back to the original ambient dimension. Note that solving (b) for u is not possible when $t = 0$.

2.3 Equation-by-equation cascade

Algorithm 3 uses the methods in Section 2.1 to compute the witness set(s) for the intersection of a variety with a hypersurface.

Algorithm 3 Equation-by-equation cascade

Input: $F = (f_1, \ldots, f_c)$, $f_i \in \mathbb{C}[\boldsymbol{x}]$
Output: A collection C of witness sets describing equidimensional pieces of $\mathbb{V}(F)$.

1 Let $w = (\emptyset, \{\ell_1, \ldots, \ell_n\}, \mathbb{V}(\ell_1, \ldots, \ell_n))$, a witness set of a point, describing $\mathbb{P}^n$.
2 Initialize $C \leftarrow \{w\}$.
3 **for** *i from* 1 *to* c **do**
4 $C' \leftarrow \emptyset$
5 **for** *each* $w \in C$ **do**
6 $C' \leftarrow C' \cup$ `IntersectionWithHypersurface`(w, f_i)
7 $C \leftarrow$ `EliminateRedundantComponents`(C'), a routine that erases components contained in other components.

The correctness of the equation-by-equation cascade outlined Algorithm 3 hinges mainly on Algorithm 2. However, an implementer of the cascade must pay significant attention to details with a view to practical efficiency. For instance, "pruning" of the current collection of witness sets done via `EliminateRedundantComponents` after each step could be replaced by a potentially more efficient *en route* bookkeeping procedure; see [17].

Remark 2.7 Our description of homotopies is geometric: we consider homotopies on a variety X (or the cone $\tilde{X}$). However, the algebraic representation of X as a component

[4] The constant γ needs to avoid a *real* hypersurface (in fact, a union lines through the origin) in $\mathbb{R}^2 \cong \mathbb{C}$ containing "bad" choices: a choice of γ is "bad" if the homotopy paths of (3) cross.

of $\mathbb{V}(F)$ may not be *reduced*, i.e., F may be singular at some points of X. A common way to address this issue in an implementation is *deflation*, which is a technique of replacing F with a system of polynomials (potentially in more variables) that are regular at a generic point: see [31, §13.3.2] and [25].

3 *u*-generation in products of projective spaces

In this section, we generalize the homotopy $\tilde{H}_t$ to work in a product of projective spaces $\mathbb{P}^{n_1} \times \cdots \times \mathbb{P}^{n_k}$. This is an analogue of regeneration in a product of projective spaces, also known as *multiregeneration* [14, Section 4]. To ease exposition, we restrict the discussion to the case where $k = 2$, a product of two projective spaces. All results extend to an arbitrary number of factors. In Section 4.2, we study an example with $k = 3$ factors in detail.

3.1 Setting up the homotopy

We define the *double cone* of a biprojective variety $X \subset \mathbb{P}^m \times \mathbb{P}^n$ as

$$\tilde{X} = \overline{\{([u : \boldsymbol{x}],\ [v : \boldsymbol{y}]) \in \mathbb{P}^{m+1} \times \mathbb{P}^{n+1} \mid (\boldsymbol{x}, \boldsymbol{y}) \in X\}}.$$

The construction of the double cone $\tilde{X}$ when X is a single point already illustrates the main ideas needed to extend the homotopy of the previous section to the multiprojective case. For a projective curve X, we lift each witness point on X to d start points in $\tilde{X}$, where d is the degree of the intersecting hypersurface g_1. In the case of a biprojective curve $X \subset \mathbb{P}^m \times \mathbb{P}^n$, a witness point $(\boldsymbol{x}^\star, \boldsymbol{y}^\star)$ satisfies either $\ell_{\mathbf{x}}(\boldsymbol{x}^\star, \boldsymbol{y}^\star) = 0$ or $\ell_{\mathbf{y}}(\boldsymbol{x}^\star, \boldsymbol{y}^\star) = 0$. Suppose that, in either case, we lift the witness point to a point $([u : \boldsymbol{x}], [v : \boldsymbol{y}])$ in the double cone $\tilde{X}$ now requiring *both* $\ell_{\mathbf{x}}$ and $\ell_{\mathbf{y}}$ to vanish. This forces either $\boldsymbol{x}$ or $\boldsymbol{y}$ to be zero. However, if we deform the linear equations in the u and v directions, this will deform the lifted point so that both $\boldsymbol{x}$ and $\boldsymbol{y}$ are nonzero. This provides some intuition behind the construction of the multiprojective homotopy $\tilde{H}_t$ in (7) below.

Example 3.1 [Intersecting the double cone over a point] For a point $P = (\boldsymbol{x}^\star, \boldsymbol{y}^\star) \in \mathbb{P}^m \times \mathbb{P}^n$, the double cone $\tilde{P}$ is a two-dimensional plane spanned by four points in $\mathbb{P}^{m+1} \times \mathbb{P}^{n+1}$,

$$\tilde{P} = \langle([1 : \boldsymbol{x}^\star], [1 : \boldsymbol{y}^\star]), ([0 : \boldsymbol{x}^\star], [1 : \boldsymbol{y}^\star]), ([1 : \boldsymbol{x}^\star], [0 : \boldsymbol{y}^\star]), ([0 : \boldsymbol{x}^\star], [0 : \boldsymbol{y}^\star])\rangle.$$

Consider the set of d points in the intersection $\tilde{P} \cap \mathbb{V}(g, L_\epsilon)$, where the hypersurface $\mathbb{V}(g)$ is given by general $g \in \mathbb{C}[\{u, \boldsymbol{x}\}, \{v, \boldsymbol{y}\}]_{(d,e)}$ and the family of hyperplanes defined by $L_\epsilon := \epsilon v - (1 - \epsilon)\ell$ where $\ell \in \mathbb{C}[\boldsymbol{y}]_1$ is general. This set of points may be recovered in two easy steps:

1. Solve $L_\epsilon(\boldsymbol{y}^\star, v) = 0$ to obtain $v^\star = \epsilon\ell(\boldsymbol{y}^\star)/(1 - \epsilon)$.
2. Compute the d roots $u_1^\star, \ldots, u_d^\star$ of the univariate polynomial $g(u, \boldsymbol{x}^\star, v^\star, \boldsymbol{y}^\star)$.

Projecting a general affine piece of $\tilde{P}$ to the affine uv-plane $\mathbb{C}^2_{uv}$ is a surjection. The intersection $\tilde{P} \cap \mathbb{V}(g, L_\epsilon)$ can be visualized as the intersection of a line $\mathbb{V}(L_\epsilon(v, \boldsymbol{y}^\star))$ with the plane curve defined by $g(u, \boldsymbol{x}^\star, v, \boldsymbol{y}^\star) = 0$. Fig. 2 illustrates a case where $(d, e) = (2, 1)$. In the limit as $\epsilon \to 0$, $v^\star$ tends toward infinity, while the roots $u_1^\star, \ldots, u_d^\star$ tend towards the d vertical asymptotes of the curve given by g. If we write $g(u, \boldsymbol{x}^\star, v, \boldsymbol{y}^\star) = f(u)\, v^d + \text{additional terms}$, these asymptotes cross the u-axis at the points where u is a root of f.

Now, we consider the double cone over a curve $X \subset \mathbb{P}^m \times \mathbb{P}^n$ and state how they share some of the same degree information. Following the conventions in [13], the bidegree of a curve X in $\mathbb{P}^m \times \mathbb{P}^n$ with coordinate ring $\mathbb{C}[\boldsymbol{x}, \boldsymbol{y}]$ is

$$(\deg_{\boldsymbol{x}}(X), \deg_{\boldsymbol{y}}(X)) := (\deg(X \cap \mathbb{V}(\ell_{\boldsymbol{x}})), \deg(X \cap \mathbb{V}(\ell_{\boldsymbol{y}})))$$

where $\ell_{\boldsymbol{x}} \in \mathbb{C}[\boldsymbol{x}]_1$, $\ell_{\boldsymbol{y}} \in \mathbb{C}[\boldsymbol{y}]_1$ are general. Let $\ell_{u,\boldsymbol{x}} \in \mathbb{C}[u, \boldsymbol{x}]_1$ and $\ell_{v,\boldsymbol{y}} \in \mathbb{C}[v, \boldsymbol{y}]_1$ be general. Then, the bidegrees of the curves $X \subset \mathbb{P}^m \times \mathbb{P}^n$ and $\tilde{X} \cap \mathbb{V}(\ell_{u,\boldsymbol{x}}, \ell_{v,\boldsymbol{y}}) \subset \mathbb{P}^{m+1} \times \mathbb{P}^{n+1}$ coincide.

Remark 3.2 A polynomial $h \in \mathbb{C}[\boldsymbol{x}, \boldsymbol{y}]_{(d,e)}$ is defined to have bidegree (d, e). When $(m, n) = (1, 1)$, the polynomial h defines a curve $\mathbb{V}(h) \subset \mathbb{P}^1 \times \mathbb{P}^1$. The bidegree of $\mathbb{V}(h)$ and bidegree of the polynomial h are related by a transposition: $(\deg_{\boldsymbol{x}}(\mathbb{V}(h)), \deg_{\boldsymbol{y}}(\mathbb{V}(h))) = (e, d)$.

The following proposition is a multiprojective analogue of Proposition 2.1. Analogously to the homotopy in Eq. 3, for given $g_1 \in \mathbb{C}[\boldsymbol{x}, \boldsymbol{y}]_{d,e}$, we define

$$\tilde{H}_t := (g_0, \ell_{\boldsymbol{x}}, \ell_{\boldsymbol{y}}) \rightsquigarrow (g_1, u, v) \quad \text{on } \tilde{X}, \tag{7}$$

where $g_0 \in \mathbb{C}[\{u, \boldsymbol{x}\}, \{v, \boldsymbol{y}\}]_{d,e}$ is generic.

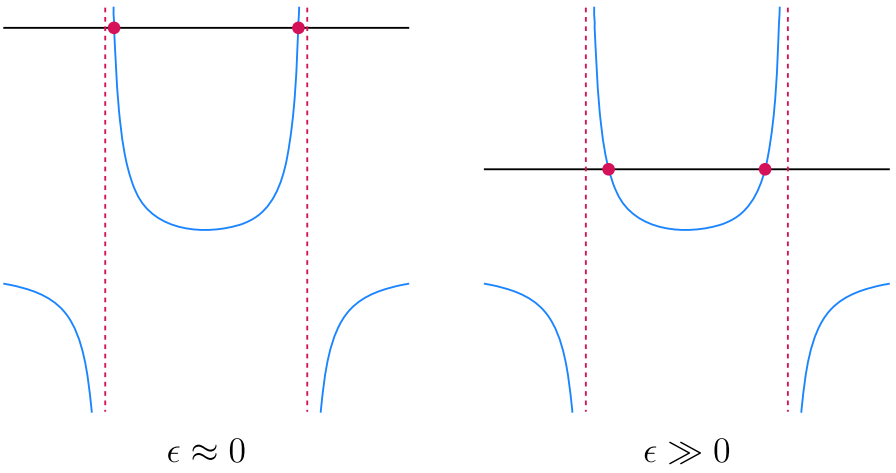

Fig. 2 Intersecting the double cone over a point $\tilde{P}$ with $\mathbb{V}(g, L_\epsilon)$ produces points $(u_1^\star, v^\star)$, $(u_2^\star, v^\star)$ in the affine plane plane $\mathbb{C}^2_{uv}$

Proposition 3.3 *For generic* $\ell_{\boldsymbol{x}} \in \mathbb{C}[\boldsymbol{x}]_1$, $\ell_{\boldsymbol{y}} \in \mathbb{C}[\boldsymbol{y}]_1$, $g_0 \in \mathbb{C}[\{u, \boldsymbol{x}\}, \{v, \boldsymbol{y}\}]_{d,e}$, *the cardinality of the set of points on* $\tilde{X}$ *satisfying* $\tilde{H}_t$ *equals, for all* $t \in (0, 1)$,

$$d \cdot \deg_{\boldsymbol{x}}(X) + e \cdot \deg_{\boldsymbol{y}}(X).$$

The endpoints of $\tilde{H}_t$ *lie in the set*

$$\left\{([0 : \boldsymbol{x}], [0 : \boldsymbol{y}]) \in \mathbb{P}^{m+1} \times \mathbb{P}^{n+1} \mid (\boldsymbol{x}, \boldsymbol{y}) \in X \cap \mathbb{V}(g_1)\right\}.$$

In case this set is finite, every point is reached.

Proof The homotopy $\tilde{H}_t$ is given by

$$g_t := (1-t)g_0 + tg_1 = 0, \quad (1-t)\ell_{\boldsymbol{x}} + tu = 0, \quad (1-t)\ell_{\boldsymbol{y}} + tv = 0 \tag{8}$$

For $t \in (0, 1)$, the intersection

$$Z_t := \tilde{X} \cap \mathbb{V}((1-t)\ell_{\boldsymbol{x}} + tu, (1-t)\ell_{\boldsymbol{y}} + tv). \tag{9}$$

is a curve in $\mathbb{P}^{m+1} \times \mathbb{P}^{n+1}$ with the same bidegree as $X \subset \mathbb{P}^m \times \mathbb{P}^n$, i.e.,

$$\deg_{\boldsymbol{x}}(X) = \deg_{u,\boldsymbol{x}}(Z_t) \text{ and } \deg_{\boldsymbol{y}}(X) = \deg_{v,\boldsymbol{y}}(Z_t).$$

since the last two equations give u and v as a linear functions of $\boldsymbol{x}$ and $\boldsymbol{y}$, respectively, when $t \in (0, 1)$.

Since g_0 is generic, so is g_t. Hence, for $t \in (0, 1)$, we get that $Z_t \cap V(g_t)$ has $d \cdot \deg_{u,\boldsymbol{x}}(Z_t) + e \cdot \deg_{v,\boldsymbol{y}}(Z_t)$ points of intersection. This proves the first part of the proposition.

The proof of the second part is similar to the proof of the second part of Proposition 2.1 concluding that a point $[\boldsymbol{x}_1 : \boldsymbol{x}_2] \in X \cap V(g_1)$ corresponds to an endpoint $([0 : \boldsymbol{x}_1], [0 : \boldsymbol{x}_2])$ of the homotopy.

Remark 3.4 If X is a general bidegree $(d_{\boldsymbol{x}}, d_{\boldsymbol{y}})$ curve in $\mathbb{P}^m \times \mathbb{P}^n$, then Z_0, as defined in (9), is a union of two linear spaces with multiplicity d and e, respectively. Specifically,

$$\begin{aligned} Z_0 = \quad & \{([u : \boldsymbol{x}], [1 : \boldsymbol{0}]) \in \tilde{X} : \ell_{\boldsymbol{x}}(\boldsymbol{x}) = 0\} \cup \\ & \{([1 : \boldsymbol{0}], [v : \boldsymbol{y}]) \in \tilde{X} : \ell_{\boldsymbol{y}}(\boldsymbol{y}) = 0\}. \end{aligned}$$

On the other hand, the homotopy $\tilde{H}_t$ of (7) converges to

$$\begin{aligned} \mathbb{V}(H_0) \cap \tilde{X} = & \{([u : \boldsymbol{x}], [1 : \boldsymbol{0}]) \mid \\ & \boldsymbol{x} \in \pi_{\boldsymbol{x}}(X \cap \mathbb{V}(\ell_{\boldsymbol{x}})), \text{ and } u \text{ satisfies } g_0(u, \boldsymbol{x}, 1, \boldsymbol{0}) = 0\} \\ \cup & \{([1 : \boldsymbol{0}], [v : \boldsymbol{y}]) \mid \\ & \boldsymbol{y} \in \pi_{\boldsymbol{y}}(X \cap \mathbb{V}(\ell_{\boldsymbol{y}})), \text{ and } v \text{ satisfies } g_0(1, \boldsymbol{0}, v, \boldsymbol{y}) = 0\} \end{aligned} \tag{10}$$

when $t \to 0$. Thus, multiple paths may tend to the same point as $t \to 0$.

3.2 Points for the homotopy at $t = \varepsilon$

In view of Remark 3.4 (see also Remark 3.5), we need to step away from $t = 0$ and find a way to produce a set of start points for the homotopy for $t = \varepsilon > 0$ for a small ε.

Fix an affine chart on $\mathbb{P}^{m+1} \times \mathbb{P}^{n+1}$; e.g., take $x_0 = y_0 = 1$. Without loss of generality, we may assume x_0 and y_0 do not vanish on any of the witness points. Consider a continuation path that tends to the point of the form $([u^* : \boldsymbol{x}^*], [1 : \mathbf{0}])$ as $t \to 0$ (a point that is not in the fixed chart) in a small punctured neighborhood of $t = 0$.

View the coordinates $(u(t), \boldsymbol{x}(t), v(t), \boldsymbol{y}(t))$ of this path as (convergent in the punctured neighborhood) Puiseux series where $u(t), \boldsymbol{x}(t), \boldsymbol{y}(t)$ have a (nonzero) constant leading term while $v(t)$ diverges as $t \to 0$. Note that the constant terms of $\boldsymbol{x}(t), \boldsymbol{y}(t)$ are given by a witness point $[\boldsymbol{x}^* : \boldsymbol{y}^*] \in X \cap \mathbb{V}(\ell_{\boldsymbol{x}})$ (with $x_0 = y_0 = 1$).

First, we note that since the path converges in $\mathbb{P}^{m+1} \times \mathbb{P}^{n+1}$ to a point $([u^* : \boldsymbol{x}^*] : [1 : \mathbf{0}])$, the leading term of $u(t)$ is a constant term. Plug $u(t)$ in the second equation in (8),

$$(1 - t)\ell_{\boldsymbol{x}} + tu = 0$$

and consider it on X (not $\tilde{X}$). It defines a continuous family of (transverse) $\boldsymbol{x}$-slices of X which limit at $\mathbb{V}(\ell_{\boldsymbol{x}})$ showing that $\boldsymbol{x}(t)$ and $\boldsymbol{y}(t)$ tend to $\boldsymbol{x}^* \neq 0$ and $\boldsymbol{y}^* \neq 0$ as $t \to 0$.

Recall that the end of the path has $[1 : \mathbf{0}]$ for $[v(t) : \boldsymbol{y}(t)]$; therefore, $v(t)$ has to diverge as $t \to 0$. Since $(1 - t)\ell_{\boldsymbol{y}} + tv = 0$, we have

$$v(t) = (1 - t^{-1})\ell_{\boldsymbol{y}}(\boldsymbol{y}(t)).$$

Thus, the leading term of $v(t)$ is $-\ell_{\boldsymbol{y}}(\boldsymbol{y}^*)t^{-1}$.

In summary, for a small ε, one may take $(u(\varepsilon), \boldsymbol{x}(\varepsilon), v(\varepsilon), \boldsymbol{y}(\varepsilon)) \approx (u^*, x^*, v^*, y^*)$ where $v^* = -\varepsilon^{-1}\ell_{\boldsymbol{y}}(y^*)$. Another asymptotically equivalent choice, which would satisfy the last equation, is $v^* = (1 - \varepsilon^{-1})\ell_{\boldsymbol{y}}(y^*)$.

An analogous argument holds if one reverses the roles of $(u, \boldsymbol{x})$ and $(v, \boldsymbol{y})$. Knowing the asymptotics of continuation paths as $t \to 0$ results in Algorithm 4 written in a chart-free form.

We conclude our discussion of computing start points with four remarks discussing issues relevant for implementation. In particular, Remarks 3.5 and 3.8 emphasize some pitfalls not encountered by the projective u-generation method of Section 2 due to singular start points. The heuristic approach of Algorithm 4, although sufficient for carrying out the experiments of Section 4.2, leaves something to be desired. It is quite possible that more sophisticated methods of function approximation (e.g., [34] and the references therein) could help to give multiprojective u-generation more of an edge over multiregeneration.

Remark 3.5 Note that, in Algorithm 4, starting at $t = \varepsilon > 0$ may be necessary not only due to several paths converging to the same point at $t = 0$ as pointed out in Remark 3.4, but also for the following reason that is bound to play a role in a practical implementation.

Algorithm 4 `uMultiprojectiveStartPoints`$(w_{\boldsymbol{x}}, w_{\boldsymbol{y}}, g_0, \varepsilon)$

Input: A biprojective curve $X \subset \mathbb{P}^m \times \mathbb{P}^n$ represented by two witness sets: $w_{\boldsymbol{x}} = (F, \ell_{\boldsymbol{x}}, W_{\boldsymbol{x}})$ where $\ell_{\boldsymbol{x}} \in \mathbb{C}[\boldsymbol{x}]_1$ and $w_{\boldsymbol{y}} = (F, \ell_{\boldsymbol{y}}, W_{\boldsymbol{y}})$ where $\ell_{\boldsymbol{y}} \in \mathbb{C}[\boldsymbol{y}]_1$; a general bihomogeneous polynomial $g_0 \in \mathbb{C}[\{u, \boldsymbol{x}\}, \{v, \boldsymbol{y}\}]_{(d,e)}$; a number $\varepsilon > 0$.

Output: A set of points that approximates $\mathbb{V}(\tilde{H}_\varepsilon)$ for the homotopy (7).

```
Initialize S ← ∅.
Let g_x ∈ ℂ[u, x] be defined as g_x(u, x) = g_0(u, x, 1, 0).
for P ∈ W_x do
    Update S ← S ∪ P̃ ∩ 𝕍(g_x, (1 − ε)ℓ_y + εv)
Let g_y ∈ ℂ[v, y] be defined as g_y(v, y) = g_0(1, 0, v, y).
for Q ∈ W_y do
    Update S ← S ∪ Q̃ ∩ 𝕍(g_y, (1 − ε)ℓ_x + εu)
return S
```

Let $F \subset \mathbb{C}[\boldsymbol{x}, \boldsymbol{y}]$ be polynomials that cut out the curve $X \subset \mathbb{P}^m \times \mathbb{P}^n$, i.e., X is a component of $\mathbb{V}(F)$. The double cone $\tilde{X}$ is a component of $\mathbb{V}(\tilde{F})$ where $\tilde{F} \subset \mathbb{C}[u, \boldsymbol{x}, v, \boldsymbol{y}]$ are polynomials F recast in a new ring. While this description of $\tilde{X}$ is practical and retains properties essential to our approach (as explained in Proposition 3.3), the variety $\mathbb{V}(\tilde{F})$ may possess extraneous components that contain points of (10) (even if they are regular points for $\tilde{H}_t$).

Remark 3.6 When $g_0 = g_1$, the homotopy (7) used in Algorithm 4 can be visualized in terms of intersecting double cones over a point with a hyperplane and hypersurface as in Fig. 2. Namely, we take the double cone over each point P in $W_{\boldsymbol{x}}$ (analogously for Q in $W_{\boldsymbol{y}}$) and intersect it with $\mathbb{V}\left(g_0, (1-\epsilon)\ell_{\boldsymbol{y}} + \epsilon v\right)$ as ϵ varies between 0 and 1.

Remark 3.7 The homotopy $\tilde{H}_t$ depends on the choice of sufficiently generic g_0. For implementation purposes, one may pick (use γ-type tricks if necessary)

1. $g_0 = (u^d - \ell_{\boldsymbol{x}}^d)(v^e - \ell_{\boldsymbol{y}}^e)$, where $\ell_{\boldsymbol{x}} \in \mathbb{C}[\boldsymbol{x}]_1$, $\ell_{\boldsymbol{y}} \in \mathbb{C}[\boldsymbol{y}]_1$ are general linear polynomials (similar to Remark 2.5), or
2. $g_0 = \prod_{i=1}^{d}(u - \ell_{\boldsymbol{x},i}) \prod_{j=1}^{e}(v - \ell_{\boldsymbol{y},j})$ where $\ell_{\boldsymbol{x},i} \in \mathbb{C}[\boldsymbol{x}]_1, \ell_{\boldsymbol{y},j} \in \mathbb{C}[\boldsymbol{y}]_1$ are general linear polynomials.

As mentioned in the beginning of this section, the results generalize straightforwardly to a curve X in a product of k projective spaces in any dimensions.

Remark 3.8 In u-generation for multihomogeneous systems, one may eliminate the additional variables (u and v in the case of two projective factors), as in Remark 2.6 when using generic affine charts in the projective factors.

However, there is a caveat: the conditioning of the resulting homotopy paths becomes much worse at $t = \varepsilon$, for small ε, in both cases (a) and (b) of Remark 2.6. In a practical implementation, one should eliminate additional variables only when the path tracking routine moves sufficiently far from $t = 0$.

3.3 Beyond curves

In the projective case, we show in Algorithm 2 how to apply u-generation for intersecting a d-dimensional projective variety with a hypersurface. Some additional work is required to extend these observations to the multiprojective setting. The variety X must now be represented by a *multiprojective witness collection* [14, Definition 1.2].

Example 3.9 The witness collection of an irreducible 3-fold $X = \mathbb{V}(F)$ in $\mathbb{P}^3 \times \mathbb{P}^1$ has a collection of witness point sets:

$$w_{3,0} := \mathbb{V}(F) \cap \mathbb{V}(\ell_{\boldsymbol{x},1}, \ell_{\boldsymbol{x},2}, \ell_{\boldsymbol{x},3}), \quad w_{2,1} := \mathbb{V}(F) \cap \mathbb{V}(\ell_{\boldsymbol{x},1}, \ell_{\boldsymbol{x},2}, \ell_{\boldsymbol{y},1}),$$
$$w_{1,2} := \mathbb{V}(F) \cap \mathbb{V}(\ell_{\boldsymbol{x},1}, \ell_{\boldsymbol{y},1}, \ell_{\boldsymbol{y},2}), \quad w_{0,3} := \mathbb{V}(F) \cap \mathbb{V}(\ell_{\boldsymbol{y},1}, \ell_{\boldsymbol{y},2}, \ell_{\boldsymbol{y},3}).$$

Note that $w_{a,b}$ with $a > 3$ or $b > 1$ is necessarily empty.

To describe the two dimensional variety $\mathbb{V}(F) \cap \mathbb{V}(g)$, we need to obtain the collection of witness points for $Y := \mathbb{V}(F) \cap \mathbb{V}(g)$:

$$w_{2,0}(Y) := Y \cap \mathbb{V}(\ell_{\boldsymbol{x},1}, \ell_{\boldsymbol{x},2}),$$
$$w_{1,1}(Y) := Y \cap \mathbb{V}(\ell_{\boldsymbol{x},1}, \ell_{\boldsymbol{y},1}),$$
$$w_{0,2}(Y) := Y \cap \mathbb{V}(\ell_{\boldsymbol{y},1}, \ell_{\boldsymbol{y},2}).$$

To obtain $w_{a,b}(Y)$, for $a + b = 2$, we run u-generation for the curve $\mathbb{V}(F) \cap \mathbb{V}(L)$ where $L = \{\ell_{\boldsymbol{x},1}, \ldots, \ell_{\boldsymbol{x},a}, \ell_{\boldsymbol{y},1}, \ldots, \ell_{\boldsymbol{y},b}\}$ using start points obtained by Algorithm 4 with $(F \cup L, \ell_{a+1}, w_{a+1,b})$ and $(F \cup L, \ell_{b+1}, w_{a,b+1})$ playing the role of $w_{\boldsymbol{x}}$ and $w_{\boldsymbol{y}}$.

In Example 3.9, we used three instances of reducing to the curve case to obtain $w_{a,b}(Y)$ with $a+b = 2, a \leq 3$ and $b \leq 1$. In a product of k projective $\mathbb{P}^{n_1} \times \cdots \times \mathbb{P}^{n_k}$, for a $b+1$-dimensional variety X and a hypersurface H not containing any irreducible component of X, to describe the intersection $Y = X \cap H$ one needs to obtain witness sets $w_{b_1,\ldots,b_k}(Y)$ with $b_1 + \cdots + b_k = b$ and $0 \leq b_i \leq n_i$. Each nonempty $w_{b_1,\ldots,b_k}$ is indexed by a lattice point $(b_1, \ldots, b_k)$ in a polymatroid base polytope associated to Y—see [13, §2.1] for details. Thus, the number of curves needed is the number of lattice points in this polytope, which is bounded above by the number of weak compositions of b with k parts and the ith part being less than n_i. If $b \leq n_i$ for all i, then there $\binom{b+k-1}{k-1}$ weak compositions of b with exactly k parts. This bound is exponential in k, or of polynomial order $O(b^k)$ when k is fixed. For the projective case, $k = 1$ and $\binom{b+k-1}{k} = 1$ as expected.

In the previous section, u-generation and regeneration can both be used inside of the equation-by-equation cascade of Algorithm 3 based on intersection with a hypersurface in projective space. The analogue in a product of projective spaces is known as *multiregeneration* [14, Section 4]. A variant based on u-generation may be developed using the ideas outlined above.

Finally, we note that systems encountered in practice may have different possible variable groupings. The upper bounds discussed above show we may need many more curves if $k > 1$ variable groups are used; however, there are potential savings in the size of each individual witness set. How to best negotiate the trade-off between the

number of witness sets and the total number of witnes points depends heavily on the problem at hand. On the other hand, for any *fixed* grouping, multiregeneration and multiprojective u-generation work with the exact same number of curves. Thus, when comparing the two methods on a given problem, it is reasonable to work with whatever variable grouping is most natural.

4 Computational experiments

In this section, we report the computational results using our initial implementation of u-generation. For a fair comparison, we also implement regeneration in a similar fashion, using the same homotopy tracking toolkit provided by the package `NumericalAlgebraicGeometry` [24] in `Macaulay2` [11]. Default path-tracker settings for the method `trackHomotopy` are used, except where noted. All computations were performed using a 2012 iMac with 16GB, working at 1.6GHz.

Once again, we point out that these experiments serve only as a proof-of-concept. Our modest aim is to convince the practitioner that u-generation is competitive against regeneration, and that the initial successes of our implementation motivate further investigation and more refined heuristics.

4.1 Intersecting with a hypersurface in projective space

The u-generation-based Algorithm 2 for computing $X \cap \mathbb{V}(g_1)$ requires tracking $\deg(g_1)\deg(X)$-many paths. Regeneration also requires tracking this many paths, plus an additional $(\deg(g_1)-1)\deg(X)$ paths during the preparation stage. In a rough analysis where we assume all paths have the same cost, we would expect that

$$\frac{t_{u\text{-gen}}}{t_{\text{prep}}+t_{\text{regen}}} \approx \frac{\deg(g_1)\deg(X)}{2\deg(g_1)\deg(X)-\deg(X)} = \frac{1}{2-1/\deg(g_1)}, \tag{11}$$

where t_{prep}, t_{regen}, and t_{ugen} denote the total time spent tracking paths during preparation, regeneration, or u-generation, respectively. This analysis would then suggest that u-generation has $\approx 33.3\%$ savings over regeneration when $\deg(g_1)=2$, and an asymptotic 50% savings when $\deg(g_1)$ is large.

The assumption that paths tracked during preparation cost the same as other paths does not hold in practice. This is partly because homotopy functions involved in the preparation stage are simpler and also due to the fact we do not expect any paths to diverge during this stage. It is therefore worth investigating to what extent, if any, our proposed method may actually deliver any savings. To that end, we considered two well-studied families of benchmark polynomial systems and performed the following experiment for each:

1. Drop an equation g_1 from the homogenized system, and compute a projective witness set for the projective curve X defined by the remaining equations.

2. Run Algorithm 2 to solve the original system.
3. Similarly to the previous step, use regeneration to solve the original system.

For all systems considered in this subsection, the witness set for X is computed using a total-degree homotopy. The timings we report for Steps 2 and 3 above do not reflect the full cost of solving these systems from scratch with equation-by-equation methods. Nevertheless, this experiment is sufficient to make a meaningful comparison between regeneration and u-generation.

The first of the benchmark systems studied is the Katsura-n family, which arose originally in the study of magnetism [22]. For a given n, this is a system of inhomogeneous equations in $n+1$ unknowns $x_0, \ldots, x_n$: writing $x_{-i} = x_i$,

$$\sum_{i=-n}^{n} x_i = 1,$$
$$\sum_{i=-n}^{n} x_l x_{m-l} = x_m, \quad m \in \{-n+1, -n, .., n-1\}.$$

For this family, the Bézout bound on the number of roots 2^{n-1} is tight. In our experiments, the dropped equation g_1 is one of the n quadratic equations. We also considered the classic cyclic n-roots problem [1]:

$$\sum_{i=1}^{n} x_i \cdot x_{i+1} \cdots x_{(i+m)\%n} = 0, \quad m \in \{1, \ldots, n-1\},$$
$$x_0 \cdot x_1 \cdots x_{n-1} - 1 = 0.$$

This system has infinitely many solutions when n is not squarefree. For small, squarefree values of n, this system has been observed to have all isolated solutions, their number matching the polyhedral bound of Bernstein's theorem [4]. As a result, the polyhedral homotopy [19] and its various implementations [5, 7, 35] are better-suited for this problem than equation-by-equation methods. We include it in our experiments as a further point of comparison between regeneration and u-generation. Based on the naive analysis, we might expect substantial savings when dropping the equation with $\deg(g_1) = n$.

In Fig. 3, we display the results of our experiment on these benchmark problems. Timings for each of the methods were averaged over 100 iterations. For cyclic-7, we observed multiple path failures for both methods with the default tracker settings, so we used a more permissive minimum stepsize of 10^{-8} for this case. We observe in all cases that u-generation outperforms regeneration in terms of runtime. Contrary to what the naive analysis would predict, typical savings are in the range of 10–15% for both the Katsura and cyclic benchmarks.

To further investigate the nature of potential savings of u-generation over regeneration, we considered the following family of *banded quadrics*. Fixing integers

		regeneration			u-generation		
system	# solutions	# paths	t_{prep}	t_{regen}	# paths	$t_{u\text{-gen}}$	$\frac{t_{u\text{-gen}}}{t_{\text{prep}}+t_{\text{regen}}}$
katsura-8	256	384	1.1×10^{-1} s	2.5×10^{-1} s	256	3.2×10^{-1} s	**.87**
katsura-9	512	786	2.7×10^{-1} s	6.0×10^{-1} s	512	7.3×10^{-1} s	**.84**
katsura-10	1024	1536	6.2×10^{-1} s	1.5 s	1024	1.8 s	**.86**
katsura-11	2048	3072	1.5 s	3.2 s	2048	3.9 s	**.83**
katsura-12	4096	6144	3.6 s	8.5 s	4096	10.4 s	**.85**
katsura-13	8192	12 288	9.3 s	22.7 s	8192	26.9	**.84**
cyclic-5	70	126	2.1×10^{-2} s	4.4×10^{-2} s	70	5.6×10^{-2} s	**.87**
cyclic-6	156	286	5.7×10^{-2} s	1.4×10^{-1} s	156	1.8×10^{-1} s	**.90**
cyclic-7	924	1716	4.5×10^{-1} s	1.4 s	924	1.7 s	**.89**

Fig. 3 Timing Algorithm 2 for benchmark polynomial systems

$2 \leq k \leq n$, this is a homogeneous system $f_1, \ldots, f_n \in \mathbb{C}[\boldsymbol{x}]$ where f_1 is linear and $f_2, \ldots, f_n$ have the form

$$f_i(\boldsymbol{x}) = \begin{bmatrix} x_i & \cdots & x_{(i+k \mod n)} \end{bmatrix} \begin{bmatrix} c_{i,1,1} & \cdots & c_{i,1,k} \\ \vdots & \ddots & \vdots \\ c_{i,k,1} & \cdots & c_{i,k,k} \end{bmatrix} \begin{bmatrix} x_i \\ \vdots \\ x_{(i+k \mod n)} \end{bmatrix}.$$

Here, the real and imaginary parts of the parameters $c_{i,j,k}$ are drawn randomly from the interval $[0, 1]$. We apply the same experiment from before to the homogeneous system, using a projective witness set for the projective curve $\mathbb{V}(f_1, \ldots, f_{n-1})$ to compute the intersection $\mathbb{V}(f_1, \ldots, f_{n-1}) \cap \mathbb{V}(f_n)$. Timings for $n = 12$ are given in Fig. 4.

One possible conclusion to draw from this experiment is that we may expect more savings from u-generation when the equations involved are more sparse. This is plausible for the case of banded quadrics when the value of k is small in comparison to n, in which case the cost of tracking a path in the preparation stage is more comparable to the cost for the paths in the second stage of regeneration. On the other hand, we still observe in this experiment a steady 10 % savings as k approaches n.

		regeneration			u-generation		
(n, k)	# solutions	# paths	t_{prep}	t_{regen}	# paths	$t_{u\text{-gen}}$	$\frac{t_{u\text{-gen}}}{t_{\text{prep}}+t_{\text{regen}}}$
$(12, 2)$	2048	3072	8.9×10^{-1} s	3.0 s	2048	3.4 s	**.87**
$(12, 3)$	2048	3072	1.3 s	3.3 s	2048	3.9 s	**.87**
$(12, 4)$	2048	3072	1.3 s	3.7 s	2048	4.5 s	**.90**
$(12, 5)$	2048	3072	1.5 s	4.0 s	2048	5.0 s	**.90**
$(12, 6)$	2048	3072	1.8 s	4.6 s	2048	5.6 s	**.89**
$(12, 7)$	2048	3072	1.9 s	4.8 s	2048	6.1 s	**.90**
$(12, 8)$	2048	3072	2.2 s	5.9 s	2048	7.0 s	**.90**
$(12, 9)$	2048	3072	2.5 s	6.1 s	2048	7.7 s	**.90**

Fig. 4 Timing Algorithm 2 for banded quadrics with the number of variables fixed by $n = 12$. Each system is dehomogenized with a random chart, giving a system in 13 unknowns

4.2 Maximum likelihood estimation for matrices with rank constraints

Consider a probabilistic experiment where we flip two coins. Each of the coins may be biased, but we model them with the same binary probability distribution. Let p_{11} denote the probability of obtaining two heads, p_{22} the probability of two tails, and $p_{12} = p_{21}$ the probability of one head and one tail in either order. The individual coin flips are independent if and only if

$$\operatorname{rank} \begin{bmatrix} 2p_{11} & p_{12} \\ p_{12} & 2p_{22} \end{bmatrix} \leq 1. \tag{12}$$

Similarly, the independence of two identically distributed n-ary random variables is modeled by a $n \times n$ symmetric matrix of rank 1. More generally, rank constraints

$$\operatorname{rank} \begin{bmatrix} 2p_{1n} & p_{12} & \cdots & p_{1n} \\ p_{12} & 2p_{22} & & p_{2n} \\ \vdots & & \ddots & \vdots \\ p_{1n} & p_{2n} & \cdots & 2p_{nn} \end{bmatrix} \leq r \tag{13}$$

must be satisfied for all points in the *symmetric rank-constrained model*. This statistical model is defined as the set of all points in the probability simplex

$$\{(p_{11}, \ldots, p_{nn}) \in \mathbb{R}^{(n^2+n)/2} \mid \sum_{1 \leq i \leq j \leq n} p_{ij} = 1,\ p_{ij} \geq 0\}$$

which satisfy the rank constraints (13). It is an algebraic relaxation of an associated r-th mixture model [12, 23].

If we are given a $n \times n$ symmetric matrix U counting several samples from this statistical model, it is of interest to recover the underlying model parameters $(p_{ij})_{1 \leq i \leq j \leq n}$. A popular approach in statistical inference is *maximum likelihood estimation*. In our case of interest, this means computing a global maximum of the likelihood function

$$\ell_U(p_{11}, \ldots, p_{nn}) \quad = \quad \frac{\prod_{i \leq j} p_{ij}^{u_{ij}}}{\left(\sum_{i \leq j} p_{ij}\right)^{\sum_{i \leq j} u_{ij}}} \tag{14}$$

restricted to the symmetric rank-constrained model. Local optimization heuristics such as the EM algorithm or gradient ascent are popular methods for maximum-likelihood estimation, but are susceptible to local maxima. For generic U, the number of critical points of ℓ_U on the set of complex matrices satisfying (13) is the *ML degree* of the symmetric rank-constrained model. For statistical models in general, the ML degree is an important measure of complexity and a well-studied quantity in algebraic statistics (see [33, Ch. 7]).

Previous work of Hauenstein, the third author, and Sturmfels [12] demonstrates that homotopy continuation methods are a powerful technique for computing the critical

points of ℓ_U. In [12, Theorem 2.3], a table of ML degrees for the symmetric rank-constrained model for $n \leq 6$ was obtained using the software `Bertini` [2]. In that table, ML degrees for $(n, r) \in \{(6, 2), (6, 3)\}$ were excluded. We provide the missing entries of that table in Fig. 5.

The symmetry in each column of the table of ML degrees is explained by a duality result of Draisma and the third author [8, Theorem 11], establishing a remarkable bijection between the critical points for the rank-r and rank-$(n-r+1)$ models. Thus, to complete the table, it suffices to compute the ML degree for $(n, r) = (6, 3)$, which we report to be **68774**. We managed to verify the ML degrees in this table using several different techniques: equation-by-equation methods, monodromy [9, 26], and several combinations thereof. We focus our discussion of computational results on the equation-by-equation methods, which we hope may serve as a useful starting point for future benchmarking studies.

To make computing the ML degrees more amenable to equation-by-equation approach, we use a *symmetric local kernel formulation* of the problem, which is a square polynomial system in $\binom{n+1}{2}$ unknowns [12, Eq 2.13]. The unknowns are the entries of three auxiliary matrices,

$$P_1 := \begin{bmatrix} 2p_{12} & p_{12} & \cdots & p_{1r} \\ p_{12} & 2p_{22} & & p_{2r} \\ \vdots & & \ddots & \vdots \\ p_{1r} & p_{2r} & \cdots & 2p_{rr} \end{bmatrix}, \quad L_1 := \begin{bmatrix} \ell_{1,1} & \cdots & \ell_{1,r} \\ \vdots & \ddots & \vdots \\ \ell_{n-r,1} & \cdots & \ell_{n-r,r} \end{bmatrix},$$

$$\Lambda := \begin{bmatrix} \lambda_{11} & \lambda_{12} & \cdots & \lambda_{1,n-r} \\ \lambda_{12} & \lambda_{22} & & \lambda_{2,n-r} \\ \vdots & & \ddots & \vdots \\ \lambda_{1,n-r} & \lambda_{2,n-r} & \cdots & \lambda_{r,n-r} \end{bmatrix}.$$

$r \setminus n$	2	3	4	5	6
1	1	1	1	1	1
2	1	6	37	270	2341
3		1	37	1394	**68774**
4			1	270	**68774**
5				1	2341
6					1

Fig. 5 ML degrees for the symmetric rank-constrained model, for $n \times n$ symmetric matrices of rank $\leq r$, for all $1 \leq r \leq n \leq 6$

The equations which make up this square system are the column sums and the entries above the diagonal in the $n \times n$ symmetric matrix

$$\begin{bmatrix} P_1 & P_1 L_1^T \\ L_1 P_1 & L_1 P_1 L_1^T \end{bmatrix} \odot \begin{bmatrix} L_1^T \Lambda L_1 & \Lambda L_1 \\ L_1 \Lambda & \Lambda \end{bmatrix} + \sum_{i \le j} u_{ij} \begin{bmatrix} P_1 & P_1 L_1^T \\ L_1 P_1 & L_1 P_1 L_1^T \end{bmatrix} - 2 I_n \odot U, \quad (15)$$

where $\odot$ denotes the Hadamard product. There are three natural variable groups given by the auxiliary matrices P_1, L_1, and Λ. Dropping a single equation g_1 from our square system gives equations which vanish on an affine patch of an irreducible curve

$$X \subset \mathbb{P}^{r \times r} \times \mathbb{P}^{(n-r) \times r} \times \mathbb{P}^{(n-r+1) \times (n-r)/2}.$$

Slicing in each of the three projective factors, we may compute three sets of witness points $W_{P_1}, W_{L_1}, W_\Lambda$ for X using monodromy. These witness points can be used to compute start points for both regeneration and u-generation. In our experiments, the dropped equation g_1 is the $(1, n)$-th entry of the matrix (15), which has degree $(1, 2, 1)$. Thus, there are $\#W_{P_1} + 2\#W_{L_1} + \#W_{\Lambda_1}$ start points for both methods, and an additional $\#W_{L_1}$ paths must be tracked in the preparation phase for regeneration. For u-generation, the start points were obtained using the heuristic proposed in Algorithm 4 with $\epsilon = 10^{-5}$. We do not eliminate the three u-variables corresponding to each factor—see Remark 2.6.

We use two non-default path-tracker tolerances: a minimum stepsize of 10^{-14} by setting `tStepMin=>1e-14` and a maximum of 2 Newton iterations for every predictor step by setting `maxCorrSteps=>2`. The latter option is more conservative than the default of ≤ 3 Newton steps, which increases the chances of path-jumping. With fewer corrector steps, a smaller timestep may be needed to track paths successfully. We note that comparable tolerances are the defaults used in `Bertini` [3, Appendix E.4.4, E.4.7].

Figure 6 and 8 show that the ML degrees are significantly smaller than the number of start points. This implies that many endpoints will lie at the hyperplane at infinity in one of the projective factors. We declare an endpoint to be finite when its three homogeneous coordinates exceed 10^{-6} in magnitude.

In Fig. 6, we list timings for computing the previously-known ML degrees. We see that the number of additional paths required by regeneration is rather small in comparison with the ML degree. Thus, we should not expect to see much of an advantage

		regeneration			u-generation		
(n, r)	ML degree	#paths	t_{prep}	t_{regen}	#paths	$t_{u\text{-gen}}$	$\frac{t_{u\text{-gen}}}{t_{\text{prep}}+t_{\text{regen}}}$
(3,2)	6	19	3×10^{-3} s	3×10^{-2} s	15	5×10^{-2} s	**1.52**
(4,2)	37	156	6×10^{-2} s	5×10^{-1} s	118	6×10^{-1} s	**1.07**
(5,2)	270	1313	3×10^{-1} s	6 s	980	9 s	**1.43**
(5,3)	1394	6559	5 s	55 s	5185	66 s	**1.10**
(6,2)	2341	9137	8 s	94 s	12 343	121 s	**1.18**

Fig. 6 Timings of regeneration and u-generation for computing previously-known ML degrees

of u-generation over regeneration. Indeed, although the timings of both methods are competitive, regeneration is the clear winner. Although t_{regen} and $t_{u-\text{gen}}$ time the same number of path-tracks, we see that $t_{u\text{-gen}}$ is consistently larger. One explanation is that u-generation uses an extra three variables here, increasing costs associated with repeated function evaluation and numerical linear algebra.

Remark 4.1 The heuristic for computing start points in Algorithm 4 imposes an additional overhead for the multiprojective u-generation. This routine is implemented in the top-level language of `Macaulay2` and currently takes a significant portion of the runtime. Since this heuristic may be improved in future work, we did not pursue a low-level implementation, which would dramatically reduce the runtime of this routine making it negligible in comparison to the other parts.

For the previously-unknown ML degree when $(n, r) = (6, 3)$, we conducted an experiment where we ran each of the two equation-by-equation methods four times and used the *union* of all finite endpoints to estimate the ML degree. The four runs are not identical because the homotopies are randomized using the γ-trick. Timings are shown in Fig. 7. The variability of the timings is not so surprising, since paths are randomized according to γ-type tricks and also due to the permissive step-size of 10^{-14}; if there are many ill-conditioned paths, the adaptively-chosen stepsize for each of them may shrink and remain small throughout path-tracking. Ultimately, both methods yield the same root count, which u-generation attains by the third iteration. We also note that the total timing for regeneration and u-generation turn out to be very close, though this might well be an anomaly.

One might reasonably be concerned that multiple runs of both equation-by-equation methods were needed to compute the ML degree for $(n, r) = (6, 3)$. In theory, both are probability-one methods. In practice, most implementations of homotopy continuation will miss some solutions when presented with a sufficiently difficult problem. One well-known practical issue is path-jumping, which may lead in some cases to duplicate endpoints. The existence of many solutions at infinity, which are often highly-singular, presents another obstacle. This obstacle would be even more of a concern if we were to consider other homotopy continuation methods. We note, for instance, the number of paths tracked by the celebrated polyhedral homotopy may be prohibitive for the $(n, r) = (6, 3)$ case. Using [5] and [20], which both implement the mixed volume algorithm described in [21], we determined that the polyhedral root count for the symmetric local kernel formulation is **27174865**—three orders of magnitude over the ML degree, and two over the number of paths tracked in our experiments.

Figure 8 displays the results of another experiment where we ran each equation-by-equation method *exactly once* and collected any missed solutions using a monodromy loop. A similar strategy for dealing with failed solutions is outlined in [6, Sec. 3.2]. This experiment gives us more confidence in the reported root count of **68774**. For

method	t_1	# collected	t_2	# collected	t_3	# collected	t_4	# collected	t_{total}
regeneration	12 498 s	68 767	12 666 s	68 773	20 210 s	68 773	20 393 s	**68 774**	65 767 s
u-generation	13 723 s	68 729	13 900 s	68 773	25 433 s	**68 774**	13 689 s	**68 774**	66 745 s

Fig. 7 Timings t_1, t_2, t_3, t_4 of four runs of regeneration and u-generation for computing the ML degree of **68774** for $(n, r) = (6, 3)$

		regeneration					u-generation			
(n,r)	ML degree	#paths	t_{prep}	t_{regen}	t_{mon}	t_{total}	#paths	$t_{u\text{-gen}}$	t_{mon}	t_{total}
(5,2)	270	1313	7×10^{-1} s	5 s	9 s	14 s	980	9 s	10 s	19 s
(5,3)	1394	6559	4 s	47 s	34 s	85 s	5185	61 s	52 s	113 s
(6,2)	2341	9137	7 s	88 s	124 s	219 s	12 343	110 s	109 s	219 s
(6,3)	68 774	387 114	1136 s	11 822 s	6557 s	19 515 s	301 675	13 883 s	6509 s	20 392 s

Fig. 8 Regeneration and u-generation followed by a monodromy loop

$(n,r) = (6,3)$, the total runtime compares favorably to Fig. 7, suggesting another strategy the practitioner may keep in mind. We point out that such strategies may be of interest in numerical irreducible decomposition, where the *monodromy breakup algorithm* [29] is run following the cascade of Algorithm 3.

We close with a concrete numerical example of maximum-likelihood estimation, where the target system and its solutions resulting from u-generation may now serve as the start-system for a parameter homotopy [31, Ch. 7].

Example 4.1 Consider the matrix of counts

$$U = \begin{bmatrix} 2 & 4 & 2 & 4 & 4 & 4 \\ 4 & 2 & 2 & 6 & 6 & 6 \\ 2 & 2 & 1 & 2 & 2 & 2 \\ 4 & 6 & 2 & 3 & 6 & 6 \\ 4 & 6 & 2 & 6 & 3 & 6 \\ 4 & 6 & 2 & 6 & 6 & 3 \end{bmatrix}.$$

Let P_{emp} denote the empirical P-matrix

$$P_{\text{emp}} = \frac{1}{\sum_{i \le j} u_{ij}} \begin{bmatrix} 2u_{11} & u_{12} & \cdots & u_{16} \\ u_{12} & 2u_{22} & \cdots & u_{26} \\ \vdots & \vdots & \ddots & \vdots \\ u_{16} & u_{26} & \cdots & 2u_{66} \end{bmatrix}$$

This matrix has rank 4. To compute a maximum-likelihood estimate given U, we use a parameter homotopy with **68774** start points furnished by u-generation. Among the approximate endpoints of this parameter homotopy are 1082 labeled as failed paths, which have large and non-real coordinates. Among the successful endpoints, there are 4108 real solutions. However, only three are statistically valid in the sense that the rank-constrained P-matrix has non-negative entries. This matrix may be recovered with the formula

$$P = \begin{bmatrix} P_1 & P_1 L_1^T \\ L_1 P_1 & L_1 P_1 L_1^T \end{bmatrix}.$$

We list the P-matrices and likelihood values for the three statistically valid critical points, along with the empirical P-matrix (which does not lie on the rank-constrained

model):

$$P^{(1)} \approx \begin{pmatrix} .046 & .053 & .03 & .054 & .054 & .054 \\ .053 & .053 & .026 & .079 & .079 & .079 \\ .030 & .026 & .024 & .026 & .026 & .026 \\ .054 & .079 & .026 & .079 & .079 & .079 \\ .054 & .079 & .026 & .079 & .079 & .079 \\ .054 & .079 & .026 & .079 & .079 & .079 \end{pmatrix}, \quad \log \ell_U(P^{(1)}) = -223.264,$$

$$P^{(2)} \approx \begin{pmatrix} .053 & .051 & .026 & .053 & .053 & .053 \\ .051 & .069 & .026 & .074 & .074 & .074 \\ .026 & .026 & .026 & .026 & .026 & .026 \\ .053 & .074 & .026 & .080 & .080 & .080 \\ .053 & .074 & .026 & .080 & .080 & .080 \\ .053 & .074 & .026 & .080 & .080 & .080 \end{pmatrix}, \quad \log \ell_U(P^{(2)}) = -222.979,$$

$$P^{(3)} \approx \begin{pmatrix} .048 & .054 & .020 & .056 & .056 & .056 \\ .054 & .052 & .029 & .077 & .077 & .077 \\ .020 & .029 & .014 & .032 & .032 & .032 \\ .056 & .077 & .032 & .077 & .077 & .077 \\ .056 & .077 & .032 & .077 & .077 & .077 \\ .056 & .077 & .032 & .077 & .077 & .077 \end{pmatrix}, \quad \log \ell_U(P^{(3)}) = -222.901,$$

$$P_{\text{emp}} \approx \begin{pmatrix} .053 & .053 & .026 & .053 & .053 & .053 \\ .053 & .053 & .026 & .079 & .079 & .079 \\ .026 & .026 & .026 & .026 & .026 & .026 \\ .053 & .079 & .026 & .079 & .079 & .079 \\ .053 & .079 & .026 & .079 & .079 & .079 \\ .053 & .079 & .026 & .079 & .079 & .079 \end{pmatrix}, \quad \log \ell_U(P_{\text{e}}) = -222.860.$$

The matrix $P^{(3)}$ is a good candidate for the maximum likelihood estimate.

5 Conclusion

We propose u-generation as a novel equation-by-equation homotopy continuation method for solving polynomial systems. The main theoretical merits of this method are that it works in both projective and multiprojective settings and requires tracking fewer total paths than regeneration. Furthermore, our setup is easily implementable. Computational experiments show that the method is promising and can be used to solve nontrivial problems, although we do not observe a decisive advantage of u-generation over regeneration. More specifically, our proof-of-concept implementation achieves modest, but notable, savings in terms of computational time on several examples. Furthermore, we show that multiprojective u-generation is a viable method for solving structured polynomial systems arising in maximum-likelihood estimation, including one previously unsolved case.

As a future direction, we point out that more thorough experimentation with heuristics such as Algorithm 4 might lead to a more robust implementation of u-

generation. Further exploration in both projective and multiprojective settings is warranted, particularly in the context of numerical irreducible decomposition, where equation-by-equation methods play a vital role.

Author Contributions All authors contributed to the preparation of the manuscript and the research it contains.

Funding T.D. acknowledges support from an NSF Mathematical Sciences Postdoctoral Research Fellowship (DMS-2103310). Research of A.L. is supported in part by NSF DMS-2001267. Research of J.I.R. is supported by the Office of the Vice Chancellor for Research and Graduate Education at U.W. Madison with funding from the Wisconsin Alumni Research Foundation.

Data availability Code available upon request.

Declarations

Ethical approval N/A

Competing interests The authors declare no competing interests.

References

1. J. Backelin and R. Fröberg. How we proved that there are exactly 924 cyclic 7-roots. In S. M. Watt, editor, Proceedings of the 1991 International Symposium on Symbolic and Algebraic Computation, ISSAC '91, Bonn, Germany, July 15-17, 1991, pages 103–111. ACM, 1991
2. D. J. Bates, J. D. Hauenstein, A. J. Sommese, and C. W. Wampler. Bertini: Software for Numerical Algebraic Geometry. Available at bertini.nd.edu with permanent doi: dx.doi.org/10.7274/R0H41PB5
3. D. J. Bates, J. D. Hauenstein, A. J. Sommese, and C. W. Wampler. Numerically solving polynomial systems with Bertini, volume 25 of Software, Environments, and Tools. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2013
4. Bernstein, D.N.: The number of roots of a system of equations. Funkcional. Anal. i Prilož. **9**(3), 1–4 (1975)
5. P. Breiding and S. Timme. Homotopycontinuation.jl: A package for homotopy continuation in Julia. In Mathematical Software – ICMS 2018, pages 458–465. Springer International Publishing, 2018
6. Brysiewicz, T., Rodriguez, J.I., Sottile, F., Yahl, T.: Decomposable sparse polynomial systems. J. Softw. Algebra Geom. **11**(1), 53–59 (2021)
7. 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 International Congress on Mathematical Software, pages 183–190. Springer, 2014
8. Draisma, J., Rodriguez, J.: Maximum likelihood duality for determinantal varieties. Int. Math. Res. Not. IMRN **20**, 5648–5666 (2014)
9. Duff, T., Hill, C., Jensen, A., Lee, K., Leykin, A., Sommars, J.: Solving polynomial systems via homotopy continuation and monodromy. IMA J. Numer. Anal. **39**(3), 1421–1446 (2019)
10. Giusti, M., Lecerf, G., Salvy, B.: A Gröbner free alternative for polynomial system solving. J. Complexity **17**(1), 154–211 (2001)
11. D. R. Grayson and M. E. Stillman. Macaulay2, a software sy,stem for research in algebraic geometry. Available at http://www.math.uiuc.edu/Macaulay2/
12. Hauenstein, J., Rodriguez, J.I., Sturmfels, B.: Maximum likelihood for matrices with rank constraints. J. Algebr. Stat. **5**(1), 18–38 (2014)
13. Hauenstein, J.D., Leykin, A., Rodriguez, J.I., Sottile, F.: A numerical toolkit for multiprojective varieties. Math. Comp. **90**(327), 413–440 (2021)
14. Hauenstein, J.D., Rodriguez, J.I.: Multiprojective witness sets and a trace test. Adv. Geom. **20**(3), 297–318 (2020)

15. Hauenstein, J.D., Sommese, A.J.: Witness sets of projections. Appl. Math. Comput. **217**(7), 3349–3354 (2010)
16. Hauenstein, J.D., Sommese, A.J., Wampler, C.W.: Regeneration homotopies for solving systems of polynomials. Math. Comp. **80**(273), 345–377 (2011)
17. Hauenstein, J.D., Sommese, A.J., Wampler, C.W.: Regenerative cascade homotopies for solving polynomial systems. Appl. Math. Comput. **218**(4), 1240–1246 (2011)
18. Hauenstein, J.D., Wampler, C.W.: Unification and extension of intersection algorithms in numerical algebraic geometry. Appl. Math. Comput. **293**, 226–243 (2017)
19. Huber, B., Sturmfels, B.: A polyhedral method for solving sparse polynomial systems. Math. Comp. **64**(212), 1541–1555 (1995)
20. A. N. Jensen. Gfan, a software system for Gröbner fans and tropical varieties. Available at [SPSurl1urlSPS]
21. A. N. Jensen. Tropical homotopy continuation. arXiv preprint arXiv:1601.02818, 2016
22. S. Katsura. Spin glass problem by the method of integral equation of the effective field. New Trends in Magnetism, pages 110–121, 1990
23. Kubjas, K., Robeva, E., Sturmfels, B.: Fixed points EM algorithm and nonnegative rank boundaries. Ann. Statist. **43**(1), 422–461 (2015)
24. Leykin, A.: Numerical algebraic geometry. Journal of Software for Algebra and Geometry **3**(1), 5–10 (2011)
25. A. Leykin, J. Verschelde, and A. Zhao. Higher-order deflation for polynomial systems with isolated singular solutions. In Algorithms in algebraic geometry, volume 146 of IMA Vol. Math. Appl., pages 79–97. Springer, New York, 2008
26. A. Martín del Campo and J. I. Rodriguez. Critical points via monodromy and local methods. J. Symbolic Comput., 79(part 3):559–574, 2017
27. D. Mumford. Stability of projective varieties. Enseign. Math. (2), 23(1-2):39–110, 1977
28. A. J. Sommese and J. Verschelde. Numerical homotopies to compute generic points on positive dimensional algebraic sets. journal of complexity, 16(3):572–602, 2000
29. A. J. Sommese, J. Verschelde, and C. W. Wampler. Symmetric functions applied to decomposing solution sets of polynomial systems. SIAM J. Numer. Anal., 40(6):2026–2046 (2003), 2002
30. A. J. Sommese, J. Verschelde, and C. W. Wampler. Solving polynomial systems equation by equation. In Algorithms in algebraic geometry, volume 146 of IMA Vol. Math. Appl., pages 133–152. Springer, New York, 2008
31. A. J. Sommese and C. W. Wampler, II. The numerical solution of systems of polynomials. World Scientific Publishing Co. Pte. Ltd., Hackensack, NJ, 2005. Arising in engineering and science
32. F. Sottile. General witness sets for numerical algebraic geometry. In Proceedings of the 45th International Symposium on Symbolic and Algebraic Computation, ISSAC '20, page 418–425, New York, NY, USA, 2020. Association for Computing Machinery
33. S. Sullivant. Algebraic statistics, volume 194 of Graduate Studies in Mathematics. American Mathematical Society, Providence, RI, 2018
34. Telen, S., Van Barel, M., Verschelde, J.: A robust numerical path tracking algorithm for polynomial homotopy continuation. SIAM J. Sci. Comput. **42**(6), A3610–A3637 (2020)
35. Verschelde, J.: Algorithm 795: Phcpack: a general-purpose solver for polynomial systems by homotopy continuation. ACM Transactions on Mathematical Software **25**(2), 251–276 (1999)