Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational and Applied Mathematics 201 (2007) 284–299
JOURNAL OF COMPUTATIONAL AND APPLIED MATHEMATICS
www.elsevier.com/locate/cam

BiCGStab, VPAStab and an adaptation to mildly nonlinear systems

P.R. Graves-Morris

Department of Computing, University of Bradford, Bradford BD7 1DP, UK

Received 16 September 2005; received in revised form 6 January 2006

Abstract

The key equations of BiCGStab are summarised to show its connections with Padé and vector-Padé approximation. These considerations lead naturally to stabilised vector-Padé approximation of a vector-valued function (VPAStab), and an algorithm for the acceleration of convergence of a linearly generated sequence of vectors. A generalisation of this algorithm for the acceleration of convergence of a nonlinearly generated system is proposed here, and comparative numerical results are given.

MSC: 41A21; 65B05; 65D15; 65H10

Keywords: BiCGStab; Vector-Padé approximation; Nonlinear system; Bratu equation; Burghers' equation

1. Introduction

A popular method for the solution of large, sparse linear systems is Van der Vorst's BiCGStab [28,29], the stabilised biconjugate gradient method. In order to justify the results and explain the links between the topics of this paper, we begin with the relevant facts about BiCGStab and the scalar-product class of methods of vector-Padé approximation. The first principal aim of this paper is to introduce, in context, stabilised vector-Padé approximation as a new method for approximation of a vector-valued function $\mathbf{f}(\lambda)$ in a suitable neighbourhood of $\lambda = 1$. The other principal aim is to introduce a new algorithm (closely related to BiCGStab and VPAStab) for the solution of a mildly nonlinear system of equations.

There are many methods for the solution of general, large sparse systems of linear equations, especially GMRes [24,23] which has several adaptations for nonlinear systems, see e.g., [5,21,8,18,19]. BiCGStab is a fast iterative method for linear systems having well-known modifications that give it good reliability. This paper includes a new algorithm that largely retains the advantages of BiCGStab for the case of mildly nonlinear systems, such as often arise at each time-step in the numerical solution of time-dependent PDEs.

In Section 2, the principal equations of BiCG [20,7] and BiCGStab [28,29] are summarised to facilitate comparisons with vector-Padé approximation. A connection between BiCG and Padé approximation was first pointed out by Gragg [11], and there are links with the topological ε-algorithm [2].

In Section 3, the methods of vector-valued Padé approximation (in the scalar product class) are described in terms of the construction of a sequence of vector-valued rational fractions whose MacLaurin series agree, as far as possible, with

E-mail address: p.r.graves-morris@bradford.ac.uk.

0377-0427/$ - see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.cam.2006.02.022

those of a given vector of functions. The rational fractions constituting each vector-valued Padé approximant (VPA) are required to have a common denominator polynomial and therefore common poles [1,31].

In Section 4, VPAStab, standing for stabilised vector-Padé approximation, is introduced as an algorithm that is intended to combine the advantages of BiCGStab with the methods of vector-Padé approximation for the acceleration of convergence of a sequence of vectors [14]. It is then extended, in a natural way, to the approximation of a vector-valued function $\mathbf{f}(\lambda)$ defined by a MacLaurin series whose coefficients are generated by a linear one-point iteration formula.

In Section 5, it is pointed out that there is a natural extension of the previous formalism to the case of acceleration of convergence of a sequence generated by a one-point iteration formula, and an example is given to show in detail how the method is applied to the solution of a nonlinear PDE.

In Section 6, the performance of the method is explicitly compared with that of the restricted minimal residuals (RMR) method, and implicitly with other standard methods [8], on two nonlinear benchmark PDEs.

To make the main themes as clear as possible, exceptional cases are mostly ignored. For examples, exact degree degeneracy and null pivots cause algorithms in the classes considered here to fail [17,13]. There are methods that circumvent these difficulties in practice. The most important extension for reliability is a look-ahead adaptation to cross small blocks of degenerate or nearly degenerate approximants, as is done with BiCGStabℓ, BiCGStab(ℓ) etc., [3,15,25–27] and reviewed in [6,17,29]. These modifications are also very important because unsymmetric matrices can have complex eigenvalues. Zhang's approach [32,22] to stabilisation is useful in this case, and also in the context of Section 4 [12].

2. Aspects of BiCG and BiCGStab

BiCGStab is an iterative method [28,29,6] for the solution of a large system of linear equations conventionally expressed as

$$A\mathbf{x} = \mathbf{b}, \tag{2.1}$$

with $\mathbf{x}, \mathbf{b} \in \mathbb{R}^d$ and $A \in \mathbb{R}^{d\times d}$ is nonsingular but not necessarily symmetric. The generalisation of BiCGStab to complex-valued systems is straightforward [17]. During each iteration of BiCGStab, an estimate $\mathbf{x}_i$ of the solution $\mathbf{x}$ is made and also of its residual

$$\mathbf{r}_i := \mathbf{b} - A\mathbf{x}_i, \quad i = 0, 1, 2, \ldots . \tag{2.2}$$

BiCGStab is an instance of a Lanczos-type product (or hybrid BiCG) method in which each residual is expressed as

$$\mathbf{r}_i = Q_i(A)P_i(A)\mathbf{r}_0 = U_i(A)\mathbf{r}_0 \in \mathscr{K}_{2i+1}(A, \mathbf{r}_0) \tag{2.3}$$

indicating that $\mathbf{r}_i$ lies in the Krylov subspace of dimension $2i+1$ germinated by $\mathbf{r}_0$ and generated by A. Here, we follow the notation of Van der Vorst [28] as closely as possible. $P_i(A)$ is a Lanczos polynomial and $Q_i(A)$ is called a stabilising polynomial; both are of degree i precisely, consequently, $\deg\{U_i(A)\} = 2i$. The estimates $\mathbf{x}_i = \mathbf{x}_0 + A^{-1}(I - U_i(A))\mathbf{r}_0$, and $\mathbf{x}_i - \mathbf{x}_0 \in \mathscr{K}_{2i}(A, \mathbf{b})$. For simplicity, we now assume that $\mathbf{x}_0 = \mathbf{0}$ and so $\mathbf{r}_0 = \mathbf{b}$; this situation can be achieved by making the indicated affine transformation on the $\mathbf{x}_i$ [18]. The Lanczos polynomial is chosen so that the orthogonality property

$$\mathbf{r}_i^{\mathrm{BiCG}} := P_i(A)\mathbf{b} \perp \mathscr{K}_{i-1}(A^{\mathrm{T}}, \mathbf{w}) \tag{2.4}$$

holds. Here, $\mathbf{w} \in \mathbb{R}^d$ is a more or less arbitrary vector that germinates the shadow space, and $\mathbf{w} = \mathbf{b}$ is a popular choice. Ultimately, when $i = d+1$, $\mathbf{r}_i^{\mathrm{BiCG}}$ is orthogonal to the whole space, and so, in principle, it must vanish. In practice, however, it is found that $\|\mathbf{r}_i^{\mathrm{BiCG}}\|$ mostly decreases as i increases, but often erratically. The superscript on $\mathbf{r}_i^{\mathrm{BiCG}}$ indicates that it is the residual calculated by the BiCG algorithm; it is not calculated by BiCGStab.

The factor $Q_i(A)$ in (2.3) is defined in factored form by

$$Q_i(A) = (I - \omega_1 A)(I - \omega_2 A)\ldots(I - \omega_i A), \tag{2.5}$$

with $\omega_i \in \mathbb{R}$. At stage i of the iteration, ω_i is chosen to minimise $\|\mathbf{r}_i\|_2$. This minimisation is remarkably effective at both smoothing the convergence of $\|\mathbf{r}_i\|$ compared to that of $\|\mathbf{r}_i^{\mathrm{BiCG}}\|$ and at reducing its size.