Journal of Computational and Applied Mathematics 304 (2016) 138–159

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

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

Efficient numerical differentiation of implicitly-defined curves for sparse systems

David A. Brown *, David W. Zingg

University of Toronto, Institute for Aerospace Studies, Toronto, Ontario, M3H 5T6, Canada

ARTICLE INFO

Article history:
Received 15 November 2015
Received in revised form 25 February 2016

Keywords:
Numerical differentiation
Implicitly-defined curves
Homotopy
Continuation
High-order predictor
Sparse systems

ABSTRACT

A numerical technique is developed for the efficient numerical differentiation of regular implicitly-defined curves existing in high-dimensional real space such as those representing homotopies, where the system of equations which defines the curve implicitly is assumed to be sparse. The calculation is verified numerically through its application to the curve defined implicitly by a homotopy constructed based on a discretization of the equations governing compressible aerodynamic fluid flow. Consideration is given to computational cost, data storage, and accuracy. This method is applicable to any implicitly-defined curves or trajectories which can occur, for example, in dynamical systems analysis or control. Applications also exist in the area of homotopy continuation where implicitly-defined curves are approximately traced numerically. Such applications include the analysis of curve traceability and the construction of higher order predictors. The latter is investigated numerically and it is found that increasing the order of accuracy of the predictor can significantly improve the curve-tracing accuracy within a limited radius.

1. Introduction

Consider a curve segment defined implicitly by the system of equations

$$\mathcal{H}\left(\mathbf{q}\left(\lambda\right),\lambda\right)=\mathbf{0}, \tag{1}$$

$\mathcal{H}:\mathbb{R}^N\times\mathbb{R}\to\mathbb{R}^N$, $\mathbf{q}\in\mathbb{R}^N$, $\lambda\in\mathbb{R}$ on some interval $\lambda\in\Lambda$, $\Lambda\subset\mathbb{R}$. Without loss of generality, let $\Lambda=\{[0,1]\}$, and assume that the curve is oriented in the direction of decreasing λ. In this paper we assume that $\mathcal{H}$ is at least C^1 differentiable, invertible, and that the curve is regular on Λ. As such, the curve derivatives cannot vanish and no bifurcations are present. Hence, Eq. (1) is said to describe a regular homotopy [1] in $\mathbb{R}^N$. While homotopies including bifurcations have garnered much interest for the study of systems of equations of multiple solutions [2–8], and we do recognize the importance of including consideration for such points, they fall outside the scope of our particular applications and hence the scope of this paper.

It may be of practical interest to calculate higher derivatives of the curve, either for analysis or for application to numerical algorithms. While such calculations have been performed previously [9–11], the authors have not included special consideration for sparsity and the calculations can become prohibitively expensive if $\mathcal{H}$ is large and sparse. It is also important that the calculations be efficient if the calculation is to be used as part of a cost-competitive continuation algorithm. An example where this is important is homotopy continuation [1], which we have been developing as an efficient continuation strategy for solving the sparse algebraic systems of equations arising in computational fluid dynamics (CFD)

* Corresponding author.
E-mail address: davidbrown172@hotmail.com (D.A. Brown).

http://dx.doi.org/10.1016/j.cam.2016.03.002
0377-0427/

problems [12–14]. These systems of equations are sparse and it is not uncommon for the equations to number in the tens of millions, or even higher in some applications.

The calculations in this paper are developed in a Jacobian-free Krylov framework using the flexible generalized minimal residual (FGMRES) [15] method, though any linear solver suitable for solving linear systems of the form

$$\nabla_{\mathbf{q}}\mathcal{H}\mathbf{x} = \mathbf{b} \tag{2}$$

could be used, where $\mathbf{x}, \mathbf{b} \in \mathbb{R}^N$ and $\nabla\mathcal{H}$ indicates the Jacobian[1] of $\mathcal{H}$. The distinction that we are making with Eq. (2) is that the linear system is represented by a Jacobian matrix. Some linear solvers can make use of approximate Jacobian-vector products to avoid forming the Jacobian matrix explicitly.

2. Tangent vector

Consider the curve defined implicitly by a regular homotopy

$$\mathcal{H}\left(c\left(s\right)\right) = \mathbf{0}, \tag{3}$$

where the curve $c\left(s\right) = \left(\mathbf{q}\left(s\right); \lambda\left(s\right)\right)$, $c : \mathbb{R} \to \mathbb{R}^N \times \mathbb{R}$ has an arclength parametrization [16] defined implicitly by

$$\dot{c}\left(s\right) \cdot \dot{c}\left(s\right) = 1, \tag{4}$$

$s \in \mathcal{S}$, $\mathcal{S} \subset \mathbb{R}$, $\mathcal{S} = \{[0, s_{\text{tot}}]\}$. Differentiating both sides of Eq. (3) with respect to the arclength parameter s gives:

$$\nabla\mathcal{H}\left(c\left(s\right)\right)\dot{c}\left(s\right) = \mathbf{0}, \tag{5}$$

which can also be written:

$$\nabla_{\mathbf{q}}\mathcal{H}\left(c\left(s\right)\right)\dot{\mathbf{q}}\left(s\right) + \dot{\lambda}\left(s\right)\frac{\partial}{\partial\lambda}\mathcal{H}\left(c\left(s\right)\right) = \mathbf{0}, \tag{6}$$

or, after rearranging:

$$\nabla_{\mathbf{q}}\mathcal{H}\left(c\left(s\right)\right)\left[\frac{-1}{\dot{\lambda}\left(s\right)}\dot{\mathbf{q}}\left(s\right)\right] = \frac{\partial}{\partial\lambda}\mathcal{H}\left(c\left(s\right)\right). \tag{7}$$

Define the vector $\mathbf{z} \in \mathbb{R}^N$ such that

$$\nabla_{\mathbf{q}}\mathcal{H}\left(c\left(s\right)\right)\mathbf{z} = \frac{\partial}{\partial\lambda}\mathcal{H}\left(c\left(s\right)\right). \tag{8}$$

Then

$$\dot{\mathbf{q}} = -\dot{\lambda}\mathbf{z} \tag{9}$$

and

$$\dot{c}\left(s\right) \cdot \dot{c}\left(s\right) = \dot{\mathbf{q}}\left(s\right) \cdot \dot{\mathbf{q}}\left(s\right) + \dot{\lambda}\left(s\right)\dot{\lambda}\left(s\right) = \dot{\lambda}^2\left[\mathbf{z} \cdot \mathbf{z} + 1\right]. \tag{10}$$

Since $\dot{c}\left(s\right) \cdot \dot{c}\left(s\right) = 1$, Eq. (10) can be used to obtain an equation for $\dot{\lambda}\left(s\right)$:

$$\dot{\lambda}\left(s\right) = \frac{-1}{\sqrt{\mathbf{z} \cdot \mathbf{z} + 1}}, \tag{11}$$

where the negative sign has been included to force a negative orientation for $\dot{\lambda}\left(s\right)$, which is the convention that we have adopted. Substituting this back into Eq. (7) gives the expression for $\dot{\mathbf{q}}\left(s\right)$:

$$\dot{\mathbf{q}}\left(s\right) = -\dot{\lambda}\mathbf{z}. \tag{12}$$

The tangent vector can thus be calculated from Eqs. (11) and (12), where $\mathbf{z}$ is defined by Eq. (8) and requires the solution to a sparse linear system of equations.

3. Curvature vector

As with the tangent vector, the curvature vector will depend on the parametrization. Carrying over from the tangent calculation, an arclength parametrization is assumed. The derivation begins by differentiating both sides of Eq. (5), which gives

$$\nabla\mathcal{H}\left(c\left(s\right)\right)\ddot{c}\left(s\right) + \nabla^2\mathcal{H}\left(c\left(s\right)\right)\left[\dot{c}\left(s\right), \dot{c}\left(s\right)\right] = \mathbf{0}. \tag{13}$$

[1] When ∇ appears without subscript, differentiation is performed with respect to all variables including λ. The notation $\nabla_{\mathbf{q}}$ means that differentiation is with respect to the vector $\mathbf{q}$ only.