Applied Mathematics and Computation 332 (2018) 19–34

Contents lists available at ScienceDirect

Applied Mathematics and Computation

journal homepage: www.elsevier.com/locate/amc

Adaptive strategies for solving parameterized systems using homotopy continuation

Jonathan D. Hauenstein, Margaret H. Regan*

Department of Applied and Computational Mathematics and Statistics, University of Notre Dame, Notre Dame, IN, USA

A R T I C L E I N F O

Keywords:
Numerical algebraic geometry
Homotopy continuation
Parameter homotopy
Overdetermined system
Algebraic vision

A B S T R A C T

Three aspects of applying homotopy continuation, which is commonly used to solve parameterized systems of polynomial equations, are investigated. First, for parameterized systems which are homogeneous, we investigate options for performing computations on an adaptively chosen affine coordinate patch. Second, for parameterized systems which are overdetermined, we investigate options for adaptively selecting a well-constrained subsystem to restore numerical stability. Finally, since one is typically interested in only computing real solutions for parameterized problems which arise from applications, we investigate a scheme for heuristically identifying solution paths which appear to be ending at nonreal solutions and truncating them. We demonstrate these three aspects on two problems arising in computer vision.

1. Introduction

Parameterized systems of polynomial equations arise in many applications including computer vision [15,19,26], chemistry [1,22], and kinematics [12,31]. For a general setup, we assume that $F(x; p)$ is a system which is polynomial in the variables $x \in \mathbb{C}^N$ and analytic in the parameters $p \in \mathbb{C}^P$. Typically, one is interested in efficiently computing the solutions for many instances of the parameters. In computer vision, algorithms are used to solve the same system at many parameter values in order to employ the RANSAC algorithm [9,17].

When repeatedly solve many instances of a parameterized system, a typical framework is to utilize a two-phase approach. The *ab initio phase* performs preprocessing computations “offline” so that the “online” *solving phase* can more efficiently solve at the various instances. To demonstrate this framework, consider the so-called Gröbner trace approach, e.g., see [17]. In the *ab initio phase*, one first identifies algebraic manipulations of the equations at a randomly selected parameter value to discover how to reduce the corresponding system to a Gröbner basis from which the solutions can be efficiently extracted. The *solving phase* repeats these same manipulations for each given parameter instance. This approach can lead to efficient solvers in computer vision with many potential problems including the need for specialized software, a propagation of errors, instability, and expensive computations [17].

Rather than employ algebraic manipulation, we consider this framework using homotopy continuation in the form of a parameter homotopy [23] (see also [4, Chapter 6]). To employ a parameter homotopy, the *ab initio phase* computes all solutions at a randomly selected parameter value, say p^*. The *solving phase* tracks the solution paths using homotopy continuation as p^* is deformed to the given parameter instance, say $\hat{p}$, shown in Fig. 1.

* Corresponding author.
E-mail address: mregan9@nd.edu (M.H. Regan).

https://doi.org/10.1016/j.amc.2018.03.028
0096-3003/

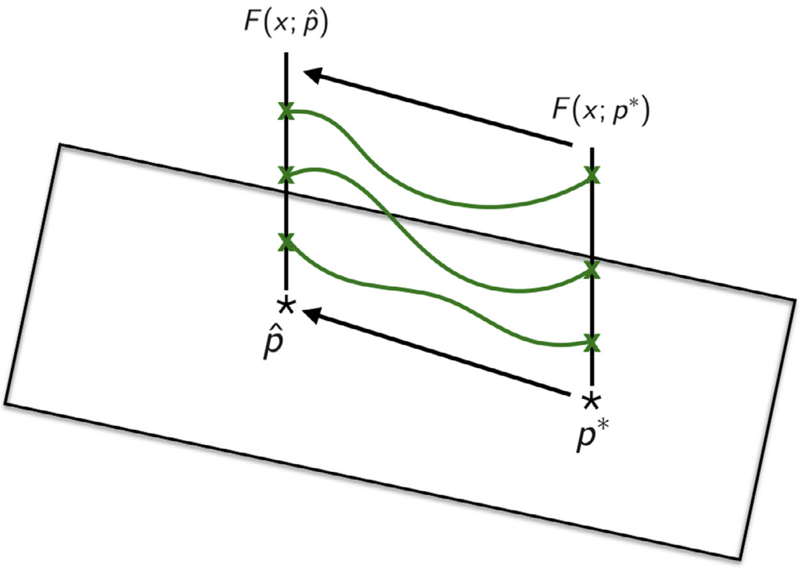

Fig. 1. Visualization of the parameter homotopy solving phase starting with the solutions of $F(x; p^*) = 0$ and ending with the solutions of $F(x; \hat{p}) = 0$.

Although parameter homotopies have been used to solve many instances of a parameterized system [2,3], there are three aspects of this computation that warrant further consideration. First, it is common for the parameterized system $F(x; p)$ to be homogeneous with respect to the variables x so that one treats the solutions as points in the projective space $\mathbb{P}^{N-1} = \mathbb{P}(\mathbb{C}^N)$. For example, 3×3 essential matrices in 3D image reconstruction are naturally considered as points in $\mathbb{P}^8 = \mathbb{P}(\mathbb{C}^{3\times3}) = \mathbb{P}(\mathbb{C}^9)$ [6]. For problems which are naturally formulated affinely, projective space is used to improve the solving process, particularly for solutions with large norm as well as handling nongeneric parameter values which have solutions at infinity [21]. Computationally, one natural approach for handling projective space is to utilize an affine coordinate patch. We will consider three strategies for selecting an affine patch: a fixed coordinate patch which is used throughout the computation [21] (Section 3.1), an adaptive orthogonal patch [25] (Section 3.2), and an adaptive coordinate-wise patch proposed in Section 3.3.

Second, it is common for parameterized problems to be overdetermined. For example, the set of essential matrices in computer vision consists of 3×3 matrices of rank 2 where the two nonzero singular values are equal. Since scaling is irrelevant, as mentioned above, this set is naturally defined on $\mathbb{P}^8$ by the vanishing of the determinant and the 9 cubic Demazure polynomials [6], namely

$$2EE^TE - \mathrm{trace}(\mathrm{EE}^\mathrm{T})\mathrm{E} = 0. \tag{1}$$

This system of 10 polynomials is overdetermined since it defines an irreducible set of codimension 3. Due to the numerical instability of solving overdetermined systems, we explore three techniques for reducing down to well-constrained systems: a fixed global randomization [27] (Section 4.1), an adaptive pseudoinverse randomization proposed in Section 4.2, and an adaptive leverage score randomization proposed in Section 4.3.

Third, when solving parameterized problems arising from applications, typically only the real solutions are of interest. That is, one need not compute the nonreal endpoints of solution paths defined by a homotopy. We propose a heuristic strategy in Section 6 for identifying and truncating paths which appear to be ending at nonreal solutions thereby saving computational time.

The remainder of the paper is as follows. Section 2 provides a short introduction to parameter homotopies and path tracking with more details provided in [4,28]. Section 3 compares the three strategies for affine patches while Section 4 compares the three strategies for randomizing down to a well-constrained subsystem. Section 5 presents pseudocode for the path tracking methods with Section 6 presenting our heuristic truncation scheme for nonreal solutions. We compare all of the approaches on two applications in computer vision in Section 7. The paper concludes in Section 8.

2. Parameter homotopies and path tracking

Throughout, we assume that the parameterized system $F(x; p)$ is polynomial in the variables $x \in \mathbb{C}^N$ and analytic in the parameters $p \in \mathbb{C}^P$. This setup ensures that the number of nonsingular isolated solutions of $F = 0$ has a generic behavior with respect to the parameter space $\mathbb{C}^P$, e.g., [28, Theorem 7.1.5]. This enables path tracking on the parameter space via a parameter homotopy [23] described below. We note that one could also consider positive-dimensional components using linear slicing and singular isolated solutions using deflation techniques, e.g., [11,20], to reduce to the nonsingular isolated case.