DOI 10.1007/s10958-021-05371-w
Journal of Mathematical Sciences, Vol. 255, No. 3, June, 2021

PARALLEL VARIABLE-TRIANGULAR ITERATIVE METHODS IN KRYLOV SUBSPACES

V. P. Il'in* UDC 519.6

The paper considers parallel preconditioned iterative methods in Krylov subspaces for solving systems of linear algebraic equations with large sparse symmetric positive-definite matrices resulting from grid approximations of multidimensional problems. For preconditioning, generalized block algorithms of symmetric successive over-relaxation or incomplete factorization type with matching row sums are used. Preconditioners are based on variable-triangular matrix factors with multiple alternations in triangular structure. For three-dimensional grid algebraic systems, methods are based on nested factorizations, as well as on two-level iterative processes. Successive approximations in Krylov subspaces are computed by applying a family of conjugate direction algorithms with various orthogonality and variational properties, including preconditioned conjugate gradient, conjugate residual, and minimal error methods. Bibliography: 23 *titles.*

1. Introduction

The aim of this paper is to design and study parallel iterative methods in Krylov subspaces for solving systems of linear algebraic equations (SLAEs)

$$Au = f, \quad A \in \mathcal{R}^{N,N}, \quad u, f \in \mathcal{R}^N, \tag{1}$$

with real large sparse matrices of order $N \approx 10^{10}$ and more having large condition numbers ($\geqslant 10^{13}$), whose implementation on modern multiprocessor computer systems (MPS) is a challenging practical problem. In particular, in solving direct and inverse interdisciplinary problems of mathematical modeling with real data, including nonlinear and nonstationary ones, this stage of computations can take about 80% of the time of the machine experiment because here the amount of computer resources consumed increases nonlinearly as the number of degrees of freedom grows.

We are mainly interested in SLAEs that arise from approximations of multi-dimensional initial-boundary-value problems, characterized by variable coefficients and contrasting physical properties, using finite-difference, finite volume, finite element, or discontinuous Galerkin methods [1]. It is assumed that in such cases, special high-performance methods of Fast Fourier Transform type are not directly applicable.

The main approaches used are based on preconditioned iterative algorithms in Krylov subspaces. A typical easily invertible preconditioner is an approximate triangular factorization of the form

$$B = (G + L)G^{-1}(G + U) = G + L + U + LG^{-1}U, \tag{2}$$

where L and U are the lower and upper block triangular parts of an original matrix $A = D + L + U$, and G, D are nonsingular block diagonal matrices. Note that if in (2) the matrix G is determined from the matrix equation

$$G = D - LG^{-1}U, \tag{3}$$

then an exact block factorization of the matrix A is obtained.

*Institute of Computational Mathematics and Mathematical Geophysics SB RAS, Novosibirsk, Russia; Novosibirsk State University, Novosibirsk, Russia, e-mail: ilin@sscc.ru.

Translated from *Zapiski Nauchnykh Seminarov POMI*, Vol. 496, 2020, pp. 104–119. Original article submitted October 23, 2020.

On the base of approximate counterparts of formula (3), various methods of symmetric successive over-relaxation (SSOR) type, as well as explicit and implicit incomplete factorization methods (in particular, ILU algorithms for incomplete triangular decomposition) are constructed, see the surveys [2–7].

The methods considered are efficient methods for solving block tridiagonal SLAEs, arising in many applications. For example, if, in a two-dimensional rectangular domain on a rectangular (possibly nonuniform) grid with the node number $N = N_x N_y$ (N_x and N_y are the numbers of steps along the coordinates x and y, respectively), one uses the natural ordering and the standard five- or nine-points approximation of diffusion type equations [1], then an algebraic system of the form

$$Au = \begin{bmatrix} D_1 & U_1 & & & 0 \\ L_2 & D_2 & U_2 & & \\ & \ddots & \ddots & \ddots & \\ & & & & U_{N_x-1} \\ 0 & & & L_{N_x} & D_{N_x} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \cdot \\ \\ u_{N_x} \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ \cdot \\ \\ f_{N_x} \end{bmatrix} \tag{4}$$

is obtained. Here, the matrices $D = \{D_i\}, L = \{L_i\}$, and $U = \{U_i\}$ are block diagonal ($D_i, L_i \in \mathcal{R}^{N_y,N_y}$), and the matrix A is assumed to be symmetric positive definite (SPD).

Possible generalizations will separately be discussed below. Note that if the matrix A has the block tridiagonal structure (4), then its exact block factorization is provided by relations (2), (3), which, in this case, reduce to block sweeping and involve the block diagonal matrix $G = \text{block-diag}\{G_i\}$, whose blocks G_i are determined recursively. Since the implementation of such an algorithm requires time-consuming inversion of the matrices G_i, it is preferable to construct cost-effective iterative methods that use banded approximate inverses of banded matrices. In this case, the preconditioning quality can be improved based on the following row-sum matching approach, see [2, 5, 8]:

$$G_1 = D_1, \quad \bar{G}_i = D_i - L_i \bar{G}_{i-1}^{-1} U_{i-1} - \theta S_i, \quad i = 1, 2, ..., N_x. \tag{5}$$

Here, $\bar{G}_{i-1}^{-1}$ is the tridiagonal part of the matrix G_{i-1}^{-1} (its principal diagonal and two codiagonals); $\theta \in [0, 1]$ is a compensation parameter, and S_i are the diagonal matrices determined from the principle of generalized row-sum matching, or filtration, in accordance with the formulas

$$Sy^{(l)} = [L(G^{-1} - (\bar{G}^{-1})U]y^{(l)}, \quad l = 1, m,$$

where

$$S = \text{block-diag}\{S_i\}, \quad \bar{G} = \text{diag}\{\bar{G}_i^{-1} = \text{3-diag}\{G_i^{-1}\}\};$$

$y^{(l)}$ are some trial, or filter, vectors; $m = 1$ or $m = 2$.

A simpler way to construct a preconditioner consists in setting $G_i = \omega^{-1} D_i$, instead of solving (5), which leads to the block symmetric successive over-relaxation (BSSOR) method [2, 5]. The optimal value of the relaxation parameter ω is chosen in the semiopen interval [1, 2).

If the computational domain is a three-dimensional parallelepiped and the grid has $N = N_x N_y N_z$ nodes (N_z is the number of steps along the axis z), then the corresponding SLAE can also be represented in the form (4), but the matrices D_i, L_i, U_i will be of order $N_y N_z$.

Denote the characteristic step size of the grid by h and assume that the grid is regular, or quasi-uniform, i.e., all step sizes of the grid are of the same order as $h \to 0$. Then the condition number of the matrix A is of order

$$\text{cond}(A) = \max\{\lambda(A)\}/\min\{\lambda(A)\} = O(h^{-2}).$$

Moreover, $\text{cond}(A)$ increases as the grid becomes more nonuniform and the coefficients of the original equation to be solved become more contrasting. By using a preconditioning matrix B

of the form (2), the condition number of the resulting preconditioned SLAE can be decreased by an order of magnitude, i.e., $\mathrm{cond}(B^{-1}A) = O(h^{-1})$. By applying iterative algorithms in Krylov subspaces, such a preconditioned system can be solved in $n = O(h^{-1/2})$ iterations.

This high efficiency is deteriorated by the fact that for the algorithms under consideration, it is difficult to improve the performance on multiprocessor computer systems because inversion of the preconditioning matrix B requires solution of poorly parallelizable triangular linear systems.

In [9], a two-threaded block version of the variable-triangular factorization of the original matrix was proposed, where each of the factors in the preconditioner B was not a lower or an upper triangular matrix but consisted of block rows of different orientations: some were lower triangular, and the remaining ones were upper triangular (this decomposition was called "twisted decomposition" by the authors; in the Russian literature, this approach is conventionally referred to as the algorithm of counter sweeps, see [10]).

We will consider a generalization of this technique based on using variable-triangular matrices L and U consisting of P block rows that successively switch from the lower to upper triangular part of the matrix and vice versa. In this case, solution of linear systems with the matrices $G + L$ and $G + U$ can be parallelized on P processors.

The present paper is organized as follows. Section 2 describes the variable-triangular iterative methods of block symmetric successive over-relaxation and incomplete factorization types in Krylov subspaces for solving two-dimensional grid boundary-value problems described by block tridiagonal SLAEs of the form of (4). Section 3 deals with a generalization of the approach proposed to parallelizing solution of three-dimensional grid algebraic systems by using the nested factorization methods, originally proposed in [11, 12] and developed later by many authors, see [13–15], which are widely used in the software in oil and gas industry [16]. Section 4 describes the variable-triangular preconditioned methods in Krylov subspaces of three types (conjugate gradient, conjugate residual, and minimal error algorithms) with various variational and orthogonality properties. In the Conclusion, we discuss directions of further research that are promising with respect to increasing both the rate of convergence of iterations and the parallel performance of the algorithms considered.

2. Variable-triangular methods for solving two-dimensional grid problems

When using the matrix (2) as a preconditioner for an iterative solution method, at every iteration it is necessary to solve an auxiliary system of the form (see (32) in Sec. 4)

$$Bp \equiv (G + L)G^{-1}(G + U)p = r. \tag{6}$$

Its solution can be found by solving the two linear systems

$$(G + L)v = r, \quad (G + U)p = Gv = w. \tag{7}$$

As is known, solution of systems (7) with triangular matrices L, U is poorly parallelizable on multiprocessor computer systems. As an alternative, we will consider matrices and methods of the variable-triangular type. The definition of variable-triangular matrices and of algorithms for solving the corresponding SLAEs is first illustrated on the example of block tridiagonal systems of the form (4) in the case where $N_x = 7$ and "triangularity" changes only once:

$$(G+L)v \equiv \left[\begin{array}{ccc|c|ccc} G_1 & 0 & 0 & 0 & 0 & 0 & 0 \\ L_2 & G_2 & 0 & 0 & 0 & 0 & 0 \\ 0 & L_3 & G_3 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & L_4 & G_4 & U_4 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & G_5 & U_5 & 0 \\ 0 & 0 & 0 & 0 & 0 & G_6 & U_6 \\ 0 & 0 & 0 & 0 & 0 & 0 & G_7 \end{array}\right] \left[\begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \end{array}\right] = \left[\begin{array}{c} r_1 \\ r_2 \\ r_3 \\ r_4 \\ r_5 \\ r_6 \\ r_7 \end{array}\right] = r, \tag{8}$$

$$(G+U)p \equiv \left[\begin{array}{ccc|c|ccc} G_1 & U_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & G_2 & U_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & G_3 & U_3 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & G_4 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & L_5 & G_5 & 0 & 0 \\ 0 & 0 & 0 & 0 & L_6 & G_6 & 0 \\ 0 & 0 & 0 & 0 & 0 & L_7 & G_7 \end{array}\right] \left[\begin{array}{c} p_1 \\ p_2 \\ p_3 \\ p_4 \\ p_5 \\ p_6 \\ p_7 \end{array}\right] = \left[\begin{array}{c} w_1 \\ w_2 \\ w_3 \\ w_4 \\ w_5 \\ w_6 \\ w_7 \end{array}\right] = Gv. \tag{9}$$

In relations (8), (9), the matrices L, U are not the same as in (2), but the sums $L+U$ are the same in both cases.

For these structures of the matrices $G+L$ and $G+U$, formula (3) for G, which ensures the equality $B=A$, remains intact. However, in this case, the recurrence relation (5) for computing the blocks G_i changes significantly. Now, they are computed by the counter sweeping from top and bottom to center in accordance with the following formulas:

$$\begin{aligned} G_1 &= D_1, \quad G_i = D_i - L_i\bar{G}_{i-1}^{-1}U_{i-1} - \theta S_i, \\ S_i y^{(l)} &= L_i(G_{i-1}^{-1} - \bar{G}_{i-1}^{-1})U_{i-1}y^{(l)}, \quad i = 2, \ldots, m, \\ G_{N_x} &= D_{N_x}, \quad G_i = D_i - U_i\bar{G}_{i+1}^{-1}L_{i+1} - \theta S_i, \\ S_i y^{(l)} &= U_i(G_{i+1}^{-1} - \bar{G}_{i+1}^{-1})L_{i+1}y^{(l)}, \quad i = N_x - 1, \ldots, m+2, \\ G_{m+1} &= D_{m+1} - L_{m+1}\bar{G}_m^{-1}U_m - U_{m+1}\bar{G}_{m+2}^{-1}L_{m+2} - \theta S_{m+1}, \\ S_{m+1}y^{(l)} &= [L_{m+1}(G_m^{-1} - \bar{G}_m^{-1})U_m + U_{m+1}(G_{m+2}^{-1} - \bar{G}_{m+2}^{-1})L_{m+2}]y^{(l)}. \end{aligned}$$

Naturally, all the $2m+1$ tridiagonal matrices G_i are computed once before iteration and then stored. In this case, the forward ($i = 1, ..., m$) and backward sweeps ($i = 2m+1, ..., m+2$) are easily parallelized on two threads, or processor cores.

Each of the equations in (8), (9) can be solved in parallel as follows. From (8) the block unknowns are computed in the following order: first, $v_1 = G_1^{-1}r_1$, $v_i = G_i^{-1}(r_i - U_i v_{i+1})$ for $i = 2, 3$ and $v_7 = G_7^{-1}r_7$, $v_i = G_i^{-1}(r_i - L_i v_{i-1})$ for $i = 6, 5$ are computed synchronously; finally, one computes $v_4 = G_4^{-1}(r_4 - L_4 r_3 - U_4 r_5)$. Using similar formulas but in a different order, one solves SLAE (9): first, one computes p_4, and then p_3, p_2, p_1 and p_5, p_6, p_7 are computed in parallel.

In order to pass from a particular case to the general description of the variable-triangular method, it is necessary to formalize the large-block structure of the matrices $G+L$ and $G+U = (G+L)^\top$ for sufficiently large values of the block order N_x.

Each of the matrices in SLAEs of the forms (8), (9) can be represented as two diagonal blocks of the same (for simplicity) block order m separated by a block row and a block column. These "large" blocks are lower and upper block triangular.

We define a periodic structure as a pair of matrix blocks, the upper left one being lower triangular, and the other one being upper triangular. These blocks are separated by a "cross"

formed by a block tridiagonal block row and a block diagonal block column. Thus, the block order of these matrices is $2m+1$.

Now we turn to the more general case of multi-variable triangulation and assume that the matrices A, $G+L$, and $G+U$ have a periodic block tridiagonal structure with block order $N_x=(2m+1)P+P-1$. In this case, matrices with twice-variable triangulation of the form (8), (9) will consist of diagonal blocks, each of which has order $2m+1$ and is separated from its neighbors by a "cross" consisting of one block row and one block column. Denoting these "principal blocks" in $G+L$ and $G+U$ by $\widehat{H}_s$ and $\breve{H}_s$, respectively, $s=1,2,\ldots,P$, we can write the resulting SLAE as follows:

$$(G+L)v=\begin{bmatrix} \widehat{H}_1 & \widehat{U}_1 & 0 & 0 & & & 0 \\ 0 & \bar{G}_1 & 0 & 0 & & & \\ 0 & \widehat{L}_2 & \widehat{H}_2 & \widehat{U}_2 & & & \\ 0 & 0 & 0 & \bar{G}_2 & \ddots & & \\ & & & \ddots & \ddots & \ddots & \\ & & & & \ddots & \bar{G}_{P-1} & 0 \\ 0 & & & & 0 & \widehat{L}_P & \widehat{H}_P \end{bmatrix} \begin{bmatrix} \widehat{v}_1 \\ \bar{v}_1 \\ \widehat{v}_2 \\ \bar{v}_2 \\ \vdots \\ \bar{v}_{P-1} \\ \widehat{v}_P \end{bmatrix} = \begin{bmatrix} \widehat{r}_1 \\ \bar{r}_1 \\ \widehat{r}_2 \\ \bar{r}_2 \\ \vdots \\ \bar{r}_{P-1} \\ \widehat{r}_P \end{bmatrix}, \tag{10}$$

$$(G+U)p=\begin{bmatrix} \breve{H}_1 & 0 & 0 & 0 & & & 0 \\ \breve{L}_1 & \bar{G}_1 & \breve{U}_1 & 0 & & & \\ 0 & 0 & \breve{H}_2 & 0 & & & \\ 0 & 0 & \breve{L}_2 & \bar{G}_2 & \breve{U}_2 & & \\ & & \ddots & \ddots & \ddots & & \\ & & & \breve{L}_{P-1} & \bar{G}_{P-1} & \breve{U}_{P-1} & \\ 0 & & & 0 & 0 & \breve{H}_P \end{bmatrix} \begin{pmatrix} \breve{p}_1 \\ \bar{p}_1 \\ \breve{p}_2 \\ \bar{p}_2 \\ \vdots \\ \bar{p}_{P-1} \\ \breve{p}_P \end{pmatrix} = \begin{bmatrix} \breve{w}_1 \\ \bar{w}_1 \\ \breve{w}_2 \\ \bar{w}_2 \\ \vdots \\ \bar{w}_{P-1} \\ \breve{w}_P \end{bmatrix}. \tag{11}$$

Here, $\bar{G}_s=G_{q_s}$, where $q_s=(2m+1)s+1$, is an "ordinary" tridiagonal matrix (9) of order N_y, and the matrices $\widehat{L}_s,\breve{L}_s,\widehat{U}_s$, and $\breve{U}_s$ are the following block rows and columns:

$$\widehat{L}_s=\begin{bmatrix} L_{q_{s+1}} \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad \widehat{U}_s=\begin{bmatrix} 0 \\ \vdots \\ 0 \\ U_{q_{s-1}} \end{bmatrix} \in \mathcal{R}^{2m+1,N_y}, \qquad \begin{aligned} &\breve{L}_s=[0\ \ldots\ 0\,L_{q_s}]\in\mathcal{R}^{N_y,2m+1},\\ &\breve{U}_s=[U_{q_s}\,0\ \ldots\ 0]. \end{aligned} \tag{12}$$

As is seen from (10), (11), the vectors $\widehat{v}_s,\widehat{r}_s,\breve{p}_s,\breve{w}_s$ have dimension $(2m+1)N_y$, and each of the vectors $\bar{v}_s,\bar{r}_s,\bar{p}_s,\bar{w}_s$ has dimension N_y.

The solution of algebraic systems (10) and (11) on an MPS can be carried out in parallel, but different computational schemes must be used. In SLAE (10), first the variable separators $\bar{v}_s=\bar{G}_s^{-1}\bar{r}_s$, $s=1,2,\ldots,P-1$, are simultaneously computed, and then the large block components are determined from P systems of the form

$$\widehat{H}_s\widehat{v}_s=\widehat{r}_s-\widehat{L}_s\bar{v}_{s-1}-\widehat{U}_s\bar{v}_{s+1}. \tag{13}$$

In this case, each of systems (13) is solved similarly to SLAE (8), i.e., using counter sweeping from top and bottom to center.

Equations of the form (11) are solved in the reverse order. First, from the systems

$$\breve{H}_s\breve{p}_s=\breve{w}_s, \quad s=1,2,\ldots,P, \tag{14}$$

the "large" subvectors $\breve{p}_s$ are simultaneously determined. In this case, similarly to the implementation of (9), the counter sweepings from center to top and bottom are applied. At the final stage, the components of the separator are computed by solving the SLAEs

$$\bar{G}_s\bar{p}_s = \bar{w}_s - \bar{L}_s\breve{p}_{s-1} - \bar{U}_s\breve{p}_{s+1}, \quad s = 1, 2, \ldots, P-1. \tag{15}$$

3. Variable-triangular algorithms for three-dimensional problems

When solving three-dimensional boundary-value problems, we focus on the standard seven-point approximation of diffusion type equations on parallelepipedal grids. The coefficient matrix of the system of equations is of order $N = N_x N_y N_z$. Instead of using (4), one can represent it in the form

$$A = D + L_1 + U_1 + L_2 + U_2 + L_3 + U_3. \tag{16}$$

Here, D is the principal diagonal of the original matrix, whereas L_l and U_l, $l = 1, 2, 3$, are the codiagonals of the matrix A related to its lower and upper triangular parts. The indices $l = 1, 2, 3$ can be interpreted as those corresponding to the Cartesian variables x, y, z if three-point approximations of the second-order derivatives of the differential equation solved are used.

In this case, we will use nested factorization methods, which are obtained if a preconditioning matrix B of the form (2) is constructed in the following recursive way (see [11–16]):

$$\begin{aligned} B &= (P + L_3)P^{-1}(P + U_3) = P + L_3 + U_3 + L_3P^{-1}U_3, \\ P &= (T + L_2)T^{-1}(T + U_2) = T + L_2 + U_2 + L_2T^{-1}U_2, \\ T &= (M + L_1)M^{-1}(M + U_1) = M + L_1 + U_1 + L_1M^{-1}U_1. \end{aligned} \tag{17}$$

As a result, we have

$$B = M + A - D + L_1M^{-1}U_1 + L_2T^{-1}U_2 + L_3P^{-1}U_3. \tag{18}$$

If the natural ordering of the grid nodes and the corresponding vector components is used, then the matrices M, T, and P have diagonal, tridiagonal, and pentadiagonal forms, respectively, and the preconditioner B is defined by the formulas

$$M = D - L_M^{-1}U_1 - \theta_1 S_1 - \theta_2 S_2, \quad B = A + L_2T^{-1}U_2 + L_3P^{-1}U_3 - \theta_1 S_1 - \theta_2 S_2. \tag{19}$$

Here, θ_1 and θ_2 are iterative (relaxation, or compensating) parameters, whereas S_1 and S_2 are the diagonal matrices arising from the principle of matching row sums ($Ae = Be$), i.e.,

$$S_1 e = L_2T^{-1}U_2 e, \quad S_2 e = L_3P^{-1}U_3 e, \tag{20}$$

where $e = \{1\}$ is the all 1's vector.

Note that in (19), the matrix equation for M is uniquely solvable in the class of diagonal matrices, i.e., M has the same structure as D.

In formulas (17)–(20) of the nested factorization method, the matrices L_l, $l = 1, 2, 3$, are lower triangular, whereas U_l, $l = 1, 2, 3$, are upper triangular. If the matrices L_2, L_3, U_2, and U_3 are determined as in the previous section, then we obtain a variable-triangular version of the nested factorization algorithm. Note that the matrices L_1 and U_1 must satisfy the following condition: they must be defined as single-variable-triangular ones, and the left block in L_1 must be lower triangular. Only in this case, the matrix equation for M from (19) is solvable in the class of diagonal matrices. Actually, the implementation of factorization along the first direction reduces to the scalar version of counter sweeping, which can be implemented on $P_1 = 2$ computational threads. The variable-triangular matrices along the second and third

directions can be defined as multivariable-triangular ($P_2 \le N_y$, $P_3 \le N_z$). Thus, this version of nested factorization allows parallelization on $P = 2P_2P_3$ threads.

If a preconditioner B of the form (17), (18) is applied to an iterative process in Krylov subspaces, we obtain a three-level factorization in the classical one-level iterative method. However, when solving three-dimensional grid boundary-value problems under consideration, one also can use a two-level factorization and a two-level iterative process.

To this end, the original matrix A from (16) is represented in the form

$$A = D_3 + L_3 + U_3, \quad D_3 = D_2 + L_2 + U_2, \quad D_2 = D + L_1 + U_1. \tag{21}$$

In this case, D_3 is a block diagonal matrix with pentadiagonal blocks $D_{3,i}$ of order N_yN_z, each of which corresponds to a two-dimensional problem in the section $x = \text{const}$ and has the structure of the matrix A in (6). Then the matrix P in (17) is defined by the formula

$$P = \{G_i = D_{3,i} - \theta S_i\}, \quad S_i e = L_{3,i} G_{i-1}^{-1} U_{i-1} e, \tag{22}$$

implying that $L_3 P^{-1} U_3 = 0$ in (17). Note that L_3 and U_3 can be defined as variable-triangular, and then the implementation of the algorithm can be parallelized using block counter sweeping.

If we set $\theta = 0$ in (22), then we arrive at the block symmetric successive over-relaxation method (BSSOR, [4]). In this case, in (17), the matrix P must be replaced by $\omega^{-1}P$, where the relaxation parameter ω has an optimum value on the interval $[1, 2]$.

Note that each of the auxiliary two-dimensional SLAEs is strictly diagonally dominant and has a finite condition number, and its eigenvalues can be bounded using the Gerschgorin disks.

4. Variable-triangular methods in Krylov subspaces

We consider iterative algorithms in application to the SLAE obtained from the original system of the form (1) as a result of its two-sided preconditioning,

$$\bar{A}\bar{u} = \bar{f}, \quad \bar{A} = L_B^{-1} A U_B^{-1}, \quad \bar{u} = U_B u, \quad \bar{f} = L_B^{-1} f. \tag{23}$$

Here, L_B and U_B are the factors of the triangular decomposition of a nonsingular preconditioning matrix B,

$$B = L_B U_B, \quad B^{-1} = U_B^{-1} L_B^{-1}. \tag{24}$$

In order to solve the preconditioned SLAE $\bar{A}\bar{u} = \bar{f}$ with an SPD matrix $\bar{A} = \bar{A}^\top$, we apply an iterative process of the form

$$\begin{aligned} \bar{u}^{n+1} &= \bar{u}^n + \alpha_n p^n = \bar{u}^0 + \alpha_0 p^n + \cdots + \alpha_n p^n, \\ \bar{r}^{n+1} &= \bar{f} - \bar{A}\bar{u}^{n+1} = \bar{r}^n - \alpha_n \bar{A} p^n = \bar{r}^0 - \alpha_0 \bar{A} p^0 - \cdots - \alpha_n \bar{A} p^n, \end{aligned} \tag{25}$$

where p^n are some direction vectors; α_n are iteration parameters; $\bar{u}^0 = U_B u^0$ and $\bar{r}^0 = \bar{f} - \bar{A}\bar{u}^0$ are the preconditioned initial guess and residual, and u^0 is an arbitrary vector.

We assume that the vectors p^n in (25) satisfy the orthogonality conditions

$$\left(\bar{A}^\gamma p^n, p^k\right) = (p^n, p^k)_\gamma = \rho_n^{(\gamma)} \delta_{k,n}, \quad \rho_n^{(\gamma)} = (p^n, p^n)_\gamma, \tag{26}$$

where $\gamma = 0, 1, 2$ and $\delta_{k,n}$ is the Kronecker symbol. Under these assumptions, for the residual we have the expression

$$\left(\bar{r}^{n+1}, \bar{r}^{n+1}\right)_{\gamma-2} = \left(\bar{r}^0, \bar{r}^0\right)_{\gamma-2} - \sum_{k=0}^{n} \left[2\alpha_k \left(p^k, p^k\right)_{\gamma-1} - \alpha_k^2 \left(p^k, p^k\right)_\gamma\right].$$

It follows that if the coefficients α_n are defined by

$$\alpha_n = \sigma_n / \rho_n, \quad \sigma_k = \left(\bar{r}_0, p^k\right)_{\gamma-1}, \tag{27}$$

then the residual functional satisfies the relations

$$\Phi^{\gamma}\left(\bar{r}^{n+1}\right) = \Phi^{\gamma}\left(\bar{r}^{0}\right) - \sum_{k=0}^{n} \sigma_k^2/\rho_k, \tag{28}$$

where the superscript γ in α_k, σ_k, and ρ_k is omitted for simplicity. In this case, the functional attains its minimum in the Krylov subspaces

$$\mathcal{K}_{n+1}\left(\bar{r}^{0}, \bar{A}\right) = \operatorname{Span}\left\{\bar{r}^{0}, \bar{A}\bar{r}^{0}, \ldots, \bar{A}^{n}\bar{r}^{0}\right\}. \tag{29}$$

For the orthogonality conditions (26) to be fulfilled, the direction vectors are determined from the recurrence relation

$$p^0 = \bar{r}^0, \quad p^{n+1} = \bar{r}^{n+1} + \beta_n p_n, \quad \beta_n = -\left(\bar{r}^{n+1}, p^n\right)_{\gamma}/\rho_n, \tag{30}$$

where the rule for finding the initial vector p^0 is commonly accepted but not mandatory (strictly speaking, it can be chosen arbitrarily). In this case, the additional orthogonality conditions

$$\left(r^k, r^n\right)_{\gamma-1} = ||r^n||_{\gamma-1}\delta_{k,n}, \quad \left(r^n, p^k\right)_{\gamma-1} = 0, \quad k < n,$$

are fulfilled, and we have the equalities $(\bar{r}^0, p^n)_{\gamma-1} = (\bar{r}^n, \bar{r}^n)_{\gamma-1}$, from which the following new formulas for the coefficients σ_n and β_n are obtained:

$$\sigma_n = \left(\bar{r}^n, \bar{r}^n\right)_{\gamma-1}, \quad \beta_n = \sigma_{n+1}/\sigma_n. \tag{31}$$

For $\gamma = 1, 2$, the above-described conjugate direction algorithms are known under the names of conjugate gradient (CG) and conjugate residual (CR) methods, respectively, see [4, 14] and the references therein. From the above relations for $\gamma = 1, 2$ we obtain the following formulas in terms of the matrices A and B:

- for the conjugate gradient methods,

$$\begin{aligned} &r^0 = f - Au^0, \quad p^0 = B^{-1}r^0, \quad \alpha_n = \sigma_n/\rho_n, \\ &u^{n+1} = u^n + \alpha_n p_n, \quad r^{n+1} = r^n - \alpha_n A p^n, \quad p^{n+1} = B^{-1}r^{n+1} + \beta_n p^n, \\ &\sigma_n = \left(B^{-1}r^n, r^n\right), \quad \rho_n = \left(Ar^n, r^n\right), \quad \beta_n = \sigma_{n+1}/\sigma_n; \end{aligned} \tag{32}$$

- for the conjugate residual methods,

$$\begin{aligned} &r^0 = f - Au^0, \quad \hat{r}^0 = p^0 = B^{-1}r^0, \\ &u^{n+1} = u^n + \alpha_n p_n, \quad \hat{r}^{n+1} = \hat{r}^n - \alpha_n B^{-1}Ap^n, \quad p^{n+1} = \hat{r}^{n+1} + \beta_n p_n, \\ &\sigma_n = \left(A\hat{r}^n, \hat{r}^n\right), \quad \rho_n = \left(B^{-1}Ap^n, Ap^n\right). \end{aligned} \tag{33}$$

Note that at every iteration of both methods one multiplication by each of the matrices A and B^{-1} is required, and the vector $\hat{r}^n$ in (33) is not the true but the preconditioned residual, i.e., in exact arithmetic, we have $\hat{r}^n = B^{-1}r^n = B^{-1}(f - Au^n)$, and the value of $(B^{-1}r^n, r^n)$ is minimized at every iteration. The algorithms (32), (33) are conventionally denoted by PCG and PCR, respectively.

For $\gamma = 0$, the algorithm should be implemented in a different way because, in this case, in order to find σ_n, one needs to invert the matrix $\bar{A}$. The resulting algorithms are known as the minimum error methods, or minimum iterations. These algorithms were first studied in [17–20]; later, in connection with the algebraic problem of moments, they were investigated in [6]. Using (25), we write the error and residual vectors as

$$\begin{aligned} &v^n = \bar{u} - \bar{u}^n = \alpha_n p^n + \cdots + \alpha_M p^m, \\ &\bar{r}^n = Av^n = \alpha_n Ap^m + \cdots + \alpha_M \bar{A}p^M. \end{aligned} \qquad M \le N, \tag{34}$$

Upon multiplying the first of these relations by p^n, we obtain

$$\alpha_n = (v^n, p^n)/(p^n, p^n) = -\alpha_{n-1}(\bar{A}v^n, p^{n-1}) = -\alpha_{n-1}(\bar{r}^n, p^{n-1})/||p^n||.$$

Here, for $n \geq 1$, we use the three-term recursion from (25), (30) (if $n = 1$, then $\beta_{-1} = 0$)

$$p^n = (1 + \beta_{n-1})p^{n-1} - \alpha_{n-1}\bar{A}p^{n-1} - \beta_{n-2}p^{n-2},$$

and also the orthogonality property of the direction vectors p^n and the symmetry of the matrix $\bar{A}$. For $n = 0$, we have

$$\alpha_0 = (v^0, p^0)/(p^0, p^0).$$

Hence, due to the arbitrariness in the choice of the initial direction vector, we may set $p^0 = \bar{A}\bar{r}^0$, in which case we obtain

$$\alpha_0 = (\bar{r}^0, \bar{r}^0)/(\bar{A}\bar{r}^0, \bar{A}\bar{r}^0) = (B^{-1}r^0, r^0)/(B^{-1}AB^{-1}r^0, AB^{-1}r^0). \tag{35}$$

The coefficients β_n are computed by formula (30).

One of the important issues related to implementation of the methods (32), (33) arises in the case where the preconditioning matrix B is inverted approximately. Actually, the inversion of B reduces to iterative solution of the corresponding auxiliary SLAE. This implies that a nonzero residual and an error are allowed. In this case, we obtain a two-level iterative method. Although the problem of stopping criteria is quite subtle, we assume, for simplicity, that in the preconditioned CG and CR algorithms, the outer iterations are continued until

$$||r^n|| \leq \varepsilon_e||f||, \quad \varepsilon_e \ll 1. \tag{36}$$

Similarly, at inner iterations, we choose some other accuracy parameter $\varepsilon_i \leq 1$. For example, the approximate direction vector $\bar{p}^0$ is determined from the system $\bar{B}p^0 = r^0$ in the following way:

$$\delta^0 = r^0 - B\bar{p}^0, \quad \bar{p}^0 = B^{-1}(r^0 - \delta^0), \quad ||\delta^0|| \leq \varepsilon_i||r^0||. \tag{37}$$

5. Conclusion

The approaches to constructing variable-triangular preconditioning matrices considered in the paper provide new possibilities for parallelization of conventional and/or block iterative methods of symmetric successive over-relaxation and incomplete factorization types in Krylov subspaces (including the nested ones). So far, the theoretical problem of estimating the convergence rate of the new iterative processes remains open. In this connection, it is necessary, first of all, to conduct systematic experimental studies and testing of various modifications of the algorithms suggested. The ultimate goal, in this case, is acceleration of computations and improvement of parallel performance on multiprocessor computer systems with distributed and hierarchical shared memory.

This work was supported by the Russian Foundation for Basic Research (grant No. 18-01-00295).

Translated by V. P. Il'in.

REFERENCES

1. V. P. Il'in, *Mathematical Modeling. Part I. Continuous and Discrete Models* [in Russian], SBRAS Publ., Novosibirsk (2017).
2. O. Axelsson, *Iterative Solution Methods*, Cambridge Univ. Press, Cambridge (1994).
3. V. P. Il'in, *Iterative Incomplete Factorization Methods for Solving Algebraic Systems* [in Russian], Nauka, Moscow (1995).
4. Y. Saad, *Iterative Methods for Sparse Linear Systems*, 2nd edn., SIAM (2003).

5. V. P. Il'in, *Finite Element Methods and Technologies* [in Russian], ICM&MG SB RAS Publ., Novosibirsk (2007).
6. J. Liesen and Z. Strakos, *Krylov Subspace Methods, Principles and Analysis*, Oxford Univ. Press (2013).
7. M. A. Olshanskii and E. E. Tyrtyshnikov, *Iterative Methods for Linear Systems. Theory and Applications*, SIAM (2014).
8. V. P. Il'in and K. Yu. Laevsky, "Generalized compensation principle in incomplete factorization methods," *Russ. J. Numer. Anal. Math. Model.*, **12**, No. 5, 399–412 (1997).
9. R. Wang, Q. Niu, and L. Lu, "A weighted block tangential filtering decomposition preconditioner," *Math. Probl. Eng.* (2009), Article ID 282307.
10. V. P. Il'in and Y. I. Kuznetsov, *Tridiagonal Matrices and their Applications* [in Russian], Nauka, Moscow (1985).
11. J. R. Appleyard, I. M. Cheshire, and R. K. Pollard, "Special techniques for fully-implicit simulators," in: *Proc. European Symposium on Enhanced Oil Recovery, Bournemouth* (1981), pp. 395–408.
12. J. R. Appleyard and I. M. Cheshire, "Nested factorization," in: *Proc. Seventh SPE Symposium on Reservoir Simulation* (1983), pp. 315–324.
13. N. N. Kuznetsova, O. V. Diyankov, S. V. Kotegov, I. V. Krasnogorov, V. Y. Pravilnikov, and S. Y. Maliassov, "The family of nested factorizations," *Russ. J. Numer. Anal. Math. Model.*, **22**, 393–412 (2007).
14. P. Kumar, L. Grigori, F. Nataf, and Q. Nui, "On relaxed nested factorization and combination preconditioning," *Int. J. Comput. Math.*, **93**, No. 1, 178–199 (2016).
15. P. Kumar, L. Grigori, Q. Nui, and F. Nataf, "Fourier analysis of modified nested factorization preconditioner for three-dimensional isotropic problems," Res. Rep. INRIA - 00448291 (2010).
16. *Using ECLIPSE Parallel.* Parallel computing, 2003. `www.sis.slb.com/me-dia/about/whitepaper_parallelcomputing.pdf.`
17. E. Graig, "The n-steps iterative procedures," *J. Math. Phys.*, **34**, 64–73 (1955).
18. V. M. Freedman, "Minimal iteration method with minimal errors for a system of linear algebraic equations with symmetric matrix," *Zh. Vychisl. Mat. Mat. Fiz.*, **2**, No. 2, 341–342 (1962).
19. Yu. V. Vorobyev, *Method of Moments in Applied Mathematics* [in Russian], Fizmatgiz, Moscow (1958).
20. D. K. Faddeev and V. N. Faddeeva, *Computational Methods of Linear Algebra* [in Russian], Fizmatgiz, Moscow (1960).
21. Y. Saad, M. Yeung, J. Erhel, and F. Guyomarch, "A deflated version of conjugate gradient algorithm," *SIAM J. Sci. Comput.*, **24**, 1909–1926 (2000).
22. L. Yu. Kolotilina, "Twofold deflation preconditioning of linear algebraic systems. I. Theory," *Zap. Nauchn. Semin. POMI*, **229**, 95–152 (1995).
23. N. Venkovic, P. Mycek, L. Giraud, and O. Le Maitre, "Comparative study of harmonic and Rayleigh–Ritz procedures with applications for deflated conjugate gradients," Pressed Report Cerfacs hal-02434043 (2020).