Numer. Math. Theor. Meth. Appl. Vol. **5**, No. 3, pp. 447-492
doi: 10.4208/nmtma.2012.m1035 August 2012

ML(n)BiCGStab: Reformulation, Analysis and Implementation[⋆]

Man-Chung Yeung[*]

Department of Mathematics, University of Wyoming, Laramie, WY 82071, USA.

Received 21 May 2011; Accepted (in revised version) 1 September 2011

Available online 3 July 2012

Abstract. With the aid of index functions, we re-derive the ML(n)BiCGStab algorithm in [Yeung and Chan, SIAM J. Sci. Comput., 21 (1999), pp. 1263-1290] systematically. There are n ways to define the ML(n)BiCGStab residual vector. Each definition leads to a different ML(n)BiCGStab algorithm. We demonstrate this by presenting a second algorithm which requires less storage. In theory, this second algorithm serves as a bridge that connects the Lanczos-based BiCGStab and the Arnoldi-based FOM while ML(n)BiCG is a bridge connecting BiCG and FOM. We also analyze the breakdown situation from the probabilistic point of view and summarize some useful properties of ML(n)BiCGStab. Implementation issues are also addressed.

AMS subject classifications: 65F10, 65F15, 65F25, 65F30

Key words: CGS, BiCGStab, ML(n)BiCGStab, multiple starting Lanczos, Krylov subspace, iterative methods, linear systems.

1. Introduction

Consider the solution of the linear system

$$\mathbf{A}\mathbf{x} = \mathbf{b}, \tag{1.1}$$

where $\mathbf{A} \in \mathbb{C}^{N\times N}$ and $\mathbf{b} \in \mathbb{C}^{N}$. If we express the BiCG [4, 15] residual as $\mathbf{r}_k^{BiCG} = p_k(\mathbf{A})\mathbf{r}_0$ in terms of a polynomial $p_k(\lambda)$ of degree k and the initial residual $\mathbf{r}_0$, the residual vector $\mathbf{r}_k$ of a Lanczos-type product method[†] based on BiCG is defined to be $\mathbf{r}_k = \phi_k(\mathbf{A})p_k(\mathbf{A})\mathbf{r}_0$, where $\phi_k(\lambda)$ is some polynomial of degree k with $\phi_k(0) = 1$. In CGS [28], $\phi_k = p_k$. Since, in every iteration, CGS searches for an approximate solution in a larger Krylov subspace, it

[⋆]Dedicated to the Memory of Prof. Gene Golub. This paper was presented in Gene Golub Memorial Conference, Feb. 29-Mar. 1, 2008, at University of Massachusetts. This research was supported by 2008 Flittie Sabbatical Augmentation Award, University of Wyoming.

[*]Corresponding author. *Email address:* myeung@uwyo.edu (M.-C. Yeung)

[†]For this type of Krylov subspace methods, one can consult [9]. They are called hybrid BiCG methods in [27].

often converges much faster than BiCG. However, CGS usually behaves irregularly due to a lack of a smoothing mechanism. In BiCGStab [31], the ϕ_k is

$$\phi_k(\lambda)=\begin{cases} 1 & \text{if } k=0,\\ (1-\omega_k\lambda)\phi_{k-1}(\lambda) & \text{if } k>0. \end{cases} \tag{1.2}$$

Here ω_k is a free parameter selected to minimize the 2-norm of $\mathbf{r}_k^{BiCGStab}$ in the kth iteration. As a result, BiCGStab is generally more stable and robust than CGS. BiCGStab has been extended to BiCGStab2 [7] and BiCGStab(l) [23, 27] through the use of minimizing polynomials of higher degree. In BiCGStab2, the ϕ_k is defined by the recursion

$$\phi_k(\lambda)=\begin{cases} 1 & \text{if } k=0,\\ (1-\omega_k\lambda)\phi_{k-1}(\lambda) & \text{if } k \text{ is odd},\\ ((\alpha_k\lambda+\beta_k)(1-\omega_{k-1}\lambda)+1-\beta_k)\phi_{k-2}(\lambda) & \text{if } k \text{ is even}. \end{cases}$$

The parameters are again chosen to minimize BiCGStab2 residuals. Likewise, BiCGStab(l) defines its ϕ_k as

$$\phi_k(\lambda)=\begin{cases} 1 & \text{if } k=0,\\ (1+\sum_{j=1}^{l}\alpha_j\lambda^j)\phi_{k-l}(\lambda) & \text{if } k \text{ is a multiple of } l, \end{cases}$$

where the parameters in the factor $1+\sum_{j=1}^{l}\alpha_j\lambda^j$ yields an l-dimensional minimization in every lth step. BiCGStab2 and BiCGStab(l) usually converge faster than BiCGStab because of smaller residuals in magnitude while avoiding near-breakdowns caused by a possibly too small ω_k. CGS, BiCGStab and BiCGStab2 have been summarized and generalized by GPBi-CG [40] where ϕ_k is

$$\phi_k(\lambda)=\begin{cases} 1 & \text{if } k=0,\\ 1-\omega_1\lambda & \text{if } k=1,\\ (1+\beta_k-\omega_k\lambda)\phi_{k-1}(\lambda)-\beta_k\phi_{k-2}(\lambda) & \text{if } k>1. \end{cases}$$

GPBi-CG will become CGS, BiCGStab or BiCGStab2 when the α,β,ω are appropriately chosen. For detailed descriptions of these and other product-type methods, one is referred to [6,8,20,22,32] and the references therein. Moreover, a history of product-type methods can be found in [10]. The history starts three decades ago with IDR [36] method which can be considered as the predecessor of CGS and BiCGStab [24]. Recently, IDR has been generalized to IDR(s) with a shadow space of higher dimension, see [24, 30, 34]. IDR(s) has close relations with ML(s)BiCGStab.

Generalizations of BiCGStab to methods based on the generalizations of BiCG have been made. For example, BL-BiCGStab [3] is a BiCGStab variant built on the BL-BiCG [16] for the solution of systems with multiple right-hand sides. ML(n)BiCGStab [39] is another BiCGStab variant built on ML(n)BiCG, a BiCG-like method derived from a variant of the band Lanczos process described in [1] with n left-starting vectors and a single right-starting vector.