An efficient approach for solving saddle point problems using block structure

Hiroto Tadano$^{1,*}$ and Shota Ishikawa$^2$

$^1$Center for Computational Sciences, University of Tsukuba
$^2$Graduate School of Science and Technology, University of Tsukuba
$tadano@cs.tsukuba.ac.jp$

Received: November 29, 2020; Accepted: April 4, 2021; Published: April 15, 2021

Abstract. This paper focuses on saddle point problems with a 2-by-2 block coefficient matrix. When the number of columns in the upper-right block and the number of rows in the lower-left block of the coefficient matrix is large, the convergence behavior of Krylov subspace methods for the saddle point problems tends to be poor even if the upper-left block is a well-conditioned matrix. In this paper, an efficient approach for solving the saddle point problems using block structure of the problems is proposed. The most time-consuming part of our proposed approach is the solution of a linear system with multiple right-hand sides. To solve the linear system with multiple right-hand sides efficiently, we propose to apply Block Krylov subspace methods to this linear system. Numerical experiments show that the proposed approach with Block Krylov subspace methods can solve the saddle point problems more efficiently than the conventional approach in terms of the number of iterations and the computation time.

Keywords: Saddle point problems, Linear systems with multiple right-hand sides, Block Krylov subspace methods

1. Introduction

This paper focuses on saddle point problems with a 2-by-2 block coefficient matrix of the form

$$\tilde{A}x \equiv \begin{bmatrix} A & B \\ C^T & O \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} f \\ g \end{bmatrix} = \tilde{b},$$

(1)

where $A \in \mathbb{R}^{n \times n}$ is a nonsingular and nonsymmetric sparse matrix, $B \in \mathbb{R}^{n \times m}$ and $C \in \mathbb{R}^{m \times n}$ are full rank matrices, $x, f \in \mathbb{R}^n$ and $y, g \in \mathbb{R}^m$. The saddle point problems appear in many applications in scientific fields such as meshless methods for partial differential equations, structural analysis, fluid dynamics, and so on [1].

If the size $N = n + m$ of the saddle point problems is small, direct methods such as the Gaussian elimination method and the LU factorization method are effective. However,
when \( N \) is large, it is difficult to apply the direct methods because of a large computational complexity and a large amount of memory are required. Hence, in this case, the saddle point problems need to be solved by iterative methods such as Krylov subspace methods.

As Krylov subspace methods for linear systems with a single right-hand side, the Conjugate Gradient (CG) method [2] for symmetric matrices, the Bi-Conjugate Gradient (BiCG) method [3], the BiCGSTAB method [4] and the Generalized Minimal Residual (GMRES) method [5] for nonsymmetric matrices have been proposed. The simple approach to solving the saddle point problems is to apply the Krylov subspace methods to the problems. However, as will be described later in Section 2, as the number \( m \) of columns in matrices \( B \) and \( C \) increases, the convergence behavior of the Krylov subspace methods for the saddle point problems tends to be poor even if \( A \) is a well-conditioned matrix.

This paper proposes to reconstruct the solution process of the saddle point problems by using the block structure of the problems in order to avoid the influence of the matrices \( B \) and \( C \). The most time-consuming part of the proposed approach is the solution of a linear system with the coefficient matrix \( \tilde{A} \) and the multiple right-hand sides. Therefore, the matrices \( B \) and \( C \) do not affect the coefficient matrix of the linear system that appears in the proposed approach. It is known that Block Krylov subspace methods are efficient methods for solving the linear system with multiple right-hand sides in terms of the computation time and the number of iterations. To solve the linear system with multiple right-hand sides efficiently, we also propose to adopt the Block Krylov subspace methods to this linear system.

The paper is organized as follows. Section 2 provides the results of preliminary experiments to investigate the convergence behavior of the Krylov subspace methods for the saddle point problems. In Section 3, an approach for solving the saddle point problems using the block structure is proposed. Section 4 shows the results of numerical experiments to evaluate the performance of the proposed approach. This paper is concluded in Section 5.

2. Convergence behavior of the Krylov subspace methods for the saddle point problems

In this section, the convergence behavior of the Krylov subspace methods for the saddle point problems is investigated through the preliminary experiments. As a matrix \( A \) of the upper-left block of the coefficient matrix \( \tilde{A} \), the matrix epb2 from the SuiteSparse Matrix Collection [6] is used. The size \( n \) and the number of nonzero elements of the matrix \( A \) are 25,228 and 175,027, respectively. The elements of the matrices \( B \) and \( C \) are given by a random number generator. As the Krylov subspace methods, the BiCGSTAB method [4], the GMRES method [5], and the restarted GMRES method [5] are used. The restart frequency of the restarted GMRES method is set as 100 and 200. The initial solution is set as the zero vector. The convergence criterion is \( \|\tilde{r}_k\|_2/\|\tilde{b}\|_2 \leq 10^{-15} \), where \( \tilde{r}_k \) is a \( k \)-th residual vector of the Krylov subspace methods. The iteration process is also stopped when the number of iterations achieves 5,000.

Computations are carried out in double precision arithmetic on a single node of the Cygnus Supercomputer operated at the Center for Computational Sciences, University of Tsukuba. The details of the experimental environment are shown in Table 1. Computations
Table 1: Experimental environment.

| Item       | Description                                      |
|------------|--------------------------------------------------|
| OS         | Cent OS ver. 7.7                                 |
| CPU        | Intel Xeon Gold 6126 2.6GHz (12 Cores) × 2       |
| Memory     | 192GiB (DDR4 2666MHz)                            |
| Compiler   | Intel Fortran ver. 19.0.5                        |
| Compile options | -qopenmp -axCORE-AVX512                      |

Figure 1: Estimated condition number of the matrices $A$ and $\tilde{A}$ as a function of the number $m$ of columns of $B$ and $C$.

Through the preliminary experiments, we investigate the changes in the number of iterations and the computation time of the Krylov subspace methods when the number $m$ of columns of the matrices $B$ and $C$ is varied. The number $m$ of columns of $B$ and $C$ is varied between 1 and 50.

Figure 1 shows the estimated condition number of the matrix $A$ and $\tilde{A}$ as a function of $m$. The estimated condition numbers for these matrices were computed using the MATLAB function condest. This computation was carried out on the Workstation with two sockets of Intel Xeon Gold 5220R 2.2GHz (24 cores) and 128GiB DDR4 memory. The estimated condition number of matrix $A$ was about $5.29 \times 10^3$. On the other hand, that of the coefficient matrix $\tilde{A}$ of (1) was larger than that of $A$ for $m = 1, 2, \ldots, 50$. Therefore, for this test matrix, the matrix $\tilde{A}$ is more ill-conditioned than $A$ in terms of the estimated condition number.

Figure 2 shows the number of iterations of the Krylov subspace methods as a function of $m$. In Fig. 2, only cases where the convergence criterion was satisfied are plotted. For the GMRES method, the convergence criterion was satisfied for all $m$. On the other hand, for the BiCGSTAB method and the GMRES(200) method, the convergence criterion was
Figure 2: Number of iterations of the Krylov subspace methods as a function of the number \( m \) of columns of \( B \) and \( C \).

satisfied for \( m \leq 8 \). The GMRES(100) method satisfied the convergence criterion for \( m \leq 3 \). In other cases, the number of iterations achieved the maximum number of iterations.

Figure 3 shows the computation time for solving the saddle point problems by the Krylov subspace methods as a function of \( m \). In Fig. 3, only cases where the convergence criterion was satisfied are plotted. For \( m \) in the case where the BiCGSTAB method and the restarted GMRES method satisfied the convergence criterion, the computation time of these methods was less than that of the GMRES method. As \( m \) increases, the computation time of the GMRES method was increased. This is because the number of iterations for solving the saddle point problems was increased.

Relative residual histories of the Krylov subspace methods for the saddle point problems are shown in Fig. 4. Figures 4(a) and 4(b) are the relative residual histories for the case of \( m = 1 \) and \( m = 25 \), respectively. As shown in Fig. 4 (a), when \( m = 1 \), all the methods satisfied the convergence criterion. On the other hand, only the GMRES method satisfied the convergence criterion when \( m = 25 \). The relative residual norm of the BiCGSTAB method oscillated and did not converge. That of the restarted GMRES method stagnated after the restart process.

Through the preliminary experiments, the following results were obtained.

- As the number \( m \) of columns of the matrices \( B \) and \( C \) increases, the convergence behavior of the Krylov subspace methods for the saddle point problems tended to be poor.
- The GMRES method was more robust than the BiCGSTAB method and the restarted GMRES method for the saddle point problems.
- When \( m = 1 \), the convergence behavior of the BiCGSTAB method for the saddle point problem was good.

Although the GMRES method is robust, as the number of iterations required to satisfy the convergence criterion increases, this method requires a large computational complexity and
a large amount of memory because the long-term recursions are used in the method. Therefore, it is desirable to be able to solve the saddle point problems by iterative methods with short-term recursions.

3. Reconstruction of the solution process for the saddle point problems using block structure

As shown in the previous section, when the number \( m \) of columns of the matrices \( B \) and \( C \) is few, the convergence behavior of the Krylov subspace methods for the saddle point problems tends to be good. In this case, the coefficient matrix \( \hat{A} \) of the saddle point problems is close to the matrix \( A \). Hence, if the solution of the saddle point problems can be generated by solving the linear systems with the coefficient matrix \( A \), it is expected that iterative methods with short-term recursions can be applied. In this section, an approach for solving the saddle
3.1. Derivation of the proposed approach

The saddle point problems (1) are expressed as follows:

\[
\begin{align*}
Ax + By &= f, \\
C^T x &= g.
\end{align*}
\]  

(2)  

(3)

By multiplying the matrix \(A^{-1}\) from the left side of both sides of Eq. (2), the following equation can be obtained.

\[x = A^{-1}f - A^{-1}By.\]  

(4)

Moreover, Eq. (4) can be transformed as follows:

\[
x = A^{-1}f - A^{-1}By
\Leftarrow C^T x = C^T A^{-1}f - \left(C^T A^{-1}B\right)y
\Leftarrow \left(C^T A^{-1}B\right)y = C^T A^{-1}f - g.
\]  

(5)

Since the matrix \(A\) is nonsingular and \(B\) and \(C\) are full rank matrices, the coefficient matrix \(C^T A^{-1}B \in \mathbb{R}^{m \times m}\) of the linear system (5) is nonsingular. Hence, the linear system (5) has a unique solution \(y\). The solution vector \(x\) can be computed by substituting the vector \(y\) for Eq. (4).

In this computation procedure, the matrix \(A^{-1}B \in \mathbb{R}^{n \times m}\) and the vector \(A^{-1}f \in \mathbb{R}^n\) are required for computing the solution vectors \(x\) and \(y\). Here, the matrix \(U \in \mathbb{R}^{n \times m}\) and the vector \(u \in \mathbb{R}^n\) are defined as follows:

\[
U \equiv A^{-1}B, \quad v \equiv A^{-1}f.
\]

Moreover, the matrices \(\hat{X}\) and \(\hat{B}\) are defined as follows:

\[
\hat{X} \equiv [U \ v] = [\hat{x}_1 \ \hat{x}_2 \ \ldots \ \hat{x}_m \ \hat{x}_{m+1}] \in \mathbb{R}^{n \times (m+1)},
\]

\[
\hat{B} \equiv [B \ f] = [\hat{b}_1 \ \hat{b}_2 \ \ldots \ \hat{b}_m \ \hat{b}_{m+1}] \in \mathbb{R}^{n \times (m+1)}.
\]

The matrix \(U\) and the vector \(v\) can be obtained by solving the linear system with multiple right-hand sides

\[
A\hat{X} = \hat{B}.
\]  

(6)

The solution vectors \(x\) and \(y\) are computed as follows:

\[
x = v - Uy,
\]

\[
y = \left(C^T U\right)^{-1} \left(C^T v - g\right).
\]

The algorithm of the proposed approach is summarized as Fig. 5.
Input: \( A \in \mathbb{R}^{n \times n}, B \in \mathbb{R}^{n \times m}, C \in \mathbb{R}^{n \times m}, f \in \mathbb{R}^{n}, g \in \mathbb{R}^{m} \)

Output: \( x \in \mathbb{R}^{n}, y \in \mathbb{R}^{m} \)

1. Set \( \hat{B} = [B \ f] \in \mathbb{R}^{n \times (m+1)} \)
2. Solve \( A\hat{X} = \hat{B} \) for \( \hat{X} = [\hat{x}_1 \ \hat{x}_2 \ \ldots \ \hat{x}_m \ \hat{x}_{m+1}] \)
3. Set \( U = [\hat{x}_1 \ \hat{x}_2 \ \ldots \ \hat{x}_m], v = \hat{x}_{m+1} \)
4. Compute \( S = C^T U \) and \( t = C^T v - g \)
5. Solve \( Sy = t \) for \( y \)
6. Compute \( x = v - Uy \)

Figure 5: Algorithm of the proposed approach for solving the saddle point problem using block structure.

### 3.2. Solution of the linear system with multiple right-hand sides

As described in the previous subsection, the saddle point problem is reduced to the linear system with multiple right-hand sides by means of the proposed approach. The main computation part of the proposed approach is the solution of the linear system with multiple right-hand sides (6). Hence, if the linear system (6) can be solved fast, the saddle point problems can also be solved fast.

As efficient methods for solving the linear system with multiple right-hand sides, Block Krylov subspace methods such as the Block Conjugate Gradient (Block CG) method [7] for symmetric matrices, and the Block Bi-Conjugate Gradient (Block BiCG) method [7], the Block BiCGSTAB method [8] and the Block GMRES method [9] for nonsymmetric matrices have been proposed. The Block Krylov subspace methods can solve the linear system with multiple right-hand sides simultaneously. Moreover, as shown in [10] and [11], the number of iterations of the Block Krylov subspace methods tends to decrease as the number of right-hand sides of the linear system increases. Therefore, in this paper, the Block Krylov subspace methods are adopted to solve the linear system with multiple right-hand sides (6).

### 4. Numerical experiments

Through numerical experiments, the performance of the proposed approach is evaluated. The performance of the proposed approach is compared with the conventional approach in terms of the following points.

- The number of iterations to solve the saddle point problems (1) (conventional approach) and that to solve the linear system with multiple right-hand sides (6) (proposed approach).
- Computation time to solve the saddle point problems (1).
Figure 6: Estimated condition number of the matrices $A$ and $\tilde{A}$ as a function of the number $m$ of columns of $B$ and $C$.

- Accuracy of the approximate solution of the saddle point problems (1).

The matrices epb2 and torso3 from the SuiteSparse Matrix Collection [6] are used as the test matrix $A$ in the coefficient matrix $\tilde{A}$ of the saddle point problems (1). The matrix epb2 was also used in the preliminary experiments in Section 2. The size $n$ and the number of nonzero elements of the matrix torso3 are 259, 156 and 4,429,042, respectively. The elements of the matrices $B$ and $C$ are given by a random number generator. Figure 6 shows the estimated condition number of the matrices $A$ and $\tilde{A}$ as a function of $m$. The estimated condition numbers for these matrices were computed using the MATLAB function condest. This computation was performed on the same Workstation as used in Section 2. For the matrix torso3, due to the memory limitation of the computational environment, the estimated condition number of $\tilde{A}$ could not be computed for $m > 100$.

To evaluate the accuracy of the obtained approximate solution of the saddle point problems (1), the exact solutions $x^\ast$ and $y^\ast$ of (1) are also given by a random number generator. By using the exact solutions $x^\ast$ and $y^\ast$, the right-hand vectors $f$ and $g$ of the saddle point problem are set as follows:

$$f = Ax^\ast + By^\ast,$$
$$g = C^Ty^\ast.$$ 

In the conventional approach, the saddle point problems (1) are solved by the BiCGSTAB method, the GMRES method and the restarted GMRES method. The restart frequency of the restarted GMRES method is 100 and 200. In the proposed approach, we compare the performance of the approach when the linear system (6) is solved by the Block GWBiCGSTABrQ method [10] and when the $(m + 1)$ linear systems $A\tilde{x}_j = \tilde{b}_j$ $(j = 1, 2, \ldots, m + 1)$ of (6) are sequentially solved by the BiCGSTAB method.

The Block GWBiCGSTABrQ method is one of the Block Krylov subspace methods for solving the linear systems with multiple right-hand sides. This method requires the
orthonormalization of the residual matrix at each iteration to improve the numerical stability of the method [12]. The Cholesky QR2 method [13] is adopted for the orthonormalization of the residual matrix. The maximum number of inner iterations of the Block GWBiCGSTABrQ method is fixed to 20.

The iteration is stopped when the relative residual norm of the Krylov subspace methods and the Block GWBiCGSTABrQ method achieved to $10^{-15}$. The relative residual norm of the Block GWBiCGSTABrQ method is computed as $\frac{\|R_k\|_F}{\|\hat{B}\|_F}$, where $R_k$ and $\| \cdot \|_F$ denote the $k$th residual matrix of the Block GWBiCGSTABrQ method and a Frobenius norm of a matrix, respectively. The maximum number of iterations of the iterative methods is 5,000. The initial solution of the Krylov subspace methods and the Block GW-BiCGSTABrQ method are the zero vector and the zero matrix, respectively. The experimental environment is the same as that used in Section 2.

### 4.1. Comparison of the number of iterations

Figure 7 shows the number of iterations of the iterative methods as a function of $m$. In Fig. 7, only the cases where the convergence criterion was satisfied are plotted. Here, we deal with the number of iterations when solving the linear system with multiple right-hand sides (6) by the Block GWBiCGSTABrQ method and the BiCGSTAB method (sequential solution) and that when solving the saddle point problems (1) by the Krylov subspace methods. The number of iterations of the proposed approach with BiCGSTAB denotes the total number of iterations required for solving $(m+1)$ linear systems $A\hat{x}_j = \hat{b}_j$ ($j = 1, 2, \ldots, m+1$) by the BiCGSTAB method.

For both matrices epb2 and torso3, the GMRES method could solve the saddle point problems for all $m$. On the other hand, the proposed approach with Block GWBiCGSTABrQ could not solve the linear systems with multiple right-hand sides (6) of 4 of 500 cases for epb2 and 3 of 500 cases for torso3. For these cases, the Block GWBiCGSTABrQ method broke down due to the orthonormalization of the residual matrix by the Cholesky QR2 method failed. The proposed approach with BiCGSTAB could not solve (6) of 3 of 500 cases for epb2 and 13 of 500 cases for torso3.

As shown in Fig. 7, the number of iterations of the GMRES method and the proposed approach with BiCGSTAB (sequential solution) increased as $m$ increased. On the other hand, that of the proposed approach with Block GWBiCGSTABrQ decreased as $m$ increased. This tendency in the proposed approach takes advantage of the property of the Block Krylov subspace methods.

Comparing Fig. 7(a) and 7(b), the number of iterations required to satisfy the convergence criterion for torso3 tended to be smaller than that for epb2. As shown in Fig. 6, the estimated condition number of $A$ for torso3 was also smaller than that for epb2. However, for $m \leq 100$, the estimated condition number of $\hat{A}$ for torso3 tended to be larger than that of epb2. Therefore, it is thought that the number of iterations required to obtain approximate solutions may be affected not only by the condition number of the coefficient matrix but also by other factors such as the eigenvalue distribution of the coefficient matrix.
Figure 7: Number of iterations as a function of \( m \) to solve (6) by the Block GW-BiCGSTABrQ method and the BiCGSTAB method (sequential solution) and that to solve (1) by the Krylov subspace methods.

### 4.2. Comparison of the computation time

Figure 8 shows the computation time for solving the saddle point problems (1). In Figs. 8(a) and (b), only the cases where the convergence criterion was satisfied are plotted.

As shown in Figs. 8(a) and (b), in most cases, the computation time of the proposed approach with Block GWBiCGSTAB was shorter than that of the GMRES method. In particular, for the matrix epb2, the computation speed of the proposed approach with Block GWBiCGSTABrQ was about 5.3 times faster than that of the GMRES method when \( m = 500 \). For the matrix torso3, the difference in the computation time between the proposed approach and the GMRES method was smaller than in the case for the matrix epb2. This is because the number of iterations of the GMRES method for torso3 was fewer than that for epb2.

Comparing the computation time of the proposed approach, the proposed approach with BiCGSTAB required a longer computation time than that with Block GWBiCGSTABrQ.
Therefore, in the proposed approach, the use of the Block Krylov subspace methods is essential in terms of computation time.

4.3. Comparison of the accuracy of the approximate solution of the saddle point problems

Figure 9 shows the relative error norm of the approximate solution of the saddle point problems generated by the conventional and proposed approach as a function of \( m \). The relative error norm \( \varepsilon \) is computed as follows:

\[
\varepsilon = \left\| \begin{bmatrix} x' \\ y' \end{bmatrix} - \begin{bmatrix} x \\ y \end{bmatrix} \right\|_2 / \left\| \begin{bmatrix} x' \\ y' \end{bmatrix} \right\|_2,
\]

where \( x \) and \( y \) denote the approximate solutions of the saddle point problems (1).

For both matrices, the relative error norm \( \varepsilon \) tended to increase as \( m \) increased. Moreover, the accuracy of the approximate solutions obtained by the proposed approach was worse than that by the GMRES method.
5. Conclusions and future works

In this paper, we dealt with solving the saddle point problems by iterative methods. For the conventional approach, the convergence behavior of the Krylov subspace methods for the saddle point problems tended to be poor as the number $m$ of the columns of the matrices $B$ and $C$ increased. To overcome this situation, we have proposed an approach for solving the saddle point problems using block structure. Instead of solving the saddle point problems, our proposed method needs to solve the linear system with multiple right-hand sides. We have also proposed to efficiently solve the linear systems with multiple right-hand sides by the Block Krylov subspace methods. In the conventional approach, the number of iterations of the Krylov subspace methods increased as $m$ increased, but in the proposed approach with Block GWBiCGSTABrQ, the number of iterations required to satisfy the convergence criterion decreased. Consequently, the computation time of the proposed approach with Block GWBiCGSTABrQ was shorter than that of the GMRES method.

In contrast, the accuracy of the approximate solution obtained by the proposed approach
was worse than that by the GMRES method. Improvement of the accuracy of the approxi-
mate solution generated by the proposed approach is our future work. Moreover, the perfor-
mance evaluation of the proposed approach in a massively parallel environment is also our
future work.

Acknowledgement

This research was supported by JSPS KAKENHI Grant Number JP20K11840. This re-
search used computational resources of the Cygnus Supercomputer provided by Multidisci-
plinary Cooperative Research Program in Center for Computational Sciences, University of
Tsukuba.

References

[1] M. Benzi, G. H. Golub, J. Liesen: Numerical solution of saddle point problems, Acta
Numerica, 14 (2005), 1–137.

[2] M. R. Hestenes, E. Stiefel: Methods of conjugate gradients for solving linear systems,
J. Res. Nat. Bur. Standards, 49 (1952), 409–436.

[3] R. Fletcher: Conjugate gradient methods for indefinite systems, Lecture Notes in
Math., 506 (1976), 73–89.

[4] H. A. Van der Vorst: Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG
for the solution of nonsymmetric linear systems, SIAM J. Sci. Stat. Comput., 13 (1992),
631–644.

[5] Y. Saad, M. H. Schultz: GMRES: A generalized minimal residual algorithm for solv-
ing nonsymmetric linear systems, SIAM J. Sci. Stat. Comput., 7 (1986), 856–869.

[6] T. A. Davis, Y. Hu: The university of Florida sparse matrix collection, ACM Trans.
Math. Software, 38 (2011). (https://sparse.tamu.edu)

[7] D. P. O’Leary: The block conjugate gradient algorithm and related methods, Linear
Algebra Appl., 29 (1980), 293–322.

[8] A. El Guennouni, K. Jbilou, H. Sadok: A block version of BiCGSTAB for linear
systems with multiple right-hand sides, Electron. Trans. Numer. Anal, 16 (2003), 129–
142.

[9] B. Vital: Etude de quelques méthodes de résolution de problèmes linéaires de grande
taille sur multiprocesseur, Ph.D. thesis, Université de Rennes I, Rennes, France, 1990.

[10] H. Tadano, R. Kuramoto: Accuracy improvement of the Block BiCGSTAB method
for linear systems with multiple right-hand sides by group-wise updating technique, J.
Adv. Simulat. Sci. Eng., 6 (2019), 100–117.
[11] H. Tadano: Development of the Block BiCGGR2 method for linear systems with multiple right-hand sides, *Japan J. Indust. Appl. Math.*, (2019), 563–577.

[12] A. A. Dubrulle: Retooling the method of block conjugate gradients, *Electron. Trans. Numer. Anal.*, 12 (2001), 216–233.

[13] T. Fukaya, Y. Nakatsukasa, Y. Yanagisawa, Y. Yamamoto: Cholesky QR2: a simple and communication-avoiding algorithm for computing a tall-skinny QR factorization a large-scale parallel system, *Proc. of Scal’14*, (2014), 31–38.