Journal of Computational and Applied Mathematics 258 (2014) 181–190

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

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

A bootstrapping approach for computing multiple solutions of differential equations

Wenrui Hao [c,*], Jonathan D. Hauenstein [b], Bei Hu [a], Andrew J. Sommese [a]

[a] Department of Applied and Computational Mathematics and Statistics, University of Notre Dame, Notre Dame, IN 46556, United States
[b] Department of Mathematics, North Carolina State University, Raleigh, NC 27695, United States
[c] Mathematical Biosciences Institute Ohio State University, Columbus, OH 43210, United States

ARTICLE INFO

Article history:
Received 24 November 2012
Received in revised form 10 June 2013

MSC:
65N55
65M22
65H10
65L10

Keywords:
Domain decomposition
Homotopy continuation
Differential equations
Multiple solutions
Numerical algebraic geometry

ABSTRACT

Discretizing systems of nonlinear algebraic differential equations yields polynomial systems. When using a fine discretization, the resulting polynomial system is often too large to solve using a direct solving approach. Our approach for solving such systems is to utilize a homotopy continuation based method arising from domain decomposition. This method solves polynomial systems arising from subdomains and then uses homotopy continuation to build solutions of the original polynomial system. We illustrate this approach on both one- and two-dimensional problems.

Published by Elsevier B.V.

1. Introduction

A common approach for approximating solutions to a system of nonlinear differential equations is to discretize and solve the resulting nonlinear system of equations. When the nonlinear equations are algebraic, the resulting system is a system of polynomials. Even though modern numerical codes for computing all solutions of a polynomial system can yield new solutions (see, e.g., [1]), the systems of polynomials (arising even from very sparse grids) are usually much too large for direct solution by these codes. The realization underlying this article is that domain decomposition gives guidance on how to "bootstrap" from the solutions of many small polynomial systems to often many solutions of a polynomial system arising from a fine discretization based on a realistic grid. Coefficient-parameter homotopies [2] are used to form solutions for the fine discretization from various solutions of the subsystems. In summary, we focus on computing solutions to the polynomial system resulting from a realistic discretization. However, since the development of a realistic discretization varies from system to system, for example, to encapsulate various properties of the differential system in the discretization, we do not discuss how one would obtain this for their particular system of differential equations. Some examples in the context of solving using homotopy approaches are a floating grid scheme for free-boundary problems [3–7] and a WENO scheme from hyperbolic conservation laws [8].

[*] Corresponding author.
E-mail addresses: hao.50@mbi.osu.edu (W. Hao), hauenstein@ncsu.edu (J.D. Hauenstein), b1hu@nd.edu (B. Hu), sommese@nd.edu (A.J. Sommese).
URLs: http://www.nd.edu/~whao (W. Hao), http://www.math.ncsu.edu/~jdhauens (J.D. Hauenstein), http://www.nd.edu/~b1hu (B. Hu), http://www.nd.edu/~sommese (A.J. Sommese).

0377-0427/$ – see front matter. Published by Elsevier B.V.
http://dx.doi.org/10.1016/j.cam.2013.09.007

The technique described in [9] builds solutions as follows. First, one solves a very coarse initial grid which one can consider arising from a finite-difference scheme or some other related method. After spurious solutions are filtered out, the mesh is refined in a smooth way. The resulting solutions are used as the starting points for a homotopy moving to a finer discretization. This process can be repeated until the set of real solutions has stabilized. If one now seeks more accurate solutions, the available solutions can be extrapolated to finer meshes and used as starting points for another homotopy, which is now over the real numbers.

Domain decomposition is a powerful tool for devising parallel methods to solve partial differential equations. The basic idea of domain decomposition is to first decompose the domain into subdomains. Then, each subdomain is solved independently in parallel. The solutions from the subdomains are then merged together to form solutions of the original problem. One issue with this approach is computing approximate values to the subdomain boundary points. Since this is a major difficulty, there is a rich literature (see, for example, [10–13] and the references therein) of schemes for approximating them using a time-dependent system of partial differential equations (PDEs), typically for systems of parabolic differential equations.

In this article, we introduce a new homotopy method based on domain decomposition which we call the *bootstrapping method*. This method computes multiple solutions to discretized systems of nonlinear algebraic differential equations and is naturally parallelizable. This method is more efficient than the traditional homotopy method for computing large-scale systems. We introduce this bootstrapping method for solving problems consisting of one and two space dimensions in Sections 2 and 4, respectively. Sections 3 and 5 examine the algorithm and provide numerical examples. A summary of the results is presented in Section 6.

2. One-dimensional bootstrapping

The fundamental idea of domain decomposition and the bootstrapping approach is that the polynomial systems resulting from discretizing on the whole domain and each subdomain are structurally the same and only differ in the values of parameters. In short, this means that coefficient-parameter continuation can be repeatedly used to solve such systems. To demonstrate this in the one-dimensional case, we will consider Laplace's equation. Suppose that $u : [0, 1] \to \mathbb{R}$ solves

$$\begin{cases} u_{xx} = f(u) & \text{on } (0, 1), \\ u(0) = c_0, \\ u(1) = c_1, \end{cases} \tag{2.1}$$

where $f(u)$ is a polynomial function of u, and $c_0, c_1 \in \mathbb{R}$. When $f(u) = u^p$, this equation is the steady-state equation for thermal runaway. Consider discretizing the system using $N + 1$ grid points located at $x_i = i/N$ for $i = 0, \ldots, N$ with a second-order central difference scheme to approximate u_{xx}. The resulting polynomial system is

$$F_H(u_0, \ldots, u_N) = \begin{bmatrix} f(u_i) - H^{-2}(u_{i+1} - 2u_i + u_{i-1}), \quad 1 \le i \le N-1 \\ u_0 - c_0 \\ u_N - c_1 \end{bmatrix}, \tag{2.2}$$

where $H = 1/N$, and u_i is considered to be an approximation of $u(x_i)$. If we are given that $u(x_i) = d_i$ and $u(x_{i+1}) = d_{i+1}$, consider discretizing each subinterval $[x_i, x_{i+1}]$ using $M + 1$ grid points located at $x_{i,j} = x_i + j/(NM)$ for $j = 0, \ldots, M$ with a second-order central difference scheme to approximate u_{xx}. The resulting polynomial systems is obtained from F_H by simply modifying the parameters H, c_0, and c_1; namely,

$$G_{i,h}(u_{i,0}, \ldots, u_{i,M}) = \begin{bmatrix} f(u_{i,j}) - h^{-2}(u_{i,j+1} - 2u_{i,j} + u_{i,j-1}), \quad 1 \le j \le M-1 \\ u_{i,0} - d_i \\ u_{i,M} - d_{i+1} \end{bmatrix}, \tag{2.3}$$

where $h = (NM)^{-1}$, and $u_{i,j}$ approximates $u(x_{i,j})$. The numbers d_i and d_{i+1} are called the boundary values of the subsystem $G_{i,h} = 0$. As mentioned above, obtaining the values of d_i and d_{i+1} is a major difficulty.

The goal of the bootstrapping approach is to compute solutions of $F_{H'} = 0$ for a sufficiently small value of H', i.e., a fine discretization using a large number of grid points, by solving $F_H = 0$ for large values of H, i.e., a coarse discretization using a small number of grid points, and building from solutions to the subsystems $G_{i,h} = 0$ using appropriately selected h and boundary values d_i and d_{i+1}.

We will use the values obtained from solving $F_H = 0$ as the boundary values of the subsystems. That is, the domain decomposition system under consideration is

$$F_{h,H}(\mathbf{u}) = \begin{bmatrix} f(u_{i,j}) - h^{-2}(u_{i,j+1} - 2u_{i,j} + u_{i,j-1}), & 0 \le i \le N-1, 1 \le j \le M-1 \\ f(u_{i,0}) - H^{-2}(u_{i+1,0} - 2u_{i,0} + u_{i-1,0}), & 1 \le i \le N-1 \\ u_{0,0} - c_0 & \\ u_{N,0} - c_1 & \end{bmatrix}, \tag{2.4}$$

where $H = N^{-1}$, $h = (NM)^{-1}$, $u_{i,M} = u_{i+1,0}$, and

$$\mathbf{u} = \left[u_{0,0}, \ldots, u_{0,M-1}, \ldots, u_{N-1,0}, \ldots, u_{N-1,M-1}, u_{N,0}\right].$$