Comm. Korean Math. Soc. 10 (1995), No. 4, pp. 981–995

CONVERGENCE AND BREAKDOWN STUDY OF KRYLOV METHODS FOR NONSYMMETRIC LINEAR SYSTEMS

JAE HEON YUN

ABSTRACT. This paper first establishes some conditions for preconditioner under which PGCR does not break down. Next, VPGCR algorithm whose preconditioners can be easily obtained is introduced and then its breakdown and convergence properties are discussed. Lastly, implementation details of VPGCR are described and then numerical results of VPGCR with a certain criterion guaranteeing no breakdown are compared with those of restarted GMRES.

1. Introduction

The classical Conjugate Gradient (CG) method of Hestenes and Stiefel [6] with some preconditioning technique is one of the most effective iterative methods for solving large sparse symmetric positive definite linear systems. However, this algorithm fails in general for nonsymmetric linear systems. In the last 15 years, a large number of generalizations of the CG method which are based on Krylov subspaces have been proposed for solving nonsymmetric linear systems [2, 3, 4, 10, 11, 12]. The Generalized Conjugate Residual (GCR) method [2], the BCG [3] method, the Generalized Minimum Residual (GMRES) method [10], the CGS method [11], and the Quasi-Minimal Residual (QMR) method [4] are typical examples of Krylov iterative methods for solving nonsymmetric linear systems.

Received July 24, 1995. Revised September 16, 1995.
1991 AMS Subject Classification: 65F10.
Key words: Krylov method, breakdown, preconditioner.
This paper was supported by RESEARCH FUND offered by ChungBuk National University Academic Research Foundation, 1994.

suffer from breakdowns (more precisely, division by 0). In finite precision arithmetic, such exact breakdowns are very unlikely; however, near-breakdowns may occur, leading to numerical instabilities in subsequent iterations. GMRES and QMR with look-ahead procedure do not break down, but GMRES has a stagnation problem (i.e., it does not converge to the exact solution until the last iteration step) and look-ahead QMR method suffers from an incurable breakdown (see [4] for details)

Throughout this paper, we consider the linear system $Ax = b$, where $A \in R^{n\times n}$ is a large sparse nonsymmetric nonsingular matrix, $x \in R^n$, and $b \in R^n$. Given a set of vectors $\{p_0, p_1, \dots, p_k\}$, let $\langle p_0, p_1, \dots, p_k\rangle$ denote the subspace spanned by $\{p_0, p_1, \dots, p_k\}$. For a given vector c_0, let the m-th *Krylov subspace* $K_m(A, c_0)$ denote the subspace $\langle c_0, Ac_0, \dots, A^{m-1}c_0\rangle$. $(\cdot,\cdot)$ denotes the Euclidean inner product on $R^n \times R^n$, and $\|\cdot\|$ denotes the Euclidean vector norm on R^n as well as the matrix norm associated with the Euclidean vector norm.

In [13], it was shown that the *preconditioned GCR* (PGCR) does not break down until convergence if a preconditioner satisfies a certain condition. This paper first establishes some new conditions for preconditioner under which PGCR does not break down, and then it is shown that these conditions for preconditioner are weaker than the condition stated in [13]. Since PGCR uses a fixed preconditioner during all iteration processes, it is very difficult to find the preconditioner guaranteeing no breakdown until convergence. To this end, VPGCR algorithm whose preconditioner varies every iteration and can be easily found is introduced, and its breakdown and convergence properties are discussed. Subsequently, details of implementation of VPGCR algorithm are given and then numerical results of VPGCR with a certain criterion guaranteeing no breakdown are compared with those of restarted GMRES. Finally, some concluding remarks are drawn.

2. The Preconditioned GCR (PGCR) algorithm

The GCR algorithm for solving a nonsymmetric linear system $Ax = b$ is described in detail in [2]. In this section, we consider the *preconditioned GCR* (PGCR) which can be obtained by applying GCR to the equivalent linear system $AM^{-1}(Mx) = b$, where M is a preconditioning matrix that approximates A and can be easily inverted. The PGCR algorithm with preconditioner M is as follows.

ALGORITHM 1 : PGCR

Choose x_0 and then compute $r_0 = b - Ax_0$
for $i = 0, 1, 2, \dots,$
  if $\| r_i \|$ satisfies a certain criterion, then stop
  Solve $Mw_i = r_i$
  Compute Aw_i
  if $i = 0$
    $p_i = w_i$
    $Ap_i = Aw_i$
  else
    $h_{ji} = -\frac{(Aw_i, Ap_j)}{(Ap_j, Ap_j)}, \ 0 \le j \le (i-1)$
    $p_i = w_i + \sum_{j=0}^{i-1} h_{ji} p_j$
    $Ap_i = Aw_i + \sum_{j=0}^{i-1} h_{ji} Ap_j$
  endif
  $\alpha_i = \frac{(r_i, Ap_i)}{(Ap_i, Ap_i)}$
  $x_{i+1} = x_i + \alpha_i p_i$
  $r_{i+1} = r_i - \alpha_i Ap_i$
end

Suppose that $\{p_i\}$, $\{w_i\}$, $\{r_i\}$, and $\{x_i\}$ are the sequences generated by the PGCR. Then, it is easy to show that they satisfy the following relations which are similar to those for GCR described in [2]:

(1.a) $(Ap_i, Ap_j) = 0$ if $i \neq j$, (1.b) $(r_i, Ap_j) = 0$ if $i > j$
(1.c) $(Ap_i, Ap_i) \le (Aw_i, Aw_i)$, (1.d) $(r_i, Ap_i) = (r_i, Aw_i)$
(1.e) $\langle p_0, \dots, p_i \rangle = \langle w_0, \dots, w_i \rangle = K_{i+1}(M^{-1}A, w_0)$
(1.f) x_{i+1} minimizes the residual norm $\| b - Ax \|$ over the affine space $x_0 + \langle p_0, \dots, p_i \rangle$.

From (1.c), it can be seen that $p_i \neq 0$ implies $w_i \neq 0$. Since M is assumed to be nonsingular, $w_i \neq 0$ is equivalent to $r_i \neq 0$. Hence, $p_i \neq 0$ implies $r_i \neq 0$. Theorems 2.1 and 2.2 that show a necessary and sufficient condition for PGCR not to break down are first introduced below, see [13] for their proofs.

THEOREM 2.1. *Suppose that $p_j \neq 0$ for all $j = 0, 1, \dots, i$. If $\alpha_i = 0$, then $p_{i+1} = 0$, i.e., PGCR breaks down at the $(i+1)$-th iteration.*

THEOREM 2.2. *Suppose that $p_j \neq 0$ for all $j = 0, 1, \ldots, i$. If $\alpha_i \neq 0$, then $p_{i+1} \neq 0$ unless $r_{i+1} = 0$, i.e., PGCR does not break down at the $(i+1)$-th iteration unless the exact solution is obtained at the $(i+1)$-th iteration.*

The hypothesis $p_j \neq 0$ for all $j = 0, 1, \ldots, i$ in Theorems 2.1 and 2.2 just means that PGCR does not break down until the i-th iteration. Now, we will consider some conditions for preconditioner M under which PGCR does not break down until convergence. From now on, it is assumed that $E = A - M$ for simplicity of exposition.

THEOREM 2.3. *If $r_i \neq 0$ and M is chosen such that $\| r_j - Aw_j \| = \| EM^{-1} r_j \| \leq \| r_j \|$ for all $j = 0, 1, \ldots, i$, then $\alpha_j \neq 0$ and $p_j \neq 0$ for all $j = 0, 1, \ldots, i$, and moreover $p_{i+1} \neq 0$ unless $r_{i+1} = 0$.*

PROOF. For $i = 0$, it is clear that $r_0 \neq 0$ implies $p_0 = w_0 \neq 0$. Since $\| r_0 - Aw_0 \| \leq \| r_0 \|$, $\| Aw_0 \|^2 \leq 2(r_0, Aw_0)$. Since $Aw_0 \neq 0$, $(r_0, Aw_0) = (r_0, Ap_0) > 0$ and thus $\alpha_0 \neq 0$. Moreover, if $r_1 \neq 0$, by Theorem 2.2 $p_1 \neq 0$. Suppose that this theorem holds for $i = k (\geq 0)$. Then, consider for $i = k + 1$. Since $r_{k+1} \neq 0$, by induction hypothesis $\alpha_j \neq 0$ for all $j = 0, 1, \ldots, k$ and $p_j \neq 0$ for all $j = 0, 1, \ldots, k + 1$. Since $\| r_{k+1} - Aw_{k+1} \| \leq \| r_{k+1} \|$, $\| Aw_{k+1} \|^2 \leq 2(r_{k+1}, Aw_{k+1})$. It follows that $(r_{k+1}, Aw_{k+1}) = (r_{k+1}, Ap_{k+1}) > 0$. Hence, $\alpha_{k+1} \neq 0$ and if $r_{k+2} \neq 0$, by Theorem 2.2 $p_{k+2} \neq 0$.

COROLLARY 2.4. *Suppose that $p_j \neq 0$ for all $j = 0, 1, \ldots, i$. If M is chosen such that $\| r_i - Aw_i \| = \| EM^{-1} r_i \| \leq \| r_i \|$, then $\alpha_i \neq 0$ and hence $p_{i+1} \neq 0$ unless $r_{i+1} = 0$, i.e., PGCR does not break down at the $(i+1)$-th iteration unless the exact solution is achieved at the $(i+1)$-th iteration.*

COROLLARY 2.5. *Suppose that $r_0 \neq 0$. If M is chosen such that $\| EM^{-1} \| \leq 1$, then PGCR does not break down until the exact solution is achieved.*

PROOF. Note that $\| EM^{-1} \| \leq 1$ implies $\| EM^{-1} y \| \leq \| y \|$ for an arbitrary vector y. It follows that $\| r_i - Aw_i \| \leq \| r_i \|$ for any nonzero residual vector r_i generated by PGCR. Since $p_0 \neq 0$ and $\| r_0 - Aw_0 \| \leq \| r_0 \|$, by Corollary 2.4 $\alpha_0 \neq 0$ and $p_1 \neq 0$ unless

$r_1 = 0$. Since $p_j \neq 0$ for $j = 0, 1$ and $\| r_1 - Aw_1 \| \leq \| r_1 \|$, by Corollary 2.4 $p_2 \neq 0$ unless $r_2 = 0$. Continuing in this manner, it can be seen that PGCR does not break down until the zero residual norm is obtained.

EXAMPLE 2.6. Consider $Ax = b$, where

$$A = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \quad \text{and} \quad b = \begin{pmatrix} 2 \\ 1 \end{pmatrix}.$$

Choose $M = \begin{pmatrix} 0.5 & 0.5 \\ -0.5 & 0.5 \end{pmatrix}$. Then

$$E = \begin{pmatrix} -0.5 & 0.5 \\ -0.5 & -0.5 \end{pmatrix} \quad \text{and} \quad EM^{-1} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}.$$

Since $\| EM^{-1} \| = 1$, Corollary 2.5 guarantees that PGCR does not break down until convergence. Let's try the PGCR for this example. For given $x_0 = (0,0)^T$, we have $r_0 = (2,1)^T$, $p_0 = M^{-1}r_0 = (1,3)^T$, and $\alpha_0 = \frac{1}{2}$. Using these results, $r_1 = \frac{1}{2}(1,3)^T$, $p_1 = \frac{1}{2}(-3,1)^T$, and $\alpha_1 = 1$. Hence, $x_2 = (-1,2)^T$, the exact solution of $Ax = b$, is obtained in the second iteration. However, GCR breaks down for this example unless the exact solution is chosen to be x_0.

We showed in [13] that *if* $\| EA^{-1} \| \leq \frac{1}{2}$, *then PGCR does not break down until the exact solution is obtained.* Lemma 2.7 and Example 2.8 given below show that $\| EM^{-1} \| \leq 1$ in Corollary 2.5 is **weaker** condition than $\| EA^{-1} \| \leq \frac{1}{2}$ in [13].

LEMMA 2.7. *If* $\| EA^{-1} \| < 1$, *then*

$$\| EM^{-1} \| \leq \frac{\| EA^{-1} \|}{1 - \| EA^{-1} \|}. \tag{2}$$

PROOF. Notice that $AM^{-1} = I + EM^{-1}$. Since $\| EA^{-1} \| < 1$,

$$AM^{-1} = (I - EA^{-1})^{-1} = I + \sum_{j=1}^{\infty} (EA^{-1})^j.$$

Hence $EM^{-1} = \sum_{j=1}^{\infty}(EA^{-1})^j$. From this identity, the inequality (2) is obtained.

From this lemma, it can be easily seen that $\| EA^{-1} \| \le \frac{1}{2}$ implies $\| EM^{-1} \| \le 1$. The following example shows that $\| EM^{-1} \| \le 1$ does not imply $\| EA^{-1} \| \le \frac{1}{2}$.

EXAMPLE 2.8. Let $A = \begin{pmatrix} 0 & -1 \\ \frac{2}{3} & -\frac{1}{2} \end{pmatrix}$ and $M = \begin{pmatrix} 0 & -2 \\ 1 & 0 \end{pmatrix}$. Since $E = A - M$,

$$EA^{-1} = \begin{pmatrix} -1 & 0 \\ \frac{3}{4} & -\frac{1}{2} \end{pmatrix} \quad \text{and} \quad EM^{-1} = \begin{pmatrix} -\frac{1}{2} & 0 \\ \frac{1}{4} & -\frac{1}{3} \end{pmatrix}.$$

Since $\| EM^{-1} \|$ is equal to the maximum singular value of EM^{-1}, $\| EM^{-1} \| = \frac{\sqrt{109} + \sqrt{13}}{24} < 1$. However, $\| EA^{-1} \| \ge 1$, i.e, $\| EA^{-1} \| > \frac{1}{2}$.

The following theorem shows that the better M approximates A, the faster the residual norm converges to 0.

THEOREM 2.9. *If PGCR algorithm does not break down until the i-th iteration, then* $\| r_{i+1} \| \le \| r_i - Aw_i \| = \| EM^{-1} r_i \|$.

PROOF. Let P be an orthogonal projection of R^n onto $\langle Ap_0, \ldots, Ap_{i-1} \rangle$. From relations (1.a) and (1.b), it can be seen that $Ap_i = (I - P)Aw_i$ and $Pr_i = 0$. Using these properties, one obtains

$$\begin{aligned} \| r_{i+1} \| &= \| r_i - \alpha_i Ap_i \| \le \| r_i - Ap_i \| \\ &= \| (I - P)(r_i - Aw_i) \| \le \| r_i - Aw_i \| . \end{aligned}$$

This completes the proof.

There are some well-known techniques for finding a preconditioner M, see [1, 5, 7, 8] for details. However, none of these techniques provide any information about the norm of EM^{-1}. Notice that the problem of finding a preconditioner M such that $\| EM^{-1} r \| \le \| r \|$ for a given nonzero vector r is much easier than that of finding a preconditioner M such that $\| EM^{-1} \| \le 1$ or $\| EM^{-1} r \| \le \| r \|$ for all r belonging to a finite set of nonzero vectors. To this end, we consider in the next section a variant of PGCR algorithm whose preconditioner varies every iteration depending upon r_i, where r_i is a residual vector at the i-th iteration step.

3. GCR algorithm with a variable preconditioner

In Section 2, it was shown that the PGCR using a fixed preconditioner M during all iteration processes does not break down until convergence if $\| EM^{-1} \| \leq 1$. Unfortunately, there are no known methods for finding a preconditioner M such that $\| EM^{-1} \| \leq 1$. So, we need to consider a variant of PGCR whose preconditioner guaranteeing no break-down until convergence can be easily obtained. For this purpose, GCR algorithm with a variable preconditioner, called VPGCR algorithm, is considered in this section. The main motivation of VPGCR algorithm results from a flexible inner-outer preconditioned GMRES algorithm which is proposed recently by Saad [9] and uses a variable preconditioner. VPGCR algorithm which is exactly the same as PGCR algorithm except for using a variable preconditioner M_i at the i-th iteration step is given below.

ALGORITHM 2 : VPGCR

Choose x_0 and then compute $r_0 = b - Ax_0$
for $i = 0, 1, 2, \ldots,$
  if $\| r_i \|$ satisfies a certain criterion, then stop
  Solve $M_i w_i = r_i$
  Compute Aw_i
  if $i = 0$
    $p_i = w_i$
    $Ap_i = Aw_i$
  else
    $h_{ji} = -\frac{(Aw_i, Ap_j)}{(Ap_j, Ap_j)}, \; 0 \leq j \leq (i-1)$
    $p_i = w_i + \sum_{j=0}^{i-1} h_{ji} p_j$
    $Ap_i = Aw_i + \sum_{j=0}^{i-1} h_{ji} Ap_j$
  endif
  $\alpha_i = \frac{(r_i, Ap_i)}{(Ap_i, Ap_i)}$
  $x_{i+1} = x_i + \alpha_i p_i$
  $r_{i+1} = r_i - \alpha_i Ap_i$
end

It is assumed that the preconditioner M_i in VPGCR varies depending upon r_i. More precisely, it is assumed that if $r_{i+1} = r_i$, then $M_{i+1} = M_i$. For the sequences $\{p_i\}$, $\{w_i\}$, $\{r_i\}$, and $\{x_i\}$ generated by VPGCR, it is easy to see that properties (1.a) to (1.d) and property (1.f) in Section 2 are also satisfied. Since the preconditioner for VPGCR varies every iteration, property (1.e) in Section 2 should be modified to $\langle p_0, \ldots, p_i \rangle = \langle w_0, \ldots, w_i \rangle$. Theorem 3.1 in the below shows a sufficient condition for VPGCR to break down.

THEOREM 3.1. *Suppose that $p_j \neq 0$ for all $j = 0, 1, \ldots, i$. If $\alpha_i = 0$, then $p_{i+1} = 0$, i.e., VPGCR breaks down at the $(i+1)$-th iteration.*

PROOF. Since $\alpha_i = 0$ and $p_i \neq 0$, $r_{i+1} = r_i \neq 0$ and so $M_{i+1} = M_i$. It follows that $Aw_{i+1} = Aw_i$. Let P be an orthogonal projection of R^n onto $\langle Ap_0, \ldots, Ap_i \rangle$. Since $Aw_i \in \langle Ap_0, \ldots, Ap_i \rangle$, $Ap_{i+1} = (I - P)Aw_{i+1} = (I - P)Aw_i = 0$. Hence, $p_{i+1} = 0$.

The following example shows that Theorem 2.2 in Section 2 is no longer true for VPGCR algorithm.

EXAMPLE 3.2. Consider $Ax = b$, where

$$A = \begin{pmatrix} 0 & 1 \\ 2 & 0 \end{pmatrix} \quad \text{and} \quad b = \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

Let's try VPGCR for this example. For given $x_0 = (0,0)^T$, we have $r_0 = (1,1)^T$. If we choose $M_0 = I$, then $p_0 = (1,1)^T$, $\alpha_0 = \frac{3}{5}$, and $r_1 = \frac{1}{5}(2,-1)^T$. Now choose $M_1 = \begin{pmatrix} 0 & 2 \\ -1 & 0 \end{pmatrix}$. Then, $w_1 = M_1^{-1} r_1 = \frac{1}{5}(1,1)^T$ and $p_1 = (0,0)^T$. This example shows that VPGCR produces $p_1 = (0,0)^T$ even if $\alpha_0 \neq 0$ and $r_1 \neq (0,0)^T$.

We now consider a condition for preconditioner M_i under which VPG CR does not break down until convergence. Note that if $(r_k, Ap_k) > 0$, then $p_k \neq 0$ and $\alpha_k \neq 0$. Using this fact, the following theorem whose proof is analogous to that of Theorem 2.3 can be obtained.

THEOREM 3.3. *If $r_i \neq 0$ and M_j's are chosen such that $\| r_j - Aw_j \| = \| r_j - AM_j^{-1} r_j \| \leq \| r_j \|$ for all $j = 0, 1, \ldots, i$, then $p_j \neq 0$ and $\alpha_j \neq 0$ for all $j = 0, 1, \ldots, i$.*

From Theorems 3.1 and 3.3, the following corollary is immediately obtained.

COROLLARY 3.4. *Suppose that* $p_j \neq 0$ *for all* $j = 0, 1, \ldots, (i-1)$ *and* $\alpha_{i-1} \neq 0$. *If* $r_i \neq 0$ *and* M_i *is chosen such that* $\| r_i - Aw_i \| = \| r_i - AM_i^{-1} r_i \| \leq \| r_i \|$, *then* $p_i \neq 0$ *and* $\alpha_i \neq 0$, *i.e., VPGCR does not break down at the i-th iteration.*

Since Theorem 2.2 does not hold for VPGCR, one more assumption $\alpha_{i-1} \neq 0$ has been added to Corollary 3.4 as compared with Corollary 2.4. Both Theorem 3.3 and Corollary 3.4 assume that M_i is chosen such that $\| r_i - AM_i^{-1} r_i \| \leq \| r_i \|$. It is easy to see that Theorem 3.3 and Corollary 3.4 still hold under the following assumption which replaces the above assumption:

A *nonzero* vector w_i is chosen such that $\| r_i - Aw_i \| \leq \| r_i \|$.

This means that we do not have to find M_i explicitly such that $\| r_i - AM_i^{-1} r_i \| \leq \| r_i \|$ when implementing VPGCR algorithm to solve a practical problem. Instead, it is sufficient to choose a nonzero vector w_i directly such that $\| r_i - Aw_i \| \leq \| r_i \|$. Choosing such a nonzero vector w_i is equivalent to finding a nonzero approximate solution w_i to $Aw = r_i$ such that $\| r_i - Aw_i \| \leq \| r_i \|$. Any iterative method can be used to find such an approximate solution w_i. This approach reduces the complexity of VPGCR algorithm since $w_i = M_i^{-1} r_i$ can be obtained directly *without finding variable preconditioner* M_i *explicitly.* When using an iterative method to find w_i, the zero vector is used as an initial guess. The main reason for using the zero initial vector is that zero vector already satisfies the condition $\| r_i - Aw_i \| \leq \| r_i \|$ and hence a nonzero vector w_i satisfying this condition can be obtained in small number of iteration steps. An important result of VPGCR is described in the following theorem when GMRES method is used to find a nonzero vector w_i.

THEOREM 3.5. *Let* r_i *and* p_i *be nonzero vectors generated by VPGCR algorithm. Suppose that GMRES method with zero initial vector is used to find a nonzero approximate solution* w_i *to* $Aw = r_i$. *If* $\| r_i - Aw_i \| = \epsilon \| r_i \|$, *then* $\| P(Aw_i) \| \leq \epsilon \| Aw_i \|$, *and* $1 \leq \frac{\|Aw_i\|}{\|Ap_i\|} \leq \sqrt{2}$ *when* $\epsilon \leq \frac{\sqrt{2}}{2}$, *where* P *is an orthogonal projection of* R^n *onto* $\langle Ap_0, \ldots, Ap_{i-1} \rangle$ *and* $0 < \epsilon < 1$.

PROOF. By the minimal residual property of GMRES, it is easy to see that

$$(3) \qquad (r_i, Aw_i) = (Aw_i, Aw_i).$$

From relation $r_{i+1} = r_i - \alpha_i Ap_i$ and equation (3),

$$(4) \qquad \| r_{i+1} \|^2 = \| r_i \|^2 - \frac{\| Aw_i \|^2}{\| Ap_i \|^2} \| Aw_i \|^2 .$$

From equation (3), $\| r_i - Aw_i \|^2 = \| r_i \|^2 - \| Aw_i \|^2$ and hence

$$(5) \qquad \| Aw_i \| = \sqrt{1 - \epsilon^2} \| r_i \| .$$

Since $\| r_{i+1} \| \geq 0$, from equations (4) and (5)

$$(6) \qquad \frac{\| Ap_i \|}{\| r_i \|} \geq 1 - \epsilon^2.$$

Since $\| Ap_i \| \leq \| Aw_i \|$, from (5) and (6)

$$(7) \qquad (1 - \epsilon^2) \| r_i \| \leq \| Ap_i \| \leq \sqrt{1 - \epsilon^2} \| r_i \| .$$

From (5) and (7), one obtains

$$\begin{aligned} \| P(Aw_i) \|^2 &= \| Aw_i \|^2 - \| Ap_i \|^2 \\ &\leq (1 - \epsilon^2) \| r_i \|^2 - (1 - \epsilon^2)^2 \| r_i \|^2 \\ &= \epsilon^2 \| Aw_i \|^2 . \end{aligned}$$

It follows that $\| P(Aw_i) \| \leq \epsilon \| Aw_i \|$. From this inequality and (7), it is easy to see that $\| P(Aw_i) \| \leq \| Ap_i \|$ if $\epsilon \leq \frac{\sqrt{2}}{2}$. Therefore, $1 \leq \frac{\|Aw_i\|}{\|Ap_i\|} \leq \sqrt{2}$ when $\epsilon \leq \frac{\sqrt{2}}{2}$.

Theorem 3.5 implies that if ϵ is close to 0, then $\| P(Aw_i) \|$ is close to 0 and hence $\| Ap_i \|$ is close to $\| Aw_i \|$. Therefore, if ϵ is close to 0, then from equation (4) $\| r_{i+1} \| = \| r_i - \alpha_i Ap_i \|$ is close to $\| r_i - Aw_i \|$, i.e., the norm of r_{i+1} obtained using $p_i = w_i + \sum_{j=0}^{i-1} h_{ji}p_j$ is close to that of r_{i+1} obtained using $p_i = w_i$. This means that all computational steps of VPGCR for finding p_i and r_{i+1} from a nonzero vector w_i, computed by GMRES, are of little use if ϵ is close to 0. Notice that VPGCR can be thought of as a combination of GMRES and GCR when GMRES method is used to compute $w_i = M_i^{-1}r_i$. From equation (4), it can be also seen that the larger $\frac{\|Aw_i\|}{\|Ap_i\|}$ is, the smaller $\| r_{i+1} \|$ is. It follows that GCR part of VPGCR becomes useful computational steps when $\frac{\|Aw_i\|}{\|Ap_i\|}$ can be made large. Theorem 3.5 gives an idea of how an optimal number of ϵ may be chosen to make use of advantages of both GMRES (used for computing w_i) and GCR (used for computing p_i and r_{i+1}) methods. Numerical experiments in next section show that a good choice of ϵ may range from 0.8 to 0.9 (see Tables 1 and 2). Notice that the good choice of ϵ is greater than $\frac{\sqrt{2}}{2}$

4. Numerical results

In this section, we present numerical results for both VPGCR and GMRES(k) algorithms on the CRAY-C90 supercomputer at the Systems Engineering Research Institute (SERI) in KIST. All tests have been carried out using 64-bit arithmetics (i.e., single precision on the CRAY).

The test problem considered in this paper is to solve a block tridiagonal linear system $Ax = b$ whose coefficient matrix A is obtained from a discretization of the following type of PDE problem

$$-(u_{xx} + u_{yy}) + \gamma(u_x + u_y) = f \text{ on } \Omega$$
$$u|_{\partial\Omega} = 0$$

where $\Omega = [0, 1] \times [0, 1]$ and γ is a constant. In the discretization, we use the standard five point central finite difference approximation and the same step size h in both x and y directions. Then the matrix A is of the form

$$A = \begin{pmatrix} D & B & & & \\ C & D & B & & \\ & \ddots & \ddots & \ddots & \\ & & C & D & B \\ & & & C & D \end{pmatrix} \quad \text{and} \quad D = \begin{pmatrix} 4 & \alpha & & & \\ \beta & 4 & \alpha & & \\ & \ddots & \ddots & \ddots & \\ & & \beta & 4 & \alpha \\ & & & \beta & 4 \end{pmatrix},$$

where $\alpha = -1+\delta$, $\beta = -1-\delta$, $B = (-1+\delta)I$, $C = (-1-\delta)I$, and $\delta = \frac{\gamma h}{2}$. If the step size $h = \frac{1}{N+1}$, then the order n of matrix A is $n = N^2$. The right-hand side vector b is chosen so that $b = A[1,1,\ldots,1]^T$. The initial vector x_0 is set to $[2,2,\ldots,2]^T$ for both algorithms to make a fair comparison. The termination criterion (i.e., convergence criterion) used for both algorithms is $\frac{\|r_i\|}{\|b\|} < 10^{-8}$.

Test runs were made for $\gamma = 1$ and $\gamma = 50$. When testing VPGCR algorithm, a nonzero vector w_i satisfying a criterion $\| Aw - r_i \| \le \epsilon \| r_i \|$ is computed using GMRES(10) with zero initial vector, where $0 < \epsilon < 1$. For the case where such a nonzero w_i is found within the first one period of GMRES(10), all 10 steps of that period of GMRES(10) are executed. According to numerical experiments, this methodology significantly reduces the number of vectors of order n to be stored. To find an optimal range of ϵ which provides good performance, numerical experiments for VPGCR are carried out for 5 different numbers of ϵ : $\epsilon = 0.5,\ 0.6,\ 0.7,\ 0.8$, and 0.9.

TABLE 1: Numerical results for $\gamma = 1$

$\sqrt{n}$	Method	Iter	Spmv	Mem
50	GMRES(30)	316	316	30
	VPGCR($\epsilon = 0.9$)	16	169	42
	VPGCR($\epsilon = 0.8$)	16	169	42
	VPGCR($\epsilon = 0.7$)	16	179	42
	VPGCR($\epsilon = 0.6$)	16	194	42
	VPGCR($\epsilon = 0.5$)	15	191	40
50	GMRES(30)	587	587	30
	VPGCR($\epsilon = 0.9$)	21	231	52
	VPGCR($\epsilon = 0.8$)	22	238	54
	VPGCR($\epsilon = 0.7$)	20	253	50
	VPGCR($\epsilon = 0.6$)	19	272	48
	VPGCR($\epsilon = 0.5$)	15	302	40
50	GMRES(30)	1050	1050	30
	VPGCR($\epsilon = 0.9$)	30	324	70
	VPGCR($\epsilon = 0.8$)	27	344	64
	VPGCR($\epsilon = 0.7$)	25	385	60
	VPGCR($\epsilon = 0.6$)	21	438	52
	VPGCR($\epsilon = 0.5$)	16	485	42

Numerical results for $\gamma = 1$ and $\gamma = 50$ are listed in Tables 1 and 2, respectively. *Iter* refers to the number of iterations required to get an approximate solution which satisfies the convergence criterion mentioned above, *Spmv* denotes the number of sparse matrix times vector operations, and *Mem* denotes the number of vectors of order n to be stored. Notice that *Mem* does not count the number of temporary vectors of small size which is much less than n. Storages for A, x, and b are not considered for comparison since these are used in both GMRES(k) and VPGCR algorithms. VPGCR generally requires more vector updates and inner products than GMRES(k). Since vector update and inner product operations usually take much less execution time than sparse matrix times vector operations, we only consider *Spmv* to evaluate the performance of each algorithm. For all test problems, VPGCR performs best when $\epsilon = 0.9$ and VPGCR has less sparse matrix times vector operations than GMRES(30) (see Tables 1 and 2). As ϵ gets smaller, VPGCR requires less storages, but its performance gets worse. As compared with GMRES(30), the relative performance of VPGCR for $\gamma = 1$ is much better than that of VPGCR for $\gamma = 50$.

TABLE 2: Numerical results for $\gamma = 50$

$\sqrt{n}$	Method	Iter	Spmv	Mem
50	GMRES(30)	299	299	30
	VPGCR($\epsilon = 0.9$)	18	191	46
	VPGCR($\epsilon = 0.8$)	17	213	44
	VPGCR($\epsilon = 0.7$)	16	220	42
	VPGCR($\epsilon = 0.6$)	16	245	42
	VPGCR($\epsilon = 0.5$)	15	245	40
50	GMRES(30)	431	431	30
	VPGCR($\epsilon = 0.9$)	21	246	52
	VPGCR($\epsilon = 0.8$)	21	279	52
	VPGCR($\epsilon = 0.7$)	21	300	52
	VPGCR($\epsilon = 0.6$)	18	320	46
	VPGCR($\epsilon = 0.5$)	16	338	42
50	GMRES(30)	506	506	30
	VPGCR($\epsilon = 0.9$)	25	319	60
	VPGCR($\epsilon = 0.8$)	25	372	60
	VPGCR($\epsilon = 0.7$)	23	410	56
	VPGCR($\epsilon = 0.6$)	21	451	52
	VPGCR($\epsilon = 0.5$)	15	483	40

5. Concluding Remarks

Most of the existing preconditioned iterative methods use a fixed preconditioner which can be usually found using various incomplete factorization techniques, see [1] for details. Unfortunately, there is no known method which finds a preconditioner guaranteeing no breakdown. Also, computational steps for finding a preconditioner require expensive costs and contain very small portion which can be executed in parallel.

Since GMRES is used to find a nonzero vector w_i such that $\| Aw_i - r_i \| \leq \| r_i \|$, VPGCR does not break down until convergence and its preconditioning step contains a lot of portion which can be executed in parallel. VPGCR also has an advantage of being able to find $w_i = M_i^{-1} r_i$ directly *without finding variable preconditioner* M_i *explicitly*. However, VPGCR may have a stagnation problem since GMRES used to find a nonzero w_i may stagnate. When testing VPGCR, a criterion $\| Aw_i - r_i \| \leq \epsilon \| r_i \|$ is used for the purpose of finding an optimal number of ϵ for which VPGCR performs well without breakdown. According to Theorem 3.5 and numerical experiments carried out in this paper, it can be expected that an optimal number of ϵ for VPGCR may range from 0.8 to 0.9 even for other types of problems. Lastly, since VPGCR can be viewed as a combined iterative method which utilizes advantages that both GCR and GMRES have, we recommend the use of VPGCR with a suitably chosen ϵ for the problems that GMRES does not perform very well.

References

1. R. Barrett et al., *Templates for the solution of linear systems : Building blocks for iterative methods*, SIAM, Philadelphia, 1994.
2. S. C. Eisenstat, H. C. Elman, and M. H. Schultz, *Variational iterative methods for nonsymmetric systems of linear equations*, SIAM J. Numer. Anal. **20** (1983), 345-357.
3. R. Fletcher, *Conjugate gradient methods for indefinite systems*, Lecture Notes in Math. **506** (1976), 73-89.
4. R. Freund and N. Nachtigal, *QMR : A quasi-minimal residual method for non-Hermitian linear systems*, Numer. Math. **60** (1991), 315-339.
5. I. Gustafsson, *A class of first order factorizations*, BIT **18** (1978), 142-156.
6. M. R. Hestenes and E. Stiefel, *Methods of conjugate gradients for solving linear systems*, J. Res. Nat. Bur. Standards **49** (1952), 409-435.

7. T. Manteuffel, *An incomplete factorization technique for positive definite linear systems*, Math. Comp. **34** (1980), 473-497.
8. T. Meijerink and H. van der Vorst, *An iterative solution method for linear systems of which the coefficient matrix is a symmetric M-matrix*, Math. Comp. **31** (1977), 148-162.
9. Y. Saad, *A flexible inner-outer preconditioned GMRES algorithm*, SIAM J. Sci. Stat. Comput. **14** (1993), 461-469.
10. Y. Saad and M. H. Schultz, *GMRES : a generalized minimum residual algorithm for solving nonsymmetric linear systems*, SIAM J. Sci. Stat. Comput. **7** (1986), 856-869.
11. P. Sonneveld, *CGS, a fast Lanczos-type solver for nonsymmetric linear systems*, SIAM J. Sci. Stat. Comput. **10** (1989), 36-52.
12. D. Young and K. Jea, *Generalized conjugate-gradient acceleration of nonsymmetrizable iterative methods*, Lin. Alg. and Appl. **34** (1980), 159-194.
13. Jae H. Yun, *Efficient parallel iterative methods for solving large nonsymmetric linear systems*, Comm. Korean Math. Soc. **9** (1994), 449-465.

Department of Mathematics
College of Natural Sciences
Chungbuk National University
Cheongju, 360-763, Korea
e-mail: gmjae@cbucc.chungbuk.ac.kr