Kinematic synthesis over curves using cellular decompositions and Chebyshev interpolants

Silviana Amethyst[1], Jonathan D. Hauenstein[2], and Charles W. Wampler[2]

[1] Max Planck Institute of Molecular Cell Biology and Genetics, Dresden, Germany
amethyst@mpi-cbg.de
[2] University of Notre Dame, Notre Dame, IN, USA
{hauenstein,cwample1}@nd.edu

Abstract. An approach for kinematic synthesis of mechanisms is proposed when the constraints define an algebraic curve. This method first computes a numerical cellular decomposition of the real part of the curve. In such a decomposition, each edge is represented by an interior point and a homotopy that permits tracking along the edge. This numerical representation of each edge is converted to Chebyshev interpolants which facilitate efficient optimization of an analytic or non-analytic function. Illustrative examples along with synthesizing a four-bar linkage are provided which utilize `Bertini_real` to compute a numerical cellular decomposition and `Chebfun` to compute Chebyshev interpolants.

Keywords: kinematic synthesis, numerical cellular decomposition, numerical algebraic geometry, Chebyshev interpolants

1 Introduction

Kinematic synthesis of a mechanism can be formulated mathematically as a system of basic constraints [9]. It often happens that constraining a synthesis problem so that there are only finitely many solutions is unproductive since many of the solutions turn out to be nonphysical (such as links having non-real lengths), useless (such as a linkage which visits the precision points in an unintended order or on two different branches of the coupler curve), or impractical (such as a linkage having poor force transmission so that it is susceptible to jamming or having links too long to fit within geometric constraints). In such circumstances, a useful tactic is to remove a constraint, thereby allowing the designer to search over the real part of a curve to find a desirable answer. One way to automate such a search is for the designer to specify an optimization criterion and it is often the case that the criterion is not algebraic. Even with a non-analytic objective function, we aim to find the global optimum in this search.

For a specific example, Section 5 considers a mechanism design problem wherein a curve of degree 72 in 10 variables describes all four-bars whose coupler motion interpolates one precision point and four precision poses, corresponding to Alt-type [1] and Burmester-type [6] constraints, respectively. A four-bar linkage is composed of two dyads (commonly referred to as an RR linkage) shown in red and green in Fig. 3. After building a Chebyshev representation of that degree 72 curve, a subsequent optimization step finds all four-bars on it that minimize the sum of the lengths of the four links in the two dyads. Although the sum of squares of lengths is algebraic and hence analytic, the sum of the lengths is not. Nonetheless, our approach finds all local minima and the global minimum. Moreover, the initial computations for representing the curve is independent of the optimization criterion, allowing the designer to quickly and reliably explore alternative objective functions, always obtaining the global minimum.

The approach couples two mathematical ideas to solve this problem. The first ingredient is a numerical cellular decomposition of the real part of the curve (implicit representation) [3,4,8], which can be computed using `Bertini_real` [4][3] A cellular decomposition is a description of the curve as a finite union of vertices and edges that have a smooth interior as illustrated in Fig. 1. Numerically, each edge is represented by one interior point together with a homotopy-based approach for tracking along the edge starting from this interior point. This representation of each edge can be used for deciding membership and sampling points.

The second ingredient is a Chebyshev interpolant (extrinsic representation), e.g., see [7], of each edge of the numerical cellular decomposition which permits optimization to be performed efficiently. For simplicity, we always consider edges to represent a bounded part of the curve, e.g., constrained to be inside of a sphere. We show how to compute an analytic reparameterization of each edge of a numerical cellular decomposition that works even when a vertex is singular. This reparameterization permits efficient approximation of each edge using Chebyshev interpolants. `Chebfun` [7][4] includes out-of-the-box methods for performing computations using Chebyshev interpolants, including optimization.

The structure of the remainder of the paper is as follows. Sections 2 and 3 describe numerical cellular decomposition (implicit representation) and Chebyshev interpolant (explicit representation), respectively. Section 4 provides two illustrative planar examples, then Section 5 applies the technique to a four-bar kinematic synthesis problem. A short conclusion is provided in Section 6.

2 Numerical cellular decomposition and analytic reparameterization

For a curve $C \subset \mathbb{C}^N$, a numerical cellular decomposition provides a representation of the real part $C \cap \mathbb{R}^N$ as a finite list of vertices and edges with [3,4,8] providing more details. Such a decomposition is computed with respect to a linear projection map $\pi : \mathbb{R}^N \to \mathbb{R}$ such that $C \cap \pi^{-1}(t)$ consists of finitely many points for every $t \in \mathbb{C}$. This property holds for a general linear projection map.

The vertices $V \subset C \cap \mathbb{R}^N$ in a numerical cellular decomposition contain the points in $C \cap \mathbb{R}^N$ that are singular, including isolated points, as well as the end-

[3] Available at http://bertinireal.com, which depends upon `Bertini` [2] that is available at http://bertini.nd.edu.

[4] Available at http://chebfun.org, which runs in standard `Matlab` that is available at http://matlab.com.

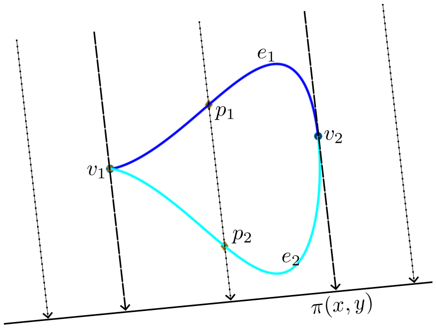


Fig. 1. A cellular decomposition of the real part of $x^6 - x^3 + y^2 = 0$ using projection $\pi(x,y) = 0.9x + 0.1y$. Critical point $v_1 = (0,0)$ is singular and invariant with respect to π, while $v_2 \approx (0.9916, 0.1557)$ depends on the choice of projection. Edges e_1, e_2 have smooth midpoints p_1, p_2. Slicing outside the curve gives no midpoint and thus no edge.

points of the edges. Each edge is represented numerically by three points: a "left" vertex $v_\ell \in V$, a "right" vertex $v_r \in V$, and an interior point $p \in C \cap \mathbb{R}^N$. Suppose that $f = \{f_1, \ldots, f_{N-1}\}$ is a polynomial system such that each irreducible component of C is an irreducible component of $\mathcal{V}(f) = \{x \in \mathbb{C}^N \mid f(x) = 0\}$ and has multiplicity one with respect to f. The vertices $v \in V$ satisfy $f = 0$ and the $N \times N$ Jacobian matrix of $\{f, \pi\}$ is rank deficient. Hence, the maximum number of vertices is

$$\deg C \cdot \left(1 - N + \sum_{i=1}^{N-1} \deg f_i\right).$$

Since this upper bound arises by considering the complex numbers whereas the vertices are real, the actual number of vertices depends upon the choice of π as illustrated in Ex. 1. If $L = \pi(v_\ell)$ and $R = \pi(v_r)$, then $L < \pi(p) < R$ and one has an implicit representation of the edge by having the ability to track along the edge via the homotopy

$$H(x;t) = \left[\, f(x), \;\; \pi(x) - t \,\right] = 0 \tag{1}$$

starting at $(x,t) = (p, \pi(p))$ and letting t vary in the interval $[L,R]$.

Example 1. Consider the degree six plane curve $x^6 - x^3 + y^2 = 0$, with the real part depicted in Fig. 1. It has one singular point at the origin. Using a sufficiently general projection $\pi(x,y) = 0.9x + 0.1y$, there is one critical point at $v_2 \approx (0.9916, 0.1557)$, arising due to tangency with π. Slicing between the two critical points gives two smooth midpoints p_1 and p_2, which connect to v_1, v_2 to form two edges e_1, e_2. Since this curve is compact, slicing outside the critical points gives no midpoints. If the curve had been non-compact, the unbounded branches would have generated additional midpoints and edges.

If one changes the projection to, say, $\tilde{\pi}(x,y) = 0.1x + 0.9y$, then one would obtain four vertices rather than the two depicted in Fig. 1.

Viewing each edge as the union of two arcs, one from the interior point p to the "left" vertex v_ℓ and the other from p to the "right" vertex v_r, each arc can be reparameterized to be analytic at the endpoints and diffeomorphic to the interval $[-1,1]$. Considering the "left" arc, let $L = \pi(v_\ell)$ and $P = \pi(p)$. Thus, the first reparameterization is to replace $t \in [L,P]$ in (1) by

$$t(\tau) = \tau \cdot P + (1-\tau) \cdot L = L + \tau \cdot (P - L)$$

where $\tau \in [0,1]$. Therefore, the "left" arc corresponds to $x(\tau)$ for $\tau \in [0,1]$ satisfying $x(1) = p$ and $H(x(\tau), t(\tau)) \equiv 0$ where H as in (1). By local uniformization [10, § 10.2], there exists a positive integer c_ℓ, called the cycle number, such that, for $\tau(s) = s^{c_\ell}$, the "left" arc $x(s) := x(\tau(s))$ is analytic for $s \in [0,1]$. The cycle number c_ℓ can be computed using endgames, e.g., see [10, § 10.2], which are typically already employed when computing a numerical cellular decomposition.

The net result is that one has that $x(s)$ for $s \in [0,1]$ satisfying $x(1) = p$ and

$$H_\ell(x,s) = \begin{bmatrix} f(x) \\ \pi(x) - (L + s^{c_\ell} \cdot (P-L)) \end{bmatrix} = 0. \tag{2}$$

One can apply a similar construction to the "right" arc yielding a cycle number c_r which is potentially different than c_ℓ.

Example 2. Both arcs of each edge of the curve $x^6 - x^3 + y^2 = 0$ as decomposed in Fig. 1 have cycle number 2.

3 Chebyshev interpolants

The outcome of Section 2 is a collection of implicitly defined arcs $x(s)$ for $s \in [0,1]$ which are analytic. Hence, one is able to perform computations on $x(s)$ by path tracking using the corresponding homotopy, e.g., (2) for the "left" arc. Converting this implicit representation of the arc into an explicit Chebyshev interpolant, e.g., see [7], enables one to compute and optimize general functions defined on the arc. That is, after first constructing a Chebyshev interpolant for each coordinate $x_i(s)$ of $x(s)$, one can efficiently cast a large class of functions of the form $g(x)$ as $g(x(s))$, thereby facilitating tasks such as evaluation, differentiation, and optimization. In particular, these and many other operations are all implemented in the `Matlab` package `Chebfun` [7].

For $K \geq 1$, the Chebyshev points of the second kind adjusted to $[0,1]$ are

$$s_j = (1 - \cos(j \cdot \pi/K))/2 \in [0,1] \quad \text{for} \quad j = 0, \ldots, K. \tag{3}$$

Given a function $g(s)$, let $g_j = g(s_j)$. There is a unique polynomial $q_K(x)$, called the Chebyshev interpolant, with $\deg q_K \leq K$ such that $q_K(s_j) = g_j$ for $j = 0, \ldots, K$. By selecting an appropriate value of K, one can control the error

$$\|q_K - g\|_\infty = \max_{0 \leq s \leq 1} |q_K(s) - f(s)|$$

thereby creating an explicit polynomial approximation of g for $s \in [0,1]$. In Chebfun, K is selected so that the error is approximately machine precision. When g is analytic, the error converges at least geometrically, but could still lead to using large values of K depending on properties of g.

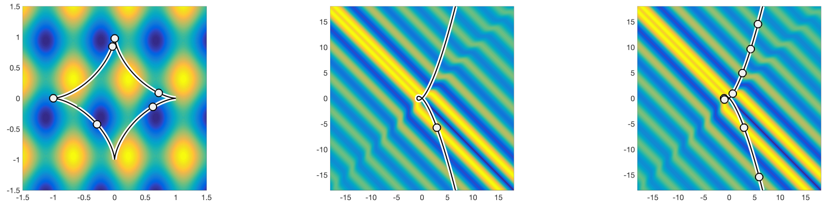
Returning to $x(s)$, each coordinate $x_i(s)$ is approximated using its own Chebyshev interpolant so the value of K can be different for each coordinate. Each $x(s_j)$ is computed via path tracking using the corresponding homotopy, e.g., (2) for the "left" arc. New points can be added to the approximation by tracking from neighboring samples. As more samples are added to an edge, neighbors become close and path tracking amounts to just a few Newton iterations.

4 Illustrative plane curves

The following two examples illustrate using the approach for optimizing analytic and non-analytic objective functions over the real part of an algebraic curve. Although these are not kinematics examples, they show the power of the approach for handling complicated optimization criteria over curves.

Consider minimizing the function $\Omega(x,y) = \sin(7x) + \sin(5y)$ over the real part of the "asteroid" curve defined by $f(x,y) = (x^2+y^2-1)^3 + 27x^2y^2 = 0$. This degree six curve has four singular points at $(\pm1,0)$ and $(0,\pm1)$ which are connected by four edges. Since both x and y as well as trigonometric functions of x and y are utilized, this is an analytic problem that is not algebraic. Nonetheless, once Chebyshev interpolants are computed for each arc of the "asteroid" curve, the minimization computation is performed with ease resulting in six local minima as shown in Fig. 2(a).

A second and more challenging example is to maximize the objective function $\Psi(x,y) = \text{besselj}(1, \text{abs}(x+y) + \text{erfc}(x-y) - 1)$ written using Matlab notation over the real part of the "alpha" curve defined by $f(x,y) = y^2 - x^2(x+1) = 0$. That is, besselj() is the Bessel function of the first kind, abs() is the absolute value, and erfc() is complementary error function. The cubic "alpha" curve has one singular point at the origin and is unbounded so it is decomposed inside of a



(a) (b) (c)

Fig. 2. (a) Six local minima for $\Omega(x,y)$ over the "asteroid" curve. (b) Global maximum and (c) several local maxima of $\Psi(x,y)$ over the "alpha" curve.

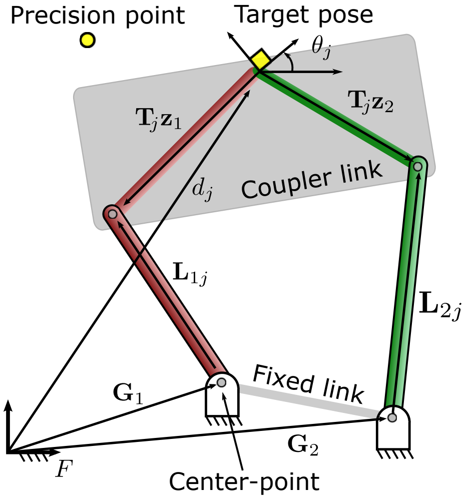


Fig. 3. A four-bar planar mechanism. The subscript j ranges over the precision points and poses.

suitable circle. Clearly, $\Psi(x, y)$ is non-analytic and defined in terms of integrals. Nonetheless, by using Chebyshev interpolants of the arcs of the "alpha" curve, numerical optimization can be easily performed yielding the global maximum as shown in Fig. 2(b) and several local maxima as shown in Fig. 2(c).

5 Synthesizing a planar four-bar mechanism

Returning to kinematics, we now show how this approach can be used to synthesize a four-bar mechanism as depicted in Fig. 3. The classic Burmester [6] problem asks to find all four-bars whose coupler link interpolates 5 precision poses. For general poses, this problem has up to 6 isolated solutions formed by choosing 2 of four possible circle-point/center-point pairs. If none of these yields a satisfactory design, one must change the precision poses and try again.

An alternative approach is to swap one precision pose for a precision point. This frees up one degree of freedom in the design, so that the 4-pose, 1-point specification yields a curve of possible four-bars. Following the formulation in [5] via isotropic coordinates, this $(4, 1)$ case of the Alt-Burmester family of design problems defines a curve of degree 72 in the 10 variables

$$(\mathbf{G}_1, \overline{\mathbf{G}}_1, \mathbf{z}_1, \overline{\mathbf{z}}_1, \mathbf{G}_2, \overline{\mathbf{G}}_2, \mathbf{z}_2, \overline{\mathbf{z}}_2, \mathbf{T}_3, \overline{\mathbf{T}}_3),$$

where $\mathbf{G}_1, \mathbf{G}_2$ give the center-point locations in the ground link, $\mathbf{z}_1, \mathbf{z}_2$ give the circle point locations in the coupler link, and $\mathbf{T}_3$ is the rotation of the coupler link when it is at the precision point, i.e., $\mathbf{T}_3 = e^{\theta_3\sqrt{-1}}$. (The subscript "3" here recognizes that the precision point is in row 3 of the precision data given

j	d_j	θ_j (deg)
1	$0.79 + 0.87\sqrt{-1}$	-11.76
2	$1.02 + 0.53\sqrt{-1}$	-33.56
3	$0.90 + 1.00\sqrt{-1}$	–
4	$1.28 + 0.33\sqrt{-1}$	-35.06
5	$1.30 + 0.71\sqrt{-1}$	-8.14

Table 1. Task specification with four precision poses and one precision point

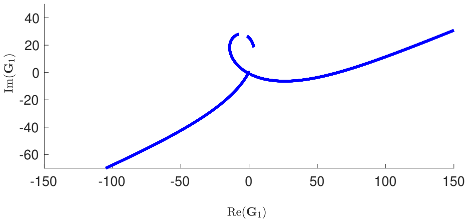


Fig. 4. Projection of a one-dimensional family of four-bar mechanisms onto $\mathbf{G}_1$. The "gaps" in this curve are where no physical mechanism can be constructed.

in Table 1.) Moreover, each is interpreted as a vector in the complex plane, so in the isotropic formulation, the variable list also includes the conjugate variable for each, e.g., $\overline{\mathbf{G}}_1$ is the conjugate of $\mathbf{G}_1$. A point on the curve in $\mathbb{C}^{10}$ is a "real" point corresponding to a physically realizable four-bar when each pair, $(\mathbf{G}_1, \overline{\mathbf{G}}_1)$, etc., consists of complex conjugates. The degree 72 curve in $\mathbb{C}^{10}$ yields a curve of degree 48 when projecting onto the first eight variables in accordance with the results presented in [5, Table 1].

Using the specifications in Table 1, the real part of the degree 72 curve of design alternatives is readily decomposed by `Bertini_real` [4]. In our particular run for this paper, the numerical cellular decomposition of this real curve which exists in a 10-dimensional space resulted in 1247 edges. (As described in Section 2, this number depends on the projection chosen for the decomposition). To help visualize it, Fig. 5 shows the corresponding center-point curve, which consists of the real and imaginary parts of $\mathbf{G}_1$. In accordance with [5, Table 2], this is a cubic curve. Nonetheless, one observes "gaps" in this image where no real physical design exists. Although the solution of the $(4, 1)$ problem includes real pairs $(\mathbf{G}_1, \overline{\mathbf{G}}_1)$ that would fill these gaps, at least one of the other pairs, i.e., $(\mathbf{z}_1, \overline{\mathbf{z}}_1)$, $(\mathbf{G}_2, \overline{\mathbf{G}}_2)$, $(\mathbf{z}_2, \overline{\mathbf{z}}_2)$, or $(\mathbf{T}_3, \overline{\mathbf{T}}_3)$, is non-real in the gaps.

After converting the implicit cellular decomposition of the curve to an extrinsic representation of each arc using Chebyshev interpolation, we are ready to find points on the design curve that optimize a secondary design criterion. For demonstration, we choose to minimize the sum of the lengths of the four links in the two dyads:

$$\Omega = |L_{11}| + |L_{21}| + |z_1| + |z_2|. \tag{4}$$

This will tend to select compact designs that take up less space and use less material to build. To this end, `Chebfun` determined there were 28 local minima over the real part of the curve restricted to the sphere centered at the origin of radius $2.744 \cdot 10^3$. Each physical design appears twice in this list since swapping the subscripts labeling the left and right dyads of the four-bar does not change the mechanism. Of the 14 distinct designs, 9 reach all precision specifications on the same branch of the coupler curve, while the other 5 are have branch defects. Two viable designs and one with a branch defect are shown in Fig. 5.

6 Conclusion

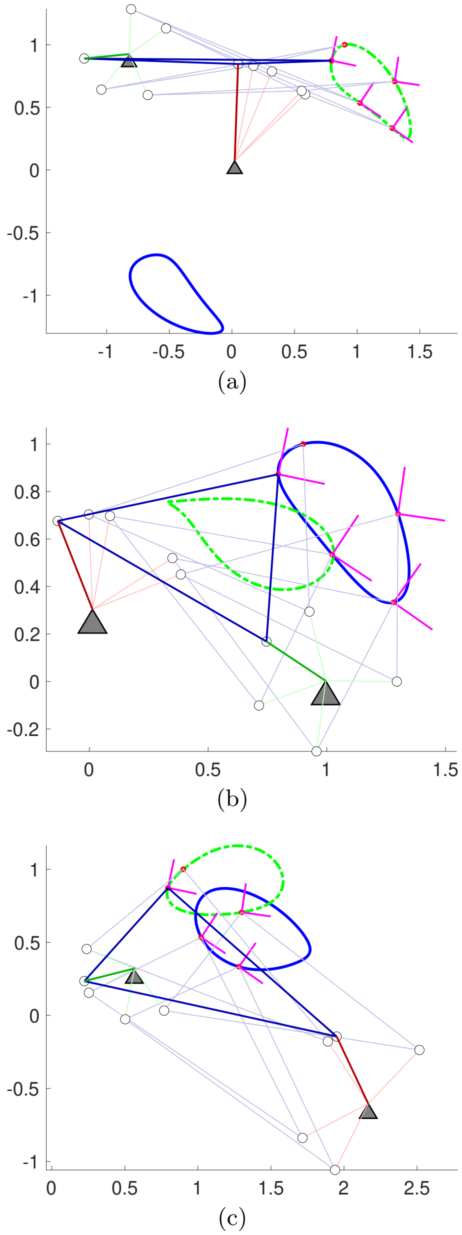
For kinematic synthesis problems where the basic constraints define an algebraic curve, we have shown how to use numerical cellular decomposition and Chebyshev interpolants to build an explicit representation of the real part of the curve that facilitates subsequent computations on the curve, such as optimizing an analytic or non-analytic function constrained to the curve. Cycle numbers obtained when computing a numerical cellular decomposition yield an analytic reparameterization so that each arc can be represented to machine precision as a Chebyshev interpolant even as it approaches a singularity. In our implementation, after using `Bertini_real` to compute a numerical cellular decomposition and `Chebfun` to compute a Chebyshev interpolant for each arc, all the functionality that `Chebfun` provides for functions on a real interval are available for use with this representation. A mechanical design problem with one degree of freedom involving a four-bar linkage demonstrates the practicality of this approach. Moreover, other mechanical design problems having one degree of freedom can be handled with this approach. Extending the approach to problems involving more degrees of freedom is left for future work.

Acknowledgments

The authors thank Nick Trefethen for helpful discussions regarding `Chebfun`. JDH was supported in part by NSF CCF 2331400, Robert and Sara Lumkins Collegiate Professorship, and Simons Foundation SFM-00005696. CWW was supported in part by Huisking Foundation, Inc Research Collegiate Professorship.

References

1. H. Alt. Über die erzeugung gegebener ebener kurven mit hilfe des gelenkvierecks. *ZAMM*, 3(1):13–19, 1923.
2. D. J. Bates, J. D. Hauenstein, A. J Sommese, and C. W. Wampler. Bertini: Software for Numerical Algebraic Geometry. Available at http://bertini.nd.edu.
3. G. M. Besana, S. Di Rocco, J. D. Hauenstein, A. J. Sommese, and C. W. Wampler. Cell decomposition of almost smooth real algebraic surfaces. *Num. Algor.*, 63(4):645–678, 2013.
4. D. A. Brake, D. J. Bates, W. Hao, J. D. Hauenstein, A. J. Sommese, and C. W. Wampler. Algorithm 976: Bertini_real: Numerical decomposition of real algebraic curves and surfaces. *ACM Trans. Math. Softw.*, 44(1):10, 2017.
5. D. A. Brake, J. D. Hauenstein, A. P. Murray, D. H. Myszka, and C. W. Wampler. The complete solution of Alt–Burmester synthesis problems for four-bar linkages. *J. Mech. Robot.*, 8(4):041018, 2016.
6. L. Burmester. *Lehrbuch der Kinematic.* Verlag Von Arthur Felix, Leipzig, Germany, 1886.
7. T. A. Driscoll, N. Hale, and L. N. Trefethen. Chebfun guide. https://www.chebfun.org/docs/guide/, 2024.



(a)

(b)

(c)

Fig. 5. Coupler curves for three local minima of Ω as in (4) for four-bar mechanisms satisfying the conditions of Table 1. (a) & (b) Coupler curves of two nondegenerate four-bar mechanisms having no branch defects. (c) Coupler curve of a four-bar mechanism which has a branch defect.

8. Y. Lu, D. Bates, A. Sommese, and C. Wampler. Finding all real points of a complex curve. *Contemp. Math.*, 448:183–205, 2007.
9. J. M. McCarthy and G. S. Soh. *Geometric Design of Linkages*, 2/e, Springer-Verlag, New York, 2011.
10. A. J. Sommese and C. W. Wampler, II. *The Numerical Solution of Systems of Polynomials Arising in Engineering and Science.* World Scientific Publishing Co. Pte. Ltd., Hackensack, NJ, 2005.