



The Loss of Orthogonality in the Gram-Schmidt Orthogonalization Process

L. GIRAUD AND J. LANGOU
CERFACS, 42 Avenue Gaspard Coriolis
31057 Toulouse Cedex 1, France
<giraud><langou>@cerfacs.fr

M. ROZLOZNIK
Institute of Computer Science
Academy of Sciences of the Czech Republic and Technical University of Liberec
Pod vodárenskou věží 2
CZ-182 07 Prague 8, Czech Republic
miro@cs.cas.cz

Abstract—In this paper, we study numerical behavior of several computational variants of the Gram-Schmidt orthogonalization process. We focus on the orthogonality of computed vectors which may be significantly lost in the classical or modified Gram-Schmidt algorithm, while the Gram-Schmidt algorithm with reorthogonalization has been shown to compute vectors which are orthogonal to machine precision level. The implications for practical implementation and its impact on the efficiency in the parallel computer environment are considered. © 2005 Elsevier Ltd. All rights reserved.

Keywords—Numerical linear algebra, QR factorization, Gram-Schmidt orthogonalization, Re-orthogonalization, Rounding error analysis.

1. INTRODUCTION

Scientific computing and mathematical models in engineering are becoming increasingly dependent upon development and implementation of efficient parallel algorithms on modern high performance computers. Numerical methods and in particular algorithms of numerical linear algebra represent the most widely used computational tools in such area. Matrix computations such as the solution of systems of linear equations, least squares problems, singular value decomposition, and algebraic eigenvalue problems, govern the performance of many applications on vector and parallel computers. In almost all of them, one can frequently meet as a fundamental subproblem the orthogonal basis problem, i.e., the problem to construct or to compute an orthogonal basis of some linear subspace or a space generated by column vectors of an associated rectangular matrix.

In this paper, we consider the Gram-Schmidt orthogonalization process, the most widely known and used representative of a broad class of orthogonalization techniques and strategies (for a deep

The work of the third author was supported by the project 1ET400300415 within the National Program of Research “Information Society” and by the Institutional Research Plan AV0Z10300504 “Computer Science for the Information Society: Models, Algorithms, Applications”.

survey, we refer to [1–3]). In particular, we consider its classical and modified variant together with their variants with reorthogonalization. We will examine the level of orthogonality among the computed basis vectors produced by these schemes in connection with some characteristics of the initial column matrix A such as its dimensions or condition number $\kappa(A)$. Then, we use these results in the context of Arnoldi process for constructing an orthogonal basis of a sequence of associated Krylov subspaces. Presented results will lead to important conclusions about parallel implementation and efficiency of computational variants of the Gram-Schmidt algorithm.

The organization of the paper is as follows. Section 2 briefly recalls the Gram-Schmidt algorithm for a rectangular matrix A and gives an overview of basic results on the orthogonality of computed vectors developed for its different variants. In particular, we focus on recent roundoff analysis of the Gram-Schmidt algorithm with reorthogonalization. In Section 3, we consider the Arnoldi process based on four different orthogonalization schemes, namely, the classical and modified Gram-Schmidt orthogonalizations and their variants with reorthogonalization. Theoretical results are illustrated by numerical experiments on a real-world problem from the Harwell-Boeing collection. Throughout this paper, $\|X\|$ denotes the 2-norm (spectral norm) of matrix X , $\sigma_{\min}(X)$ stands for its minimal singular value and $\|x\|$ denotes the Euclidean norm of vector x . The condition number of X is denoted by $\kappa(X)$ and it is defined as $\kappa(X) = \|X\|/\sigma_{\min}(X)$.

2. LOSS OF ORTHOGONALITY IN QR FACTORIZATION

Let $A = (a_1, \dots, a_n)$ be a real $m \times n$ matrix ($m \geq n$) with full column rank ($\text{rank}(A) = n$). The Gram-Schmidt (GS) orthogonalization process [4] produces an orthogonal basis $Q = (q_1, \dots, q_n)$ of $\text{span}(A)$, such that $A = QR$, where R is upper triangular matrix of order n . The orthogonal matrix Q is constructed successively column-by-column so that for each $j = 1, \dots, n$, we have $Q_j = (q_1, \dots, q_j)$ and $\text{span}(q_1, \dots, q_j) = \text{span}(a_1, \dots, a_j)$. For the purpose of QR factorization of a matrix, many orthogonalization algorithms and techniques have been proposed and are widely used, including those based on Householder transformations or Givens rotations (see, e.g., [1–3]). Also several computational variants of the Gram-Schmidt process have been proposed and analyzed. Considerably less attention, however, has been paid to their numerical stability. Indeed, their numerical behavior can significantly differ leading sometimes to a severe loss of orthogonality or even to the loss of linear independence of computed vectors.

One of the first methods for successive orthogonalization is the classical Gram-Schmidt algorithm (CGS) [1]. It was confirmed by many numerical experiments, that this technique may produce a set of vectors which is far from orthogonal and sometimes the orthogonality can be lost completely [5]. Nevertheless, despite its weakness, this technique is frequently considered and implemented, probably due to its simplicity and potential parallelism, which will be discussed later. The brief sketch of the classical Gram-Schmidt algorithm can be found in Table 1.

A simple change in the loop of the CGS scheme leads to the modified Gram-Schmidt algorithm (MGS) with better numerical properties which are also much better understood (see the MGS algorithm on Figure 1 and/or [1,3]). Indeed, Björck [5] and later Björck and Paige [6] have shown that at iteration step $j = 1, \dots, n$ the loss of orthogonality of vectors Q_j computed in the

Table 1. The classical and modified Gram-Schmidt algorithm.

Classical GS Algorithm	Modified GS Algorithm
for $j = 1, \dots, n$	for $j = 1, \dots, n$
$a_j^{(1)} = a_j$	$a_j^{(1)} = a_j$
for $k = 1, \dots, j - 1$	for $k = 1, \dots, j - 1$
$a_j^{(1)} = a_j^{(1)} - (a_j, q_k)q_k$	$a_j^{(1)} = a_j^{(1)} - (a_j^{(1)}, q_k)q_k$
end	end
$q_j = a_j^{(1)} / \ a_j^{(1)}\ $	$q_j = a_j^{(1)} / \ a_j^{(1)}\ $
end	end

modified Gram-Schmidt process can be bounded as

$$\|I - \bar{Q}_j^T \bar{Q}_j\| \leq \zeta_1(m, j) \varepsilon \kappa(A_j), \quad (2.1)$$

where $\zeta_1(m, j)$ is a low degree polynomial in m and j depending only on details of computer arithmetic, ε is the machine precision and $\kappa(A_j)$ is the condition number of the matrix $A_j = (a_1, \dots, a_j)$. The bound on the loss of orthogonality of computed vectors in (2.1) is proportional $\kappa(A_j)$. Actually, for ill-conditioned matrices, the computed vectors can be very far from orthogonal.

In several application areas, however, it is important to compute the vectors \bar{Q}_j so that their orthogonality is close to the machine precision level. As an example of such problem, we will consider the Arnoldi process for computing the basis of Krylov subspaces which is used for solving nonsymmetric eigenvalue problem. In the framework of the Arnoldi method, the orthogonality of the computed vectors is essential for obtaining an accurate projection onto the corresponding space [1,2]. In such cases, the orthogonal basis problem is solved usually with Gram-Schmidt process with reorthogonalization. Reorthogonalization could be in principle applied several times, but as we will show later, one reorthogonalization step is enough for preserving the orthogonality of computed vectors close to machine precision level. Therefore, we consider just algorithms where the orthogonalization of a current vector against the previously computed set is performed exactly twice.

At each orthogonalization step $j = 1, \dots, n$ we start with the vector a_j and we generate, successively, the vectors

$$a_j^{(1)} = (I - Q_{j-1} Q_{j-1}^T) a_j, \quad a_j^{(2)} = (I - Q_{j-1} Q_{j-1}^T) a_j^{(1)}, \quad (2.2)$$

where the new basis vector is given as $q_j = a_j^{(2)} / \|a_j^{(2)}\|$. The orthogonalization (2.2) is done with CGS, an analogous formula can be derived for MGS. This leads to the CGS and MGS algorithms with reorthogonalization, respectively (see Table 2), where we initially set $a_j^{(0)} = a_j$.

Table 2. The classical and modified Gram-Schmidt algorithms with reorthogonalization.

Classical GS with Reorthogonalization	Modified GS with Reorthogonalization
for $j = 1, \dots, n$	for $j = 1, \dots, n$
for $i = 1, 2$	for $i = 1, 2$
$a_j^{(i)} = a_j^{(i-1)}$	$a_j^{(i)} = a_j^{(i-1)}$
for $k = 1, \dots, j-1$	for $k = 1, \dots, j-1$
$a_j^{(i)} = a_j^{(i)} - (a_j^{(i-1)}, q_k) q_k$	$a_j^{(i)} = a_j^{(i)} - (a_j^{(i)}, q_k) q_k$
end	end
end	end
$q_j = a_j^{(i)} / \ a_j^{(i)}\ $	$q_j = a_j^{(i)} / \ a_j^{(i)}\ $
end	end

It is clear from (2.2) that in exact arithmetic, we have $Q_{j-1}^T a_j^{(1)} = 0$ and $a_j^{(2)} = a_j^{(1)}$. The situation in finite precision arithmetic is more complicated and it was up to now less understood. Theoretical analysis of the situation with two vectors was given by Parlett in [7]. Parlett—who attributes this result to Kahan—showed that for $n = 2$, two steps are enough for obtaining the orthogonality level prescribed by a small multiple of machine precision ε . The result of Parlett and Kahan can be generalized for the case of n vectors. It was shown in [8] that under assumption on numerical nonsingularity of the matrix A_j in the form $\zeta_2(m, j) \varepsilon \kappa(A_j) < 1$, the loss of orthogonality of vectors \bar{Q}_j computed by the CGS or MGS algorithm with reorthogonalization can be bounded as

$$\|I - \bar{Q}_j^T \bar{Q}_j\| \leq \zeta_3(m, j) \varepsilon. \quad (2.3)$$

The terms $\zeta_2(m, j)$ and $\zeta_3(m, j)$ are low degree polynomials in m and j . Indeed, this result shows that two steps are enough for ensuring the orthogonality on the level of a small multiple of machine precision, when we apply the CGS (or MGS) orthogonalization process on a set of numerically nonsingular vectors. The proof of this fact is based on the observation that even the norm of the computed projection $\|\bar{a}_j^{(1)}\|$ cannot be infinitely small and essentially it is bounded from below by the minimal singular value $\sigma_{\min}(A_j)$. This approach is similar to the analysis of Abdelmalek [9] which is based on the criterion expected to hold in most practical cases and which can be shown to hold using the results in [8]. All details can be found also in [10].

The operation count for the CGS or MGS algorithm with reorthogonalization is $2mn^2$ flops (twice as much as CGS or MGS). This is comparable to the operation count for the Householder orthogonalization process which requires $2(mn^2 - n^3/3)$ flops. However, it is clear that the CGS algorithm with reorthogonalization is a better candidate than MGS algorithm with reorthogonalization for parallel implementation (see the algorithms in Table 2). This aspect could not be neglected in certain computing environments.

Different behaviors of the CGS, MGS algorithms and their variants with reorthogonalization can be demonstrated by numerical example with the matrix FS1836 from the Harwell-Boeing collection. The dimension of the (square) matrix is $m = 183$, the condition number is $\kappa(A) = 1.7 \times 10^{11}$ with $\|A\| = 1.2 \times 10^9$. Our experiment was performed in MATLAB, where $\varepsilon = 2.2 \times 10^{-16}$. Figure 1 illustrates the loss of orthogonality of a computed set of vectors measured by Frobenius norm as $\|I - \bar{Q}_j^T \bar{Q}_j\|_F$ at each orthogonalization step $j = 1, \dots, m$. Dashed line and dotted lines represent the loss of orthogonality in the CGS and MGS algorithm, respectively. It is easy to see that after some initial phase there is a gradual loss of orthogonality of computed vectors in both algorithms leading to its complete loss—we even observe the loss of linear independence—for CGS, while the loss of orthogonality for MGS remains on the level given by the bound (2.1). Almost identical solid line and dotted-solid line in Figure 1 correspond to the loss of orthogonality

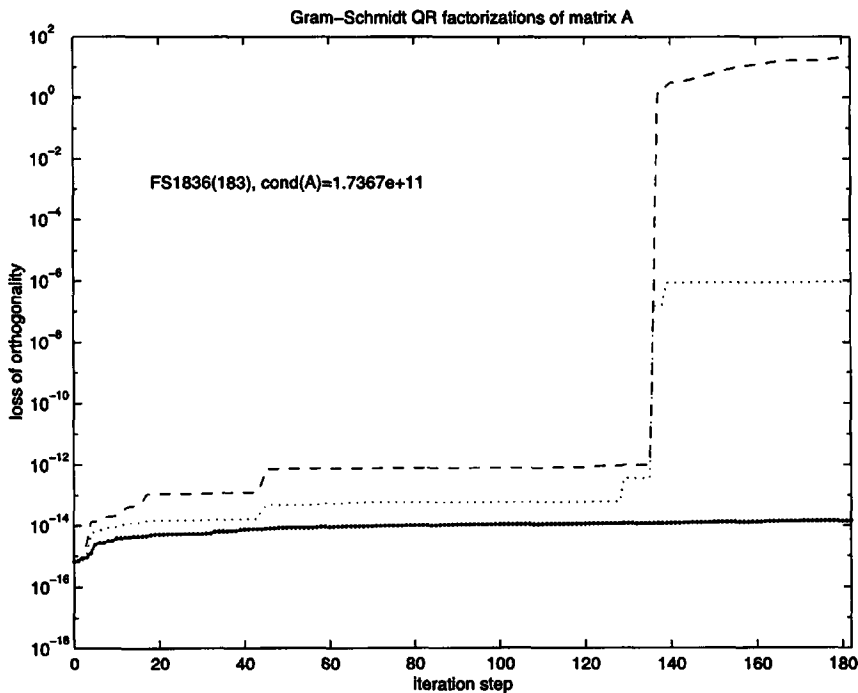


Figure 1. The loss of orthogonality in the QR factorization for different Gram-Schmidt orthogonalization variants: CGS algorithm (dashed line), MGS algorithm (dotted line), CGS algorithm with reorthogonalization (solid line), and MGS algorithm with reorthogonalization (dotted-solid line).

in the MGS and CGS algorithm with reorthogonalization. Clearly, the orthogonality of computed vectors remains close to the machine precision level and it agrees very well with the theoretical results of [8,10].

We point out here that the orthogonal basis problem is of the primary interest in this paper. The situation could be completely different in applications, where the orthogonality of computed vectors does not play a crucial role and where the MGS algorithm performs very well. Examples of this type are the solution of the least squares problems using MGS [1] or the MGS-GMRES method [11].

3. LOSS OF ORTHOGONALITY IN ARNOLDI PROCESS

The results on the general QR factorization of a rectangular matrix can be also used in the context of the Arnoldi process for constructing an orthogonal basis of the Krylov subspace. If we assume that A is a square nonsingular $m \times m$ matrix and v_1 an m -dimensional vector with $\|v_1\| = 1$, then let us define $K_j(A, v_1)$ the j^{th} Krylov subspace generated by the matrix A and vector v_1 by

$$K_j(A, v_1) = \text{span} \{v_1, Av_1, \dots, A^{j-1}v_1\}. \quad (3.4)$$

Classical way to construct a basis of the Krylov subspaces is the Arnoldi orthogonalization process. The orthonormal basis $V_{j+1} = [v_1, \dots, v_{j+1}]$ of the Krylov subspace $K_{j+1}(A, v_1)$, called the Arnoldi basis, is computed via the recurrences,

$$[v_1, AV_j] = V_{j+1} [e_1, H_{j+1,j}], \quad V_{j+1}^T V_{j+1} = I_{j+1}, \quad (3.5)$$

where $H_{j+1,j}$ is the upper Hessenberg $j+1$ by j matrix with the orthogonalization and normalization coefficients for entries. For more detailed description of the algorithm, we refer to [12].

In exact arithmetic, the Arnoldi vectors are orthogonal. However, in finite precision computation the orthogonality is lost and the behavior of Gram-Schmidt variants for computing the Arnoldi basis may differ significantly. Similarly to the previous section, now, we shall consider classical (CGS) and modified Gram-Schmidt (MGS) algorithms and their variants with reorthogonalization.

The Arnoldi process can be conceived as a column-oriented QR factorization (3.5). For the CGS variant, the loss of orthogonality in the CGS can be analogously expected to be much worse than in the MGS version. For the MGS variant, there is a very important relation between the loss of orthogonality among the computed vectors and the residual norm of an associated least squares problem. This relation has been described in [11]. It was shown that, the loss of orthogonality of vectors \bar{V}_{j+1} computed in the modified Gram-Schmidt implementation of the Arnoldi process is bounded by

$$\|I - \bar{V}_{j+1}^T \bar{V}_{j+1}\| \leq \zeta_5(m, j) \varepsilon \frac{\kappa(A)}{\min_y \|\bar{v}_1 - A\bar{V}_j y\|}, \quad (3.6)$$

where the quantity $\min_y \|\bar{v}_1 - A\bar{V}_j y\|$ is the norm of the residual of the associated least squares problem $\bar{v}_1 \sim A\bar{V}_j y$. Consequently, it can be shown that the Arnoldi vectors will loose their orthogonality completely only after the residual $\min_y \|\bar{v}_1 - A\bar{V}_j y\|$ is reduced close to the level, which is proportional to the machine precision multiplied by the condition number $\kappa(A)$. This suggests that for numerically nonsingular matrix A the loss of orthogonality in the modified Gram-Schmidt Arnoldi process occurs in a predictable way and it depends on the convergence of the residual $\min_y \|\bar{v}_1 - A\bar{V}_j y\|$. The details and connection to the GMRES method for the solution of linear systems can be found in [11].

If we want to keep the computed orthogonality as close to the machine precision as possible, we need to use the Gram-Schmidt orthogonalization scheme with reorthogonalization. Applying the results of [8] for the QR decomposition (3.5), it can be shown either for the CGS or MGS

Arnoldi process with reorthogonalization that the loss of orthogonality of the computed vectors \bar{V}_{j+1} is bounded independently of the system parameters and we have

$$\|I - \bar{V}_{j+1}^T \bar{V}_{j+1}\| \leq \zeta_6(m, j) \varepsilon. \tag{3.7}$$

where $\zeta_6(m, j)$ is a moderate degree polynomial in m and j independent of ε and of other system parameters.

The price for preserving the orthogonality near the machine precision is rather high, it is actually doubled in comparison to the simple CGS or MGS algorithm. In the parallel computational environment, however, the CGS algorithm with reorthogonalization can be the method of choice. Several experiments are reporting that even if performing twice as many operations as MGS, the CGS algorithm with reorthogonalization may be faster in some cases because it takes advantage of BLAS 2 or parallel facilities. In the Arnoldi context, we refer to [13] or [14] (see also experiments with the GCR method in [15]).

In the following, we illustrate the behavior of the Gram-Schmidt implementations of the Arnoldi process. We have used the same matrix FS1836 from the Harwell-Boeing collection with v_1 set as the normalized result of a multiplication $A(1, \dots, 1)^T$. In Figure 2, we have plotted the loss of orthogonality of computed vectors of different implementations of the Arnoldi method. Solid line represents the loss of orthogonality $\|I - \bar{V}_j^T \bar{V}_j\|_F$ for the Arnoldi using the CGS with reorthogonalization, almost identical dotted-solid line represents the loss of orthogonality for the MGS with reorthogonalization. Dashed line and dotted line and the loss of orthogonality for the CGS and MGS Arnoldi implementation, respectively. Numerical experiment clearly shows that their actual behavior corresponds to the theory developed in [8]. We note that the loss of orthogonality in CGS leads to a very poor limiting accuracy of the related GMRES method. In this case, the reorthogonalization is mandatory to obtain more accurate approximate solution. Concerning the MGS implementation of GMRES, the loss of orthogonality does not influence the convergence as predicted by the theory developed in [11].

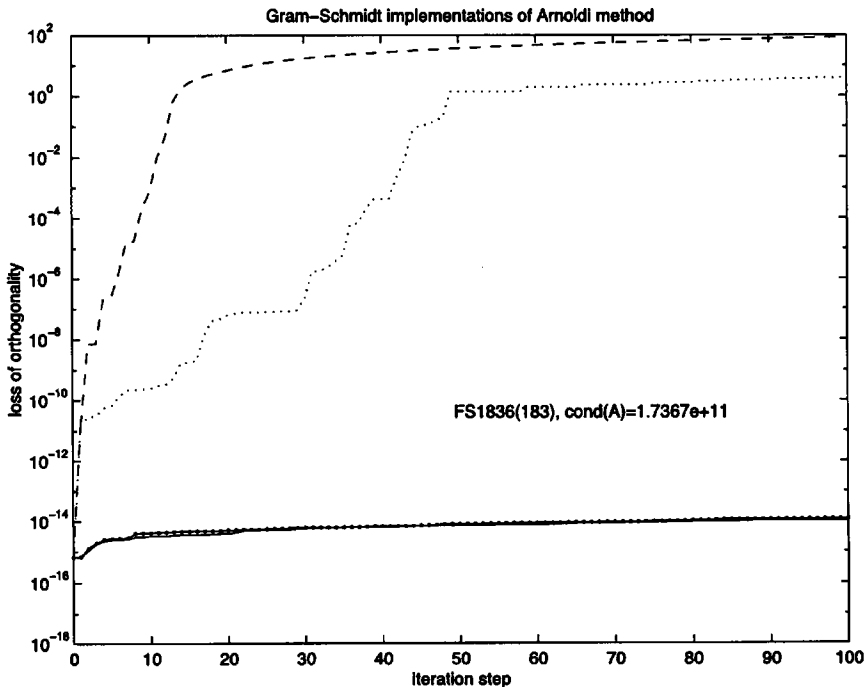


Figure 2. The loss of orthogonality in different Gram-Schmidt implementations of the Arnoldi process: CGS (dashed line), MGS (dotted line), CGS algorithm with reorthogonalization (solid line), and MGS algorithm with reorthogonalization (dotted-solid line).

4. CONCLUSION

In this paper, we have reviewed the most important theoretical results on the orthogonality of vectors computed by several computational variants of the Gram-Schmidt orthogonalization process. We have stressed that exactly two iteration-steps are already enough when full precision is wanted in the orthogonal basis problem and when the CGS or MGS algorithm is applied to (numerically) nonsingular initial set of column vectors. These results which fulfilled the theoretical gap in understanding the Gram-Schmidt process with reorthogonalization have been illustrated by a numerical example which comes from a real-world application.

REFERENCES

1. A. Björck, *Numerical Methods for Least Squares Problems*, SIAM, Philadelphia, PA, (1996).
2. G.H. Golub and C.F. Van Loan, *Matrix Computations*, Third Edition, John Hopkins University Press, Baltimore, MD, (1996).
3. N. Higham, *Accuracy and Stability of Numerical Algorithms*, SIAM, Philadelphia, PA, (1996).
4. E. Schmidt, Über die Auflösung linearer Gleichungen mit unendlich vielen Unbekannten, *Rend. Circ. Mat. Palermo. Ser. 1* **25**, 53–77, (1908).
5. A. Björck, Solving linear least squares problems by Gram-Schmidt orthogonalization, *BIT* **7**, 1–21, (1967).
6. A. Björck and C. Paige, Loss and recapture of orthogonality in the modified Gram-Schmidt algorithm, *SIAM J. Matrix Anal. Appl.* **13** (1), 176–190, (1992).
7. B.N. Parlett, *The Symmetric Eigenvalue Problem*, Prentice-Hall, Englewood Cliffs, NJ, (1980).
8. L. Giraud, J. Langou, M. Rozložník and J. van den Eshof, Rounding error analysis of the classical Gram-Schmidt orthogonalization process, *Numerische Mathematik* (to appear).
9. N. Abdelmalek, Round off error analysis for Gram-Schmidt method and solution of linear least squares problems, *BIT* **11**, 345–368, (1971).
10. J. Langou, Solving large linear systems with multiple right-hand sides, In *Ph.D. Dissertation TH/PA/03/24*, INSA, Toulouse, France, (June 2003); http://www.cerfacs.fr/algor/reports/Dissertations/TH_PA_03_24.pdf.
11. A. Greenbaum, M. Rozložník and Z. Strakoš, Numerical behaviour of the modified Gram-Schmidt GMRES implementation, *BIT* **37** (3), 706–719, (1997).
12. Y. Saad and M.H. Schultz, GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems, *SIAM J. Sci. Stat. Comput.* **7**, 856–869, (1986).
13. V. Frayssé, L. Giraud and H. Kharraz-Aroussi, On the influence of the orthogonalization scheme on the parallel performance of GMRES, In *EUROPAR'98 Parallel Processing*, pp. 751–762, Springer 1470, (1998).
14. R.B. Lehoucq and A.G. Salinger, Large-scale eigenvalue calculations for stability analysis of steady flows on massively parallel computers, *Int. J. Numerical Methods in Fluids* **36**, 309–327, (2001).
15. J. Frank and C. Vuik, Parallel implementation of a multiblock method with approximate subdomain solution, *Appl. Num. Math.* **30**, 403–423, (1999).
16. J.W. Daniel, W.B. Gragg, L. Kaufman and G.W. Stewart, Reorthogonalization and stable algorithms for updating the Gram-Schmidt QR factorization, *Math. Comp.* **30**, 772–795, (1976).
17. W. Hoffmann, Iterative algorithms for Gram-Schmidt orthogonalization, *Computing* **41**, 335–348, (1989).