Aplikace matematiky

Ľubor Malina

General theory of direct methods for solving systems of equations with band matrices

Aplikace matematiky, Vol. 24 (1979), No. 3, 161–183

Persistent URL: http://dml.cz/dmlcz/103794

GENERAL THEORY OF DIRECT METHODS FOR SOLVING SYSTEMS OF EQUATIONS WITH BAND MATRICES

LUBOR MALINA

(Received March 30, 1977)

The main goal of the present paper is to establish a general theory of the direct methods for the solution of systems of linear equations with band matrices. Here the word "general" means that the theory should cover most of the known direct methods. Due to the growing ability of computers, much work has been devoted to the development of such methods for special systems of equations such as those with band or sparse matrices. There are two possible ways, either to adjust the known general methods to the special structure of the matrix or to create new methods. Our attention will be focused to the band matrices. Though such systems often occur in practice (e.g. in the finite element or finite difference methods), there is no general theory of the methods available up to now. We have started with the following well known fact. There is a well known theory of sweep or the so called factorization methods for the solution of differential boundary value problems (cf. [4]). Babuška in [1] has found that when solving a boundary value problem of order two by the method of simple factorization and discretizing the resulting system of initial value problems by Euler's method is nothing else but the process of discretization of the original problem by the finite difference method and the solution of the system for approximate values of the exact solution (with tridiagonal matrix) by Gaussian elimination. The main idea of the sweep methods is to transfer the boundary condition at one point over the interval of definition (if there are no transient conditions) to the other point and vice versa. At every point of the interval we obtain, roughly speaking, a system of equations for the vector of the solution with a "small" matrix. Following the above mentioned analogy between boundary value problems and systems of equations with band matrices we show what the "transfer of the boundary condition" means in the latter case.

In the first part we shall discuss some ideas from the paper [5]. This paper is probably one of the first which offer a new insight into the process of Gaussian

elimination for tridiagonal matrices close to ours. This part contains also some preliminary concepts and definitions.

The second part is devoted to the definition of a general algorithm and to the proof that the methods of this algorithm solve our problem. One special variant of the algorithm, especially suitable if we have more severe restrictions on the computer memory, is also mentioned there.

In the last part it is shown how some known methods can be obtained from the general algorithm. The reader who is familiar with the notation used can read this part first. It could help him to understand ideas and technique of Part 2. The whole article is closed by some numerical results. The questions of numerical stability and still more general algorithms that presented in this article will be discussed in a forthcoming paper.

1. PRELIMINARIES

Let us consider a boundary value problem of order $2k$ over the interval $[a, b]$. Discretizing this two point boundary value problem we obtain the difference equation

$$\sum_{j=0}^{2k} a_j(x_i)\, y(x_i + jh) = f(x_i) \quad \text{for} \quad x_i = a + ih\,, \quad x_i + 2kh \leqq b\,,$$

where h is the step of the discretization. Discretized boundary conditions can be written in the form

$$\sum_{j=0}^{2k} m_{ij}\, y(a + jh) = w_i \quad \text{and} \quad \sum_{j=0}^{2k} n_{ij}\, y(b - (2k - j)\, h) = z_i$$

for $i = 1(1)\, k$.

Notation. Throughout the paper $i = j(k)\, n$ stands for $i \in \{j, j + k, j + 2k, \ldots, n\}$.

Difference equations together with the boundary conditions form a system of algebraic equations for values of the unknown function y at the points x_i. The matrix of the system is of band form.

We have already mentioned factorization methods for the solution of continuous problems and their analogy with the process of Gaussian elimination for the discretized problem in a special case. Factorization methods are of such type that the boundary condition at one point is transferred to the second point and vice versa to obtain an algebraic equation for the unknown (vector) function at every point of the interval of definition. Discretizing a differential equation we come to a difference equation and this difference equation together with the boundary conditions form a system of equations whose matrix is of band form. So, it is natural to follow the ideas of continuous case also in the discrete case.

Consider the linear difference equations

$$b_1 y_1 + h_1 y_2 = f_1\,, \tag{1.1}$$

$$
\begin{aligned}
a_2 y_1 + b_2 y_2 + h_2 y_3 &= f_2 , \\
&\dots \\
a_{N-1} y_{N-2} + b_{N-1} y_{N-1} + h_{N-1} y_N &= f_{N-1} , \\
a_N y_{N-1} + b_N y_N &= f_N .
\end{aligned}
$$

The first and the last equation of the system (1.1) is the left and the right boundary condition, respectively.

Let us define

$$
c_2 = \frac{-h_1}{b_1} , \quad c_{i+1} = \frac{-h_i}{b_i + c_i a_i} \quad \text{for} \quad i = 2(1) N - 1
$$

provided $b_i + c_i a_i$ is different from zero. Gaussian elimination for the system (1.1) yields

$$
\begin{aligned}
1 \,.\, y_1 - c_2 y_2 &= \frac{f_1}{b_1} , \\
1 \,.\, y_2 - c_3 y_3 &= (f_2 - a_2 d_1) \frac{c_3}{h_2} , \\
&\dots \\
1 \,.\, y_{N-1} - c_N y_N &= (f_{N-1} - a_{N-1} d_{N-2}) \frac{c_N}{h_{N-1}} , \\
a_N y_{N-1} + b_N y_N &= f_N ,
\end{aligned} \tag{1.2}
$$

where

$$
d_1 = \frac{f_1}{b_1} , \quad d_{i+1} = (f_{i+1} - a_{i+1} d_i) \frac{c_{i+2}}{h_{i+1}} \quad \text{for} \quad i = 1(1) N - 2 .
$$

Denoting

$$
D_{i+1,1} = 1 \quad \text{and} \quad D_{i+1,2} = -c_{i+2} \quad \text{for} \quad i = 0(1) N - 2
$$

and

$$
\boldsymbol{x}_i = [y_i, y_{i+1}]^{\mathsf{T}} ,
$$

we can rewrite the system (1.1) in the form

$$
\boldsymbol{D}_i \boldsymbol{x}_i = d_i \quad \text{for} \quad i = 1(1) N - 1 , \tag{1.3}
$$

$$
[a_N, b_N] \, x_{N-1} = f_N , \tag{1.4}
$$

where

$$
\boldsymbol{D}_i = [D_{i,1}, D_{i,2}] .
$$

Let us stop the process of Gaussian elimination at the i-th equation. Thus the system (1.1) is transformed into the form

$$\mathbf{D}_j \mathbf{x}_j = d_j \quad \text{for } j = 1(1)\, i-1,$$

$$\begin{aligned} D_{i,1} y_i + D_{i,2} y_{i+1} &= d_i\,, \\ a_j y_{j-1} + b_j y_j + h_j y_{j+1} &= f_j \quad \text{for} \quad j = i+1(1)\, N-1\,, \\ a_N y_{N-1} + b_N y_N &= f_N\,. \end{aligned} \tag{1.5}$$

The system (1.5) is part of the system (1.1) and the influence of the equations of the system (1.1) up to the i-th equation has been concentrated into the new boundary condition $\mathbf{D}_i \mathbf{x}_i = d_i$. Thus we can see that the forward step of Gaussian elimination is the transfer of the left boundary condition to the right. For the backward step the situation is the same. Namely, we can compute the values y_{N-1} and y_N from the equations (1.3)–(1.4). Thus, having computed y_{i+1} we can compute y_i from the equation $\mathbf{D}_i \mathbf{x}_i = d_i$. Again the knowledge of y_{i+1} enables us to compute all the values y_j for $j = i(-1)\, 1$. Hence the "equation" $y_{i+1} = y_{i+1}$ is the transferred right boundary condition $a_N y_{N-1} + b_N y_N = f_N$.

In the next parts we will extend the ideas to the general case of systems of linear equations with band matrices. And now we can see how natural it is to call the direct methods for solution of such systems the methods of the transfer of conditions. Let us turn to the general case. First we shall define what we mean by the band matrix.

Definition. *Let $\mathbf{G} = (g_{ij})$ be a square matrix of order N and let p be the least integer such that*

$$\text{for all } i, j \in \{1, \ldots, N\}\,, \quad |i-j| > p \text{ implies } g_{ij} = 0\,.$$

The number $2p+1$ is called the bandwidth of the matrix $\mathbf{G}$ and the matrix $\mathbf{G}$ is called a band matrix.

Let $\mathbf{G}$ be a band matrix of order N with a bandwidth $2p+1$ while $\mathbf{b} = [b_1, \ldots, b_N]^\mathsf{T}$ is an N-dimensional vector. We are looking for the solution $\mathbf{y} = [y_1, \ldots, y_N]^\mathsf{T}$ of the system

$$\mathbf{G}\mathbf{y} = \mathbf{b}\,. \tag{1.6}$$

Following the ideas of the example we shall define a system of $2p$-dimensional vectors $\mathbf{x}_i^{(j)}$ analogous to the vectors $\mathbf{x}_i$:

$$\mathbf{x}_i^{(j)} = [y_{(2p-j)(i-1)+1}, \ldots, y_{(2p-j)i+1}]^\mathsf{T} \tag{1.7}$$

for $i = 1(1)\, J+1$, where j is a chosen fixed integer from the closed interval $[0, 2p-1]$ and $J = [(N-2p)/(2p-j)]$ ($[m]$ stands for the integral part of the number m).

Notation. The superscript T will always denote the transpose of a vector. Rank of a matrix $\mathbf{M}$ will be denoted by *rank* $\mathbf{M}$. The symbol $\mathbf{I}_q$ stands for the identity

matrix of order q while $\mathbf{O}_{i,j}$ stands for the null matrix of type $i \times j$, i.e., with i rows and j columns.

Remark. If it is clear which of the values is chosen for j we shall simply write $\boldsymbol{x}_i$ if this cannot lead to any confusion.

Let us note that the definition of the vectors $\boldsymbol{x}_i$ implies that the last j components of $\boldsymbol{x}_i$ repeat as the first j components of $\boldsymbol{x}_{i+1}$ (in the example j was equal to 1).

Assumption $\mathscr{P}$. *For the sake of rather technical then fundamental reasons we shall assume* $J = (N - 2p)/(2p - j)$. *This does not affect the generality of our theory.*

In a similar way as we have divided the vector $\mathbf{y}$ we divide also the matrix $\boldsymbol{G}$ and the vector $\boldsymbol{b}$. Let us denote

$$I = (2p - j)(i - 1).$$

$$\boldsymbol{A}^i = \begin{bmatrix} g_{I+p+1,I+1}, & \dots, & g_{I+p+1,I+2p-j} \\ 0 & \ddots & \vdots \\ \vdots & \ddots & \vdots \\ 0 \dots & & 0, \; g_{I+p+2p-j,I+2p-j} \end{bmatrix},$$

$$\boldsymbol{B}^i = \begin{bmatrix} g_{I+p+1,I+2p-j+1}, & \dots, & g_{I+p+1,I+2p+1}, & 0, & \dots, & 0 \\ \dots & & & & & \\ g_{I+p+2p-j,I+2p-j+1}, & \dots, & & & & g_{I+2p-j+p,I} \end{bmatrix}$$

and

$$\boldsymbol{A}_i = \left[\begin{array}{c|c} \boldsymbol{A}^i & \boldsymbol{O}_{2p-j,j} \\ \hline \boldsymbol{O}_{j,2p-j} & \boldsymbol{I}_j \end{array}\right], \quad \boldsymbol{B}_i = \left[\begin{array}{c} \boldsymbol{B}^i \\ \hline -\boldsymbol{I}_j \;\vert\; \boldsymbol{O}_{j,2p-j} \end{array}\right], \tag{1.8}$$

$$\boldsymbol{f}_i = [b_{I+p+1}, \dots, b_{I+p+2p-j}, 0, \dots, 0]^{\mathsf{T}} \tag{1.9}$$

for $i = 1(1)\, J$, where $\boldsymbol{A}_i$ and $\boldsymbol{B}_i$ are square matrices of order $2p$ and the vectors $\boldsymbol{f}_i$ are $2p$-dimensional. The matrices $\boldsymbol{A}_i$ and $\boldsymbol{B}_i$ are generated by $2p - j$ rows of the matrix $\boldsymbol{G}$ the elements of which aplly to both $\boldsymbol{x}_i$ and $\boldsymbol{x}_{i+1}$, and completed to a square by the identities

$$\begin{aligned} y_{I+2p-j+1} &= y_{I+2p-j+1}, \\ &\dots \\ y_{I+2p} &= y_{I+2p}. \end{aligned}$$

The first p equations of the system (1.6) will be called the left boundary condition and the last the right boundary condition. We rewrite them in the form

$$\boldsymbol{A}_0 \boldsymbol{x}_1 = \boldsymbol{f}_0, \quad \boldsymbol{A}_{J+1} \boldsymbol{x}_{J+1} = \boldsymbol{f}_{J+1}, \tag{1.10}$$

where

$$\boldsymbol{A}_0 = \begin{bmatrix} g_{11}, \ldots, g_{1p}, 0, \ldots, 0 \\ \cdots \\ g_{p1}, \quad \cdots \quad , g_{p,2p} \end{bmatrix},$$

$$\boldsymbol{A}_{J+1} = \begin{bmatrix} g_{N-p+1,N-2p+1}, \quad \cdots \quad , g_{N-p+1,N} \\ 0 \\ \vdots \\ 0, \ldots, 0, g_{N,N-p+1}, \ldots, g_{N,N} \end{bmatrix},$$

$$\boldsymbol{f}_0 = [b_1, \ldots, b_p]^\mathsf{T} \quad \text{and} \quad \boldsymbol{f}_{J+1} = [b_{N-p+1}, \ldots, b_N]^\mathsf{T}.$$

2. GENERAL ALGORITHM

Using the notation (1.7)–(1.10) we can rewrite the system (1.6) in the form

$$\boldsymbol{A}_i\boldsymbol{x}_i + \boldsymbol{B}_i\boldsymbol{x}_{i+1} = \boldsymbol{f}_i \quad \text{for} \quad i = 1(1)\,J\,, \tag{2.1}$$

$$\boldsymbol{A}_0\boldsymbol{x}_1 = \boldsymbol{f}_0\,, \quad \boldsymbol{A}_{J+1}\boldsymbol{x}_{J+1} = \boldsymbol{f}_{J+1}\,.$$

Now we wish to replace the solution of the system (2.1) by a solution of systems

$$\boldsymbol{Q}_i\boldsymbol{x}_i = \boldsymbol{q}_i \quad \text{for} \quad i = 1(1)\,J+1\,, \tag{2.2}$$

where $\boldsymbol{Q}_i$ are regular matrices with $2p$ columns of the block form $\boldsymbol{Q}_i = [\boldsymbol{D}_i^\mathsf{T}, \boldsymbol{R}_i^\mathsf{T}]^\mathsf{T}$ and vectors $\boldsymbol{q}_i$ are of the form $\boldsymbol{q}_i = [\boldsymbol{d}_i^\mathsf{T}, \boldsymbol{r}_i^\mathsf{T}]^\mathsf{T}$. The pairs $\boldsymbol{D}_i$, $\boldsymbol{d}_i$ and $\boldsymbol{R}_i$, $\boldsymbol{r}_i$ are connected with a transfer of the left condition to the right and the right condition to the left, respectively. First we shall describe the transfer to the right. Let us suppose that the system (2.1) has a solution and we have already obtained a matrix $\boldsymbol{D}_i$ and a vector $\boldsymbol{d}_i$ such that

$$\boldsymbol{D}_i\boldsymbol{x}_i = \boldsymbol{d}_i \tag{2.3}$$

and the matrix $\boldsymbol{A}_i$ for this i is regular. Denoting

$$\boldsymbol{H}_i = \boldsymbol{A}_i^{-1}\boldsymbol{B}_i \quad \text{and} \quad \boldsymbol{h}_i = \boldsymbol{A}_i^{-1}\boldsymbol{f}_i$$

we can write for $\boldsymbol{x}_i$ and $\boldsymbol{x}_{i+1}$ the equation

$$\boldsymbol{x}_i + \boldsymbol{H}_i\boldsymbol{x}_{i+1} = \boldsymbol{h}_i\,. \tag{2.4}$$

After multiplying this equation by $\boldsymbol{D}_i$, (2.3) and (2.4) yield

$$\boldsymbol{D}_i\boldsymbol{H}_i\boldsymbol{x}_{i+1} = -\boldsymbol{d}_i + \boldsymbol{D}_i\boldsymbol{h}_i\,.$$

Denoting

$$\boldsymbol{D}_{i+1} = \boldsymbol{Z}_i\boldsymbol{D}_i\boldsymbol{H}_i \quad \text{and} \quad \boldsymbol{d}_{i+1} = \boldsymbol{Z}_i(-\boldsymbol{d}_i + \boldsymbol{D}_i\boldsymbol{h}_i) \tag{2.5}$$

where $\boldsymbol{Z}_i$ is a regular matrix of order equal to the number of the rows of the matrix $\boldsymbol{D}_i$ we can write

$$\boldsymbol{D}_{i+1}\boldsymbol{x}_{i+1} = \boldsymbol{d}_{i+1} \,. \tag{2.6}$$

Equations (2.5) realize the transfer of the equation (2.3) to (2.6). To follow strictly the ideas of the example from the previous part, for the transfer from the right to the left we have to complete the matrices $\boldsymbol{D}_i$ to squares and regular ones by matrices $\boldsymbol{R}_i$ in such a way that

$$\boldsymbol{R}_i\boldsymbol{x}_i = \boldsymbol{r}_i \,,$$

where $\boldsymbol{r}_i$ is an appropriate vector. Again, let us suppose the matrix $\boldsymbol{A}_i$ to be regular and the matrix $\boldsymbol{D}_{i+1}$ to be already completed to the regular matrix $\boldsymbol{Q}_{i+1}$. From the equation

$$\boldsymbol{Q}_{i+1}\boldsymbol{x}_{i+1} = \boldsymbol{q}_{i+1}$$

we can compute the vector $\boldsymbol{x}_{i+1}$. Plugging it into the equation (2.4) and multiplying the resulting equation by the matrix $\boldsymbol{R}_i$ we obtain

$$\boldsymbol{R}_i\boldsymbol{x}_i = \boldsymbol{R}_i(\boldsymbol{h}_i - \boldsymbol{H}_i\boldsymbol{Q}_{i+1}^{-1}\boldsymbol{q}_{i+1}) \,.$$

Denoting

$$\boldsymbol{r}_i = \boldsymbol{R}_i(\boldsymbol{h}_i - \boldsymbol{H}_i\boldsymbol{Q}_{i+1}^{-1}\boldsymbol{q}_{i+1}) \,, \tag{2.7}$$

the vector $\boldsymbol{r}_i$ fulfils the equation

$$\boldsymbol{R}_i\boldsymbol{x}_i = \boldsymbol{r}_i \,.$$

If the matrix $\boldsymbol{A}_i$ is singular this approach does not work. However, the following lemma which we quote without proof (cf. [4]) can serve us as a hint how to solve this case.

Lemma 2.1. *Let* $\boldsymbol{C}_1$ *be an* $a_1 \times n$ *matrix,* rank $\boldsymbol{C}_1 = h_1$ *while* $\boldsymbol{C}_2$ *is an* $a_2 \times n$ *matrix,* rank $\boldsymbol{C}_2 = h_2$ *and* rank $\boldsymbol{C}_3 = h_3$ *where* $\boldsymbol{C}_3 = [\boldsymbol{C}_1^{\mathrm{T}}, \boldsymbol{C}_2^{\mathrm{T}}]^{\mathrm{T}}$. *Then there are matrices* $\boldsymbol{S}_1$ *and* $\boldsymbol{S}_2$ *such that*

(1) $\boldsymbol{S}_1\boldsymbol{C}_1 = \boldsymbol{S}_2\boldsymbol{C}_2$,

(2) *the rank of* $\boldsymbol{S}_1\boldsymbol{C}_1$ *equals the number of its rows and is equal to* $h_1 + h_2 - h_3$ *and for every pair of matrices* $\boldsymbol{S}_1$, $\boldsymbol{S}_2$ *for which* (1) *holds, the rank of the matrix* $\boldsymbol{S}_1\boldsymbol{C}_1$ *is not greater then* $h_1 + h_2 - h_3$.

The following remark also quoted from [4] without proof appears to be useful.

Remark 1. Lemma 2.1 is proved by modifying the matrix $\boldsymbol{C}_3$ to an equivalent matrix $\boldsymbol{M}$ with $h_1 + h_2$ rows and $a_1 + a_2$ columns,

$$\boldsymbol{M} = \begin{bmatrix} \boldsymbol{B}_1 & \boldsymbol{O}_{h_1, a_2} \\ \boldsymbol{O}_{h_3 - h_1, a_1} & \boldsymbol{P}_1 \boldsymbol{B}_2 \\ -\boldsymbol{S}_1 & \boldsymbol{S}_2 \end{bmatrix},$$

where

(i) the matrix $\boldsymbol{B}_1$ is the $h_1 \times a_1$ matrix and consists of all linearly independent rows of $\boldsymbol{C}_1$ while $\boldsymbol{B}_2$ consists of those of $\boldsymbol{C}_2$ and $\boldsymbol{P}_1$ is a permutation matrix such that the first h_3 rows of the matrix

$$\begin{bmatrix} \boldsymbol{B}_1 \boldsymbol{C}_1 \\ \boldsymbol{P}_1 \boldsymbol{B}_2 \boldsymbol{C}_2 \end{bmatrix}$$

are linearly independent. The matrix $\boldsymbol{P}_1 \boldsymbol{B}_2$ is the $(h_3 - h_1) \times a_2$ matrix, $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$ are $(h_1 + h_2 - h_3) \times a_1$, $(h_1 + h_2 - h_3) \times a_2$ matrices, respectively.

(ii) h_3 is equal to the rank of the matrix

$$\begin{bmatrix} \boldsymbol{B}_1 \boldsymbol{C}_1 \\ \boldsymbol{P}_1 \boldsymbol{B}_2 \boldsymbol{C}_2 \\ \boldsymbol{O}_{h_1 + h_2 - h_3, n} \end{bmatrix},$$

(iii)
$$\boldsymbol{M}\boldsymbol{C}_3 = \begin{bmatrix} \boldsymbol{B}_1 \boldsymbol{C}_1 \\ \boldsymbol{P}_1 \boldsymbol{B}_2 \boldsymbol{C}_2 \\ \boldsymbol{O}_{h_1 + h_2 - h_3, n} \end{bmatrix}.$$

Moreover,

$$rank\ \boldsymbol{M} = h_1 + h_2 .$$

Thus, let the matrix $\boldsymbol{A}_i$ be singular. We wish to transfer the equation (2.3) to the "point $i + 1$". Equation (2.1) for this i can be written in an equivalent form

$$\begin{bmatrix} \boldsymbol{A}_{i,1} \\ \boldsymbol{O}_{n_i, 2p} \end{bmatrix} \boldsymbol{x}_i + \begin{bmatrix} \boldsymbol{B}_{i,1} \\ \boldsymbol{B}_{i,2} \end{bmatrix} \boldsymbol{x}_{i+1} = \begin{bmatrix} \boldsymbol{f}_{i,1} \\ \boldsymbol{f}_{i,2} \end{bmatrix}, \tag{2.8}$$

where $rank\ \boldsymbol{A}_i = 2p - n_i = rank\ \boldsymbol{A}_{i,1}$ and $2p - n_i$ is equal to the number of the rows of the matrix $\boldsymbol{A}_{i,1}$ while $\boldsymbol{B}_{i,1}$ and $\boldsymbol{B}_{i,2}$ are $(2p - n_i) \times 2p$ and $n_i \times 2p$ matrices, respectively. We can suppose $rank\ \boldsymbol{B}_{i,2} = n_i$ because by crossing out linearly dependent rows we do not change the set of solutions of the original problem. Lemma 2.1 implies the existence of matrices $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$ such that

$$\boldsymbol{S}_2 \boldsymbol{A}_{i,1} = \boldsymbol{S}_1 \boldsymbol{D}_i .$$

This equation together with (2.8) and (2.3) yields

$$\boldsymbol{D}_{i+1} \boldsymbol{x}_{i+1} = \boldsymbol{d}_{i+1} ,$$

where

$$D_{i+1} = Z_i \begin{bmatrix} S_2 B_{i,1} \\ B_{i,2} \end{bmatrix}, \quad d_{i+1} = Z_i \begin{bmatrix} S_2 f_{i,1} - S_1 d_i \\ f_{i,2} \end{bmatrix}. \tag{2.9}$$

Let us note that Lemma 2.1 guarantees that we transfer the maximal number of equations (2.8) to the point $i + 1$. But from the numerical point of view this is useless if we are not able to determine the values h_1, h_2 and h_3 exactly. The reason is that the process of determination of the rank of a matrix is numerically unstable.

To transfer the right boundary condition to the left we must distinguish three cases. Let the equation $R_{i+1} x_{i+1} = r_{i+1}$ be already obtained, let the matrix A_i be singular and B_i regular. Thus the equation (2.1) can be written in the form

$$B_i^{-1} A_i x_i + x_{i+1} = B_i^{-1} f_i .$$

Hence choosing

$$R_i = W_i R_{i+1} B_i^{-1} A_i , \quad r_i = W_i(-r_{i+1} + R_{i+1} B_i^{-1} f_i)$$

we have

$$R_i x_i = r_i .$$

If the matrix B_i is singular we again rewrite the equation (2.1) into an equivalent form

$$\begin{bmatrix} {}_1A_i \\ {}_2A_i \end{bmatrix} x_i + \begin{bmatrix} O_{t_i,2p} \\ {}_2B_i \end{bmatrix} x_{i+1} = \begin{bmatrix} {}_1f_i \\ {}_2f_i \end{bmatrix}, \tag{2.10}$$

where $rank\ B_i = rank\ {}_2B_i = 2p - t_i$ and ${}_2B_i$ is the $(2p - t_i) \times 2p$ matrix. Lemma 2.1 implies the existence of matrices $S^{(1)}$ and $S^{(2)}$ such that

$$S^{(1)}\ {}_2B_i = S^{(2)} R_{i+1} . \tag{2.11}$$

Multiplying the equation (2.10) by the matrix $S^{(1)}$ and using (2.11) it is easy to define the matrix R_1 and the vector r_i fulfilling the equation $R_i x_i = r_i$.

All what has been done up to now is simply a hint how to define the general algorithm. After its definition we shall prove that it is the algorithm for solution of the system (1.6).

We divide the set $\mathfrak{M} = \{1, \ldots, J + 1\}$ into four parts,

$$\mathfrak{M} = \mathfrak{M}_1 \cup \mathfrak{M}_2 \cup \mathfrak{M}_3 \cup \mathfrak{M}_4 ,$$

where

$$\mathfrak{M}_1 = \{i \in \mathfrak{M} \mid \text{both } A_i \text{ and } B_i \text{ are regular or } i = 1,\ i = J + 1\} ,$$
$$\mathfrak{M}_2 = \{i \in \mathfrak{M} \mid A_i \text{ is singular, } B_i \text{ is regular}\} ,$$
$$\mathfrak{M}_3 = \{i \in \mathfrak{M} \mid A_i \text{ is regular, } B_i \text{ is singular}\} ,$$
$$\mathfrak{M}_4 = \{i \in \mathfrak{M} \mid \text{both } A_i \text{ and } B_i \text{ are singular}\} .$$

Conserving the notation of the previous part we shall define

Algorithm $\mathscr{T}1$.

(a) ***transfer from the left to the right***

for $i \in \mathfrak{M}_1 \cup \mathfrak{M}_3$,

$$\begin{aligned} \boldsymbol{D}_1 &= \boldsymbol{A}_0\,, \quad \boldsymbol{d}_1 = \boldsymbol{f}_0\,, \\ \boldsymbol{D}_{i+1} &= \boldsymbol{Z}_i\boldsymbol{D}_i\boldsymbol{H}_i \quad \text{and} \quad \boldsymbol{d}_{i+1} = \boldsymbol{Z}_i(-\boldsymbol{d}_i + \boldsymbol{D}_i\boldsymbol{h}_i)\,; \end{aligned} \tag{2.12}$$

for $i \in \mathfrak{M}_2 \cup \mathfrak{M}_4$,

$$\boldsymbol{D}_{i+1} = \boldsymbol{Z}_i\begin{bmatrix}\boldsymbol{S}_2\boldsymbol{B}_{i,1}\\ \boldsymbol{B}_{i,2}\end{bmatrix}, \quad \boldsymbol{d}_{i+1} = \boldsymbol{Z}_i\begin{bmatrix}\boldsymbol{S}_2\boldsymbol{f}_{i,1} - \boldsymbol{S}_1\boldsymbol{d}_i\\ \boldsymbol{f}_{i,2}\end{bmatrix} \tag{2.13}$$

where $\boldsymbol{S}_2\boldsymbol{A}_{i,1} = \boldsymbol{S}_1\boldsymbol{D}_i$ and $\boldsymbol{Z}_i$ is a regular matrix.

(b) ***transfer from the right to the left***

for $i \in \mathfrak{M}_i \cup \mathfrak{M}_3$, *the matrix* $\boldsymbol{R}_i$ *is an arbitrary matrix such that* $\boldsymbol{Q}_i = [\boldsymbol{D}_i^{\mathrm{T}}, \boldsymbol{R}_i^{\mathrm{T}}]^{\mathrm{T}}$ *is a square regular matrix and*

$$\boldsymbol{r}_i = \boldsymbol{R}_i(\boldsymbol{h}_i - \boldsymbol{H}_i\boldsymbol{Q}_{i+1}^{-1}\boldsymbol{q}_{i+1})\,, \tag{2.14}$$

where $\boldsymbol{q}_i = [\boldsymbol{d}_i^{\mathrm{T}}, \boldsymbol{r}_i^{\mathrm{T}}]^{\mathrm{T}}$;

for $i = J + 1$,

$$\boldsymbol{r}_{J+1} = \boldsymbol{R}_{J+1}\begin{bmatrix}\boldsymbol{D}_{J+1}\\ \boldsymbol{A}_{J+1}\end{bmatrix}^{-1}\begin{bmatrix}\boldsymbol{d}_{J+1}\\ \boldsymbol{f}_{J+1}\end{bmatrix};$$

for $i \in \mathfrak{M}_2$,

$$\boldsymbol{R}_i = \boldsymbol{W}_i\boldsymbol{R}_{i+1}\boldsymbol{B}_i^{-1}\boldsymbol{A}_i\,, \quad \boldsymbol{r}_i = \boldsymbol{W}_i(-\boldsymbol{r}_i + \boldsymbol{R}_{i+1}\boldsymbol{B}_i^{-1}\boldsymbol{f}_i)\,;$$

for $i \in \mathfrak{M}_4$,

$$\boldsymbol{R}_i = \boldsymbol{W}_i\begin{bmatrix}\boldsymbol{S}^{(1)}\,{}_2\boldsymbol{A}_i\\ {}_1\boldsymbol{A}_i\end{bmatrix}, \quad \boldsymbol{r}_i = \boldsymbol{W}_i\begin{bmatrix}({\boldsymbol{S}^{1)}}\,{}_2\boldsymbol{f}_i - \boldsymbol{S}^{(2)}\boldsymbol{r}_{i+1}\\ {}_1\boldsymbol{f}_i\end{bmatrix} \tag{2.15}$$

where $\boldsymbol{S}^{(1)}\,{}_2\boldsymbol{B}_i = \boldsymbol{S}^{(2)}\boldsymbol{R}_{i+1}$ *and* $\boldsymbol{W}_i$ *is a regular matrix.*

Vectors $\boldsymbol{x}_i$ *for* $i = 1(1)\,J + 1$ *are defined as solutions of the systems*

$$\boldsymbol{Q}_i\boldsymbol{x}_i = \boldsymbol{q}_i \quad \text{for} \quad i = (1)\,J + 1 \tag{2.16}$$

where

$$\boldsymbol{Q}_i = \begin{bmatrix}\boldsymbol{D}_i\\ \boldsymbol{R}_i\end{bmatrix} \quad \text{and} \quad \boldsymbol{q}_i = \begin{bmatrix}\boldsymbol{d}_i\\ \boldsymbol{r}_i\end{bmatrix}.$$

From the definition of *Algorithm* $\mathscr{T}1$ it is easily seen that it is not the only method. Different methods could be obtained by a different choice of the matrices $\boldsymbol{Z}_i$ and $\boldsymbol{W}_i$.

Notation. *The system* $\{\mathbf{x}_i\}_{i=1}^{J+1}$ *of* $2p$*-dimensional vectors* $\mathbf{x}_i$ *defined by* (1.7) *is said to be a solution of the system* (1.6) *iff the vector* $\mathbf{y} = [y_1, \ldots, y_N]^T$ *is a solution of the system* (1.6).

For the proof that solutions of the systems (2.16) are solutions of (2.1) the following lemmas will be useful.

Lemma 2.2. *Let* $i \in \mathfrak{M}_2 \cup \mathfrak{M}_4$ *and let the matrices* $\mathbf{D}_i$ *and* $\mathbf{D}_{i+1}$ *be such that* $\operatorname{rank} \mathbf{D}_i = \operatorname{rank} [\mathbf{D}_i, \mathbf{d}_i]$ *while* $\operatorname{rank} \mathbf{D}_{i+1} = \operatorname{rank} [\mathbf{D}_{i+1}, \mathbf{d}_{i+1}]$ *and let there exist a solution* $\mathbf{z}$ *of the system*

$$\mathbf{D}_{i+1}\mathbf{z} = \mathbf{d}_{i+1} . \tag{2.17}$$

Then the system

$$\begin{bmatrix} \mathbf{D}_i \\ \mathbf{A}_i \end{bmatrix} \mathbf{w} = \begin{bmatrix} \mathbf{d}_i \\ -\mathbf{B}_i\mathbf{z} + \mathbf{f}_i \end{bmatrix} \tag{2.18}$$

has a solution.

Proof. We can suppose that $\operatorname{rank} \mathbf{D}_i$ equal to the number of its rows because by crossing out the linearly dependent rows we do not change the set of solutions of the equation (2.18). This system can be written in an equivalent form

$$\begin{bmatrix} \mathbf{D}_i \\ \mathbf{A}_{i,1} \\ \mathbf{O}_{n_i}, 2p \end{bmatrix} \mathbf{w} = \begin{bmatrix} \mathbf{d}_i \\ -\mathbf{B}_{i,1}\mathbf{z} + \mathbf{f}_{i,1} \\ -\mathbf{B}_{i,2}\mathbf{z} + \mathbf{f}_{i,2} \end{bmatrix} . \tag{2.19}$$

Equation (2.13) implies

$$-\mathbf{B}_{i,2}\mathbf{z} + \mathbf{f}_{i,2} = \mathbf{O}$$

for every solution $\mathbf{z}$ of the system (2.17). Hence the system (2.19) can be reduced to a form

$$\begin{bmatrix} \mathbf{D}_i \\ \mathbf{A}_{i,1} \end{bmatrix} \mathbf{w} = \begin{bmatrix} \mathbf{d}_i \\ -\mathbf{B}_{i,1}\mathbf{z} + \mathbf{f}_{i,1} \end{bmatrix} .$$

Remark 1 implies that this system can be replaced by an equivalent one with a matrix of the system whose rank is equal to the number of its rows, i.e., the system (2.18) has a solution.

Lemma 2.3. *Let* $i \in \mathfrak{M}_2 \cup \mathfrak{M}_4$, *let* $\operatorname{rank} \mathbf{R}_i = \operatorname{rank} [\mathbf{R}_i, \mathbf{r}_i]$ *while* $\operatorname{rank} \mathbf{R}_{i+1} = \operatorname{rank} [\mathbf{R}_{i+1}, \mathbf{r}_{i+1}]$ *and the vector* $\mathbf{z}$ *is a solution of the system*

$$\mathbf{R}_i\mathbf{z} = \mathbf{r}_i .$$

Then the system

$$\begin{bmatrix} \mathbf{R}_{i+1} \\ \mathbf{B}_i \end{bmatrix} \mathbf{w} = \begin{bmatrix} \mathbf{r}_{i+1} \\ -\mathbf{A}_i\mathbf{z} + \mathbf{f}_i \end{bmatrix}$$

has a solution.

Since the proof is almost the same as the previous one we omit it.

Theorem 2.1. *Let us suppose the assumption* $\mathscr{P}$ *to be fulfilled. Then*

(*i*) *every solution* $\{\boldsymbol{x}_i\}_{i=1}^{J+1}$ *of the system* (2.1) *is a solution of the systems* (2.16) *and vice versa*;

(*ii*) *systems* (2.16) *have unique solutions iff the system* (2.1) *has a unique solution*;

(*iii*) *matrices* $\boldsymbol{D}_i$ *and* $[\boldsymbol{D}_i, \boldsymbol{d}_i]$ *can change their ranks only at the points* $i \notin \mathfrak{M}_i$.

Proof. (i) Let $\{\boldsymbol{x}_i\}$ be a solution of the system (2.1). Then

$$\boldsymbol{D}_1\boldsymbol{x}_1 = \boldsymbol{d}_1$$

and the vector $\boldsymbol{x}_1$ fulfils also the equation

$$\boldsymbol{A}_1\boldsymbol{x}_1 + \boldsymbol{B}_1\boldsymbol{x}_2 = \boldsymbol{f}_1 .$$

Let number 1 be an element of $\mathfrak{M}_1 \cup \mathfrak{M}_3$. Then these equations imply

$$\boldsymbol{D}_1\boldsymbol{x}_1 + \boldsymbol{D}_1\boldsymbol{H}_1\boldsymbol{x}_2 = \boldsymbol{D}_1\boldsymbol{h}_1$$

and

$$\boldsymbol{D}_1\boldsymbol{x}_1 - \boldsymbol{d}_1 = (-\boldsymbol{d}_1 + \boldsymbol{D}_1\boldsymbol{h}_1) - \boldsymbol{D}_1\boldsymbol{H}_1\boldsymbol{x}_2 .$$

Equivalently,

$$\boldsymbol{Z}_1\boldsymbol{c}_1 = -\boldsymbol{c}_2 ,$$

where

$$\boldsymbol{c}_i = \boldsymbol{D}_i\boldsymbol{x}_i - \boldsymbol{d}_i .$$

If $1 \in \mathfrak{M}_2 \cup \mathfrak{M}_4$ then equations (2.13) imply again

$$\boldsymbol{Z}_1\boldsymbol{c}_1 = -\boldsymbol{c}_2$$

where $\boldsymbol{Z}_1$ is a regular matrix of an appropriate order. Thus we can conclude that

$$\boldsymbol{Z}_i\boldsymbol{c}_i = -\boldsymbol{c}_{i+1} \quad \text{for} \quad i = 1(1)\,J$$

where $\boldsymbol{Z}_i$ are regular matrices. But $\boldsymbol{c}_1 = \boldsymbol{0}$, hence $\boldsymbol{c}_i = \boldsymbol{0}$ for all i. For the matrices $\boldsymbol{R}_i$ and vectors $\boldsymbol{r}_i$, the proof is the same. Conversely, let $\{\boldsymbol{x}_i\}$ be a solution of the systems (2.16). It is sufficient to prove that from every solution of the system $\boldsymbol{Q}_i\boldsymbol{x}_i = \boldsymbol{q}_i$ we can construct a complete solution of the system (2.1) which consists of solutions of the systems (2.16). Let us prove this assertion. First, let $\boldsymbol{x}_k$ be a solution of the system (2.16) for $i = k$. We shall construct the solutions $\boldsymbol{x}_i$ of both the systems (2.1) and (2.16) for $i < k$. Let us suppose that $i \in \mathfrak{M}_1 \cup \mathfrak{M}_3$ and that $\boldsymbol{x}_k$ is a solution of the system $\boldsymbol{Q}_k\boldsymbol{x}_k = \boldsymbol{q}_k$. Then the system

$$\boldsymbol{A}_{k-1}\boldsymbol{z} = -\boldsymbol{B}_{k-1}\boldsymbol{x}_k + \boldsymbol{f}_{k-1} \tag{2.20}$$

has a unique solution and we denote it by $\boldsymbol{x}_{k-1}$. Then

$$\boldsymbol{x}_{k-1} = -\boldsymbol{H}_{k-1}\boldsymbol{x}_k + \boldsymbol{h}_{k-1}$$

and

$$\boldsymbol{D}_{k-1}\boldsymbol{x}_{k-1} = -\boldsymbol{D}_{k-1}\boldsymbol{H}_{k-1}\boldsymbol{x}_k + \boldsymbol{D}_{k-1}\boldsymbol{h}_{k-1},$$

where we have used the equation (2.12). In a similar way one could show that

$$\boldsymbol{R}_{k-1}\boldsymbol{x}_{k-1} = \boldsymbol{r}_{k-1},$$

i.e., the vector $\boldsymbol{x}_{k-1}$ is the solution of both the equation (2.20) and the system (2.16) for $i = k - 1$.

For $i \in \mathfrak{M}_2 \cup \mathfrak{M}_4$ the matrix $\boldsymbol{A}_{k-1}$ is singular. Lemma 2.2 implies that there is a solution $\mathbf{z}$ of the system

$$\begin{bmatrix} \boldsymbol{D}_{k-1} \\ \boldsymbol{A}_{k-1} \end{bmatrix} \mathbf{z} = \begin{bmatrix} \boldsymbol{d}_{k-1} \\ -\boldsymbol{B}_{k-1}\boldsymbol{x}_k + \boldsymbol{f}_{k-1} \end{bmatrix}$$

which we denote by $\boldsymbol{x}_{k-1}$. Directly from the definition of the matrix $\boldsymbol{R}_{k-1}$ and of the vector $\boldsymbol{r}_{k-1}$ in *Algorithm* $\mathscr{T}1$ we obtain that this $\boldsymbol{x}_{k-1}$ satisfies

$$\boldsymbol{R}_{k-1}\boldsymbol{x}_{k-1} = \boldsymbol{r}_{k-1}.$$

In each case we can construct in this way vectors $\boldsymbol{x}_i$ for $i < k$, such that

$$\boldsymbol{Q}_i\boldsymbol{x}_i = \boldsymbol{q}_i \quad \text{and} \quad \boldsymbol{A}_i\boldsymbol{x}_i + \boldsymbol{B}_i\boldsymbol{x}_{i+1} = \boldsymbol{f}_i.$$

Now we shall construct solutions $\boldsymbol{x}_j$ for $j > k$ from the vector $\boldsymbol{x}_k$. First, let us suppose $i \in \mathfrak{M}_1 \cup \mathfrak{M}_3$. Then (2.14) yields

$$\boldsymbol{r}_i = \boldsymbol{R}_i(\boldsymbol{h}_i - \boldsymbol{H}_i\boldsymbol{x}_{i+1})$$

where $\boldsymbol{x}_{i+1}$ is a solution of the system (2.16) for the index $i + 1$. For the vector $\boldsymbol{d}_i$ we have the equation

$$\boldsymbol{d}_i = -\boldsymbol{Z}_i^{-1}\boldsymbol{d}_{i+1} + \boldsymbol{D}_i\boldsymbol{h}_i.$$

But

$$\boldsymbol{d}_{i+1} = \boldsymbol{D}_{i+1}\boldsymbol{x}_{i+1},$$

$$\boldsymbol{Z}_i^{-1}\boldsymbol{D}_{i+1} = \boldsymbol{D}_i\boldsymbol{H}_i,$$

hence

$$\boldsymbol{Q}_i = \boldsymbol{q}_i(\boldsymbol{h}_i - \boldsymbol{H}_i\boldsymbol{x}_{i+1}).$$

It means that $\boldsymbol{h}_i - \boldsymbol{H}_i\boldsymbol{x}_{i+1}$ is a unique solution of the system $\boldsymbol{Q}_i\boldsymbol{x}_i = \boldsymbol{q}_i$. We shall denote it by $\boldsymbol{x}_i$.

Let $i \in \mathfrak{M}_2 \cup \mathfrak{M}_4$, i.e., the matrix $\boldsymbol{A}_i$ is singular and $\boldsymbol{x}_i$ is a solution of the system $\boldsymbol{Q}_i\boldsymbol{x}_i = \boldsymbol{q}_i$. Then Lemma 2.3 implies the existence of solutions $\boldsymbol{w}$,

$$\begin{bmatrix} \boldsymbol{R}_{i+1} \\ \boldsymbol{B}_i \end{bmatrix} \boldsymbol{w} = \begin{bmatrix} \boldsymbol{r}_{i+1} \\ -\boldsymbol{A}_i\boldsymbol{x}_i + \boldsymbol{f}_i \end{bmatrix}. \tag{2.21}$$

We choose one of them and denote it by $\boldsymbol{x}_{i+1}$. It is sufficient to show that

$$\boldsymbol{D}_{i+1}\boldsymbol{x}_{i+1} = \boldsymbol{d}_{i+1}\,.$$

Equations (2.21) and (2.8) imply

$$-\begin{bmatrix} \boldsymbol{A}_{i,1} \\ \boldsymbol{0} \end{bmatrix} \boldsymbol{x}_i + \begin{bmatrix} \boldsymbol{f}_{i,1} \\ \boldsymbol{f}_{i,2} \end{bmatrix} = \begin{bmatrix} \boldsymbol{B}_{i,1} \\ \boldsymbol{B}_{i,2} \end{bmatrix} \boldsymbol{x}_{i+1}$$

and

$$\boldsymbol{D}_{i+1}\boldsymbol{x}_{i+1} = \boldsymbol{d}_{i+1}\,.$$

Thus we have constructed a solution of the system (2.1) from the solutions of the systems (2.16).

(ii) Again, let us suppose that the system (2.1) has a unique solution but there is an index k such that the solution of the system $\boldsymbol{Q}_k\boldsymbol{x} = \boldsymbol{q}_k$ is not unique. As we have already proved, every solution of the original problem is a solution of the systems (2.16). Therefore the system (2.16) for $i = k$ has infinitely many solutions. From each of its solutions we can construct a solution of the original problem in the same way as we have done above. And this is a contradiction with the uniqueness of solution of the original problem. Similarly, if the systems (2.16) have unique solutions, point (i) implies that the same holds for the original problem.

(iii) If $i \in \mathfrak{M}_1$ then the equations (2.12) imply

$$[\boldsymbol{D}_{i+1}, \boldsymbol{d}_{i+1}] = \boldsymbol{Z}_i[\boldsymbol{D}_i, \boldsymbol{d}_i]\begin{bmatrix} \boldsymbol{H}_i & \boldsymbol{h}_i \\ 0 & -1 \end{bmatrix}$$

where both the matrices $\boldsymbol{Z}_i$ and

$$\begin{bmatrix} \boldsymbol{H}_i & \boldsymbol{h}_i \\ 0 & -1 \end{bmatrix}$$

are regular.

Remark 3. The theorem just proved implies that under the assumption that the original problem has a unique solution,

$$\operatorname{rank} \boldsymbol{Q}_i = \operatorname{rank} [\boldsymbol{Q}_i, \boldsymbol{q}_i] = 2p\,.$$

Remark 4. Matrices $\boldsymbol{R}_i$ and vectors $\boldsymbol{r}_i$ can be chosen in a special way, namely, setting

$$\boldsymbol{R}_i = \boldsymbol{W}_i\boldsymbol{R}_{i+1}\boldsymbol{B}_i^{-1}\boldsymbol{A}_i\,, \quad \boldsymbol{r}_i = \boldsymbol{W}_i(-\boldsymbol{r}_{i+1} + \boldsymbol{R}_{i+1}\boldsymbol{B}_i^{-1}\boldsymbol{f}_i)$$

for $i \in \mathfrak{M}_1 \cup \mathfrak{M}_2$ and defining them by the equations (2.15) for $i \in \mathfrak{M}_3 \cup \mathfrak{M}_4$.

Finally, we should like to mention a subgroup of *Algorithm* $\mathscr{T}1$ where we can compute matrices $\boldsymbol{R}_i$ and vectors $\boldsymbol{r}_i$ together with $\boldsymbol{D}_i$ and $\boldsymbol{d}_i$ provided $\mathfrak{M}_1 = \mathfrak{M}$. One of the possibilities is to ask that for two different initial values, say $\boldsymbol{r}_i$ and $\boldsymbol{s}_1$, all the vectors differ only by a constant. It means

$$\text{const} = \boldsymbol{r}_i - \boldsymbol{s}_i = -\boldsymbol{R}_i\boldsymbol{H}_i\boldsymbol{Q}_{i+1}^{-1}\begin{bmatrix}\boldsymbol{O}_{p,1}\\ \boldsymbol{r}_{i+1} - \boldsymbol{s}_{i+1}\end{bmatrix},$$

i.e.,

$$\boldsymbol{R}_i\boldsymbol{H}_i\boldsymbol{Q}_{i+1}^{-1} = [\boldsymbol{T}_i, -\boldsymbol{I}_p],$$

where $\boldsymbol{T}_i$ is an arbitrary square matrix of order p. Hence

$$\boldsymbol{R}_{i+1} = \boldsymbol{T}_i\boldsymbol{D}_{i+1} - \boldsymbol{R}_i\boldsymbol{H}_i .$$

For $i = J + 1$ we have

$$\text{const} = \boldsymbol{r}_{J+1} - \boldsymbol{s}_{J+1}$$

where

$$\boldsymbol{r}_{J+1} = \boldsymbol{R}_{J+1}\begin{bmatrix}\boldsymbol{D}_{J+1}\\ \boldsymbol{A}_{J+1}\end{bmatrix}^{-1}\begin{bmatrix}\boldsymbol{d}_{J+1}\\ \boldsymbol{f}_{J+1}\end{bmatrix}.$$

Conserving the notation of the previous parts we shall define

Algorithm $\mathscr{T}3$.

(i) *for* $i = 1$ *we set* $\boldsymbol{D}_1 = \boldsymbol{A}_0$, $\boldsymbol{d}_1 = \boldsymbol{f}_0$, $\boldsymbol{s}_1$ *is a p-dimensional vector and* $\boldsymbol{R}_1$ *is such a matrix that* $\boldsymbol{Q}_1$ *is a regular matrix*;

$$\boldsymbol{D}_{i+1} = \boldsymbol{Z}_i\boldsymbol{D}_i\boldsymbol{H}_i , \qquad \boldsymbol{d}_{i+1} = \boldsymbol{Z}_i(-\boldsymbol{d}_i + \boldsymbol{D}_i\boldsymbol{h}_i) ,$$

$$\boldsymbol{R}_{i+1} = \boldsymbol{T}_i\boldsymbol{D}_{i+1} - \boldsymbol{R}_i\boldsymbol{H}_i , \quad \boldsymbol{s}_{i+1} = -\boldsymbol{s}_i + \boldsymbol{T}_i\boldsymbol{d}_{i+1} + \boldsymbol{R}_i\boldsymbol{h}_i$$

for $i = 1(1)\, J + 1$.

We set

$$\boldsymbol{Q}_i = \begin{bmatrix}\boldsymbol{D}_i\\ \boldsymbol{R}_i\end{bmatrix}, \quad \boldsymbol{q}_i = \begin{bmatrix}\boldsymbol{d}_i\\ \boldsymbol{s}_i\end{bmatrix},$$

(ii) $$\boldsymbol{x}_i = \boldsymbol{Q}_i^{-1}(\boldsymbol{q}_i + \text{const}) \quad \text{for} \quad i = 1(1)\, J + 1 .$$

3. EXAMPLES

The first example of methods which come within the frame of the *Algorithm* $\mathscr{T}1$ is the so called "*driving-through*" *algorithm* (cf. [3]). Let the following system of equations with a block tridiagonal matrix $\boldsymbol{G}$ be given:

(3.1) $$\boldsymbol{C}_i\boldsymbol{Y}_{i-1} - \boldsymbol{B}_i\boldsymbol{Y}_i + \boldsymbol{A}_i\boldsymbol{Y}_{i+1} = -\boldsymbol{F}_i \quad \text{for} \quad i = 1(1)\, N - 1 ,$$

with boundary conditions

$$-\boldsymbol{B}_0\boldsymbol{Y}_0 + \boldsymbol{A}_0\boldsymbol{Y}_1 = -\boldsymbol{F}_0\,, \tag{3.2}$$

$$\boldsymbol{C}_N\boldsymbol{Y}_{N-1} + \boldsymbol{B}_N\boldsymbol{Y}_N = -\boldsymbol{F}_N$$

where $\boldsymbol{C}_i$, $\boldsymbol{B}_i$ and $\boldsymbol{A}_i$ are square matrices and both $\boldsymbol{B}_i$ and $\boldsymbol{C}_i$ are regular for all values of i.

The solution of the system (3.1)–(3.2) is given (cf. [3]) by

$$\boldsymbol{Y}_{i-1} = \boldsymbol{X}_i\boldsymbol{Y}_i + \boldsymbol{K}_i \quad \text{for} \quad i = 1(1)\,N\,, \tag{3.3}$$

where

$$\begin{aligned} \boldsymbol{X}_{i+1} &= (\boldsymbol{B}_i - \boldsymbol{C}_i\boldsymbol{X}_i)^{-1}\,\boldsymbol{A}_i\,, \\ \boldsymbol{K}_{i+1} &= (\boldsymbol{B}_i - \boldsymbol{C}_i\boldsymbol{X}_i)^{-1}\,(\boldsymbol{F}_i + \boldsymbol{C}_i\boldsymbol{K}_i)\,, \\ \boldsymbol{X}_1 &= \boldsymbol{B}_0^{-1}\boldsymbol{A}_0\,, \quad \boldsymbol{K}_1 = \boldsymbol{B}_0^{-1}\boldsymbol{F}_0\,. \end{aligned} \tag{3.4}$$

In the notation of the previous part p is equal to one and we choose $j = 1$, i.e., $J = N - 1$, $\mathfrak{M} = \mathfrak{M}_1$ and

$$\boldsymbol{x}_i^{(1)} = [\boldsymbol{Y}_{i-1}^{\mathsf{T}}, \boldsymbol{Y}_i^{\mathsf{T}}]^{\mathsf{T}}\,,$$

$$\boldsymbol{H}_i = \begin{bmatrix} -\boldsymbol{C}_i^{-1}\boldsymbol{B}_i & \boldsymbol{C}_i^{-1}\boldsymbol{A}_i \\ -\boldsymbol{I}_t & \boldsymbol{O}_{t,t} \end{bmatrix}, \quad \boldsymbol{h}_i = \begin{bmatrix} -\boldsymbol{C}_i^{-1}\boldsymbol{F}_i \\ \boldsymbol{O}_{t,1} \end{bmatrix}$$

for $i = 1(1)\,J + 1$ under the assumption that all the matrices needed are regular. Here t is the order of matrices $\boldsymbol{A}_i$, $\boldsymbol{B}_i$, $\boldsymbol{C}_i$. Let us denote $\boldsymbol{D}_i = [\boldsymbol{D}_{i,1}, \boldsymbol{D}_{i,2}]$ where $\boldsymbol{D}_{i,1}$ and $\boldsymbol{D}_{i,2}$ are square matrices of order t. Matrices $\boldsymbol{Z}_i$ are chosen so that

$$\boldsymbol{Z}_i = -(\boldsymbol{B}_i + \boldsymbol{C}_i\boldsymbol{D}_{i,2})^{-1}\,\boldsymbol{C}_i \quad \text{for} \quad i = 1(1)\,N - 1$$

provided that $\boldsymbol{B}_i + \boldsymbol{C}_i\boldsymbol{D}_{i,2}$ is regular. (For $\boldsymbol{G}$ a positive definite matrix this is true.)

Hence

$$\begin{aligned} \boldsymbol{D}_{i+1,1} &= \boldsymbol{I}_t\,, \\ \boldsymbol{D}_{i+1,2} &= -(\boldsymbol{B}_i + \boldsymbol{C}_i\boldsymbol{D}_{i,2})^{-1}\,\boldsymbol{A}_i\,, \\ \boldsymbol{d}_{i+1} &= (\boldsymbol{B}_i + \boldsymbol{C}_i\boldsymbol{D}_{i,2})^{-1}\,(\boldsymbol{F}_i + \boldsymbol{C}_i\boldsymbol{d}_i) \end{aligned} \tag{3.5}$$

for $i = 1(1)\,J$.

The left boundary condition is

$$\boldsymbol{D}_{1,1} = \boldsymbol{I}_t\,, \quad \boldsymbol{D}_{1,2} = -\boldsymbol{B}_0^{-1}\boldsymbol{A}_0\,, \quad \boldsymbol{d}_1 = \boldsymbol{B}_0^{-1}\boldsymbol{F}_0\,. \tag{3.6}$$

For $\boldsymbol{x}_{J+1}$ we have the system

$$\begin{bmatrix} \boldsymbol{I}_t & \boldsymbol{D}_{J+1,2} \\ \boldsymbol{C}_N & -\boldsymbol{B}_N \end{bmatrix} \boldsymbol{x}_N = \begin{bmatrix} \boldsymbol{d}_N \\ -\boldsymbol{F}_N \end{bmatrix},$$

i.e.,

$$\boldsymbol{Y}_N = (\boldsymbol{B}_N + \boldsymbol{C}_N\boldsymbol{D}_{N,2})^{-1}(\boldsymbol{F}_N + \boldsymbol{C}_N\boldsymbol{d}_N)\,. \tag{3.7}$$

The matrices $\boldsymbol{R}_i$ can be chosen under our regularity assumptions so that

$$\boldsymbol{R}_i = [\boldsymbol{O}_{t,t}, \boldsymbol{I}_t]\,.$$

Then

$$\boldsymbol{r}_i = -\boldsymbol{D}_{i+1,2}\boldsymbol{r}_{i+1} + \boldsymbol{d}_{i+1}\,. \tag{3.8}$$

The matrices $\boldsymbol{Q}_i$ are of the form

$$\boldsymbol{Q}_i = \begin{bmatrix} \boldsymbol{I}_t & \boldsymbol{D}_{i,2} \\ \boldsymbol{O}_{t,t} & \boldsymbol{I}_t \end{bmatrix} \tag{3.9}$$

and the vectors $\boldsymbol{x}_i$ are solutions of the systems $\boldsymbol{Q}_i\boldsymbol{x} = \boldsymbol{q}_i$. Thus (3.9) implies

$$\boldsymbol{r}_i = \boldsymbol{Y}_i$$

and we have obtained for the vectors $\boldsymbol{Y}_i$

$$\boldsymbol{Y}_i = -\boldsymbol{D}_{i+1,2}\boldsymbol{Y}_{i+1} + \boldsymbol{d}_{i+1}\,. \tag{3.10}$$

Setting $\boldsymbol{X}_i = -\boldsymbol{D}_{i,2}$, the equation (3.10) together with (3.5)–(3.7) are just the equations (3.3)–(3.4).

As the second example we should like to mention the methods from [2]. They are methods for inversion of a tridiagonal symmetric matrix $\boldsymbol{G}$:

$$\boldsymbol{G} = \begin{bmatrix} c_1 & a_2 & & & \boldsymbol{\Theta} \\ a_2 & c_2 & a_3 & & \\ \cdots & & & & \\ \boldsymbol{\Theta} & a_{n-1} & c_{n-1} & a_n & \\ & & a_n & c_n & \end{bmatrix}$$

and

$$\boldsymbol{b} = [b_1, \ldots, b_n]^{\mathsf{T}}\,, \quad \boldsymbol{y} = [y_1, \ldots, y_n]^{\mathsf{T}}$$

where

$$\boldsymbol{G}\boldsymbol{y} = \boldsymbol{b}\,.$$

We choose $j = 1$, i.e.,

$$\boldsymbol{x}_i = [y_i, y_{i+1}]^{\mathsf{T}} \quad \text{for} \quad i = 1(1)\,J + 1$$

and $J = n - 2$ because $p = 1$ so that $J = (n - 2)/(2 - 1)$.

Further,

$$\boldsymbol{A}_i = \begin{bmatrix} a_{i+1} & 0 \\ 0 & 1 \end{bmatrix}, \quad \boldsymbol{B}_i = \begin{bmatrix} c_{i+1} & a_{i+2} \\ -1 & 0 \end{bmatrix}, \quad \boldsymbol{f}_i = \begin{bmatrix} b_{i+1} \\ 0 \end{bmatrix}$$

for $i = 1(1)\, J + 1$ and

$$\boldsymbol{A}_0 = [1,\, a_2/c_1]\,, \quad \boldsymbol{A}_n = [1,\, c_n/a_n]\,,$$
$$\boldsymbol{f}_0 = b_1/c_1\,, \quad \boldsymbol{f}_n = b_n/a_n$$

provided that all divisions are possible. Matrices $\boldsymbol{Z}_i$, i.e., the concrete method of *Algorithm* $\mathscr{T}1$ are chosen so that the matrices $\boldsymbol{D}_i = [D_{i,1},\, D_{i,2}]$ are of the form

$$\boldsymbol{D}_i = [1,\, D^{(i)}]\,.$$

Thus

$$Z_i = \frac{a_{i+1}}{c_{i+1} - D^{(i)} a_{i+1}}\,, \quad \boldsymbol{Z}_i = (Z_i)\,.$$

Then

$$D^{(i+1)} = \frac{a_{i+2}}{c_{i+1} - D^{(i)} a_{i+1}}\,, \quad D^{(1)} = \frac{a_2}{c_1}\,,$$

$$d_{i+1} = \frac{b_{i+1} - a_{i+1} d_i}{c_{i+1} - D^{(i)} a_{i+1}}\,, \quad d_1 = \frac{b_1}{c_1}$$

for $i = 1(1)\, n - 2$.

Matrices $\boldsymbol{R}_i$ and vectors $\boldsymbol{r}_i$ are chosen so that $\boldsymbol{R}_i = [0,\, 1]$ and $\boldsymbol{r}_i = [y_{i+1}]$.

Let us define the quantities C_i:

$$C_1 = 0\,, \quad C_{i+1} = \frac{-a_{i+1}}{c_i + C_i a_i} \quad \text{for} \quad i = 1(1)\, n - 1\,. \tag{3.11}$$

Then

$$D^{(i)} = -C_{i+1}$$

and

$$d_{i+1} = a_{i+1} C_{i+2} \cdot \frac{d_i}{a_{i+2}} - C_{i+2} \cdot \frac{b_{i+1}}{a_{i+2}}\,.$$

For $\boldsymbol{x}_{n-1}$ we have

$$\begin{bmatrix} 1 & -C_n \\ a_n & c_n \end{bmatrix} \boldsymbol{x}_{n-1} = \begin{bmatrix} d_{n-1} \\ b_n \end{bmatrix},$$

hence

$$\boldsymbol{x}_{n-1} = \alpha \begin{bmatrix} c_n & C_n \\ -a_n & 1 \end{bmatrix} \begin{bmatrix} d_{n-1} \\ b_n \end{bmatrix}$$

where α stands for $(c_n + a_n C_n)^{-1}$. Recurrence for d_i implies

$$d_i = a_2 C_{i+1} \ldots C_3 \frac{b_1}{a_{i+1} c_1} - \sum_{j=1}^{i} \prod_{t=j+1}^{i+1} \frac{C_t b_j}{a_{i+1}} =$$

$$= -C_{i+1} \dots C_2 \frac{b_1}{a_{i+1}} - \sum_{j=1}^{i} \prod_{t=j+1}^{i+1} \frac{C_t b_j}{a_{i+1}} .$$

Then for y_n we have the equation

$$y_n = \alpha(b_n - a_n d_{n-1}) = \alpha\Big(b_n + \prod_{t=2}^{n} C_t b_1 + \sum_{j=2}^{n-1} \prod_{t=j+1}^{n} C_t b_j\Big) .$$

Denoting

$$V_i = \prod_{t=i+1}^{n} C_t \quad \text{for} \quad i = 1(1)\, n-1 , \tag{3.12}$$

this equation can be rewritten in the form

$$y_n = \alpha \sum_{i=1}^{n} V_i b_i .$$

By means of the quantities V_i we can rewrite the system $\mathbf{Q}_i \mathbf{x}_i = \mathbf{q}_i$ to

$$y_i = d_i + C_{i+1} y_{i+1} = -\sum_{j=1}^{i+1} \prod_{t=j+1}^{i+1} C_t \frac{b_j}{a_{i+1}} + C_{i+1} y_{i+1} ,$$

i.e.,

$$y_i = -\sum_{j=1}^{i+1} \prod_{t=j+1}^{i+1} C_t \cdot \frac{b_j}{a_{i+1}} - C_{i+1} \sum_{j=1}^{i+2} \prod_{t=j+2}^{i+2} C_t \cdot \frac{b_j}{a_{i+2}} -$$

$$- C_{i+1} \dots C_{n-1} \sum_{j=1}^{n} \prod_{t=j+1}^{n} C_t \frac{b_j}{a_n} + \alpha C_{i+1} \dots C_n \sum_{i=1}^{n} V_i b_i =$$

$$= \alpha V_i b_n + \left(\alpha V_i V_{n-1} - \frac{V_i}{a_n}\right) b_{n-1} +$$

$$+ \left(\alpha V_i V_{n-2} - V_i \frac{C_{n-1}}{a_n} - V_i \frac{C_n}{a_{n-1}}\right) b_{n-2} +$$

$$+ \dots + \left(\alpha V_i V_1 - V_i C_2 \dots C_{n-2} \frac{C_{n-1}}{a_n} - V_i C_2 \dots C_{n-2} \cdot \frac{1}{C_n a_n} -\right.$$

$$\left. - \dots - V_i C_2 \dots C_{i-1} \cdot \frac{1}{C_{i+1} \dots C_n a_{i+1}}\right) b_1 .$$

Let us denote

$$W_j = V_j \left(\alpha - \sum_{k=j+1}^{n} \frac{C_k}{a_k V_{k-1}^2}\right) \quad \text{for} \quad j = 1(1)\, n .$$

This recurrence implies

$$W_j = V_{j+1}^{-1} \left(V_j W_{j+1} - \frac{1}{a_{j+1}}\right) \quad \text{for} \quad j = n-1(-1)\, 1 , \tag{3.13}$$

$$W_n = \alpha . \tag{3.14}$$

Choosing special vectors $\boldsymbol{b}$, namely, $\boldsymbol{b} = \mathbf{e}_i$ for $i = 1(1)n$, where $\mathbf{e}_i = [0, 0, \ldots, 0, 1, 0, \ldots, 0]^T$ and 1 is placed at the i-th position, we can write for the elements k_{ij} of the matrix $\boldsymbol{G}^{-1}$ in virtue of the equation for y_i

$$k_{ij} = V_i W_j \quad \text{for} \quad i \leqq j \,. \tag{3.15}$$

Thus (3.15) together with equations (3.11)–(3.14) is the first of the algorithms from [2].

Realizing that

$$\boldsymbol{G} \,.\, \boldsymbol{G}^{-1} = \boldsymbol{I}_n$$

we can write for the product of the j-th row of the matrix $\boldsymbol{G}$ and the $(j+1)$-st column of the matrix $\boldsymbol{G}^{-1}$:

$$a_j V_{j-1} W_{j+1} + c_j V_j W_{j+1} + a_{j+1} V_{j+1} W_{j+1} = 0 \,,$$

i.e.,

$$V_{j+1} = -(c_j V_j + a_j V_{j-1})/a_{j+1} \quad \text{for} \quad j = 1(1)n - 1$$

and

$$V_0 = 0 \,, \quad V_1 = -\frac{1}{W_0} \,.$$

Similarly

$$\boldsymbol{G}^{-1} \,.\, \boldsymbol{G} = \boldsymbol{I}_n \,,$$

i.e.,

$$V_i W_{j-2} a_{j-1} + V_i W_{j-1} c_j + V_i W_j a_j = 0 \,.$$

Then

$$W_{j-2} = \frac{-c_{j-1} W_{j-1} - a_j W_j}{a_{j-1}}$$

for $j = n + 1(-1)2$,

$$W_n = (-1)^n \,, \quad W_{n+1} = 0 \,,$$

and this is the second algorithm from [2].

We have tested our theory on a "model example", namely, for $\boldsymbol{G}$ being the tridiagonal matrix

$$\boldsymbol{G} = \begin{bmatrix} 1 & 0 & & & \\ 1 & -2 & 1 & & \\ 0 & 1 & -2 & 1 & \\ \ldots & & & & \\ & & 1 & -2 & 1 \\ & & & 0 & 1 \end{bmatrix}, \quad \boldsymbol{b} = -2h \begin{bmatrix} 0 \\ 1 \\ . \\ . \\ 1 \\ 0 \end{bmatrix},$$

where h is a constant.

The system

$$\boldsymbol{G}\boldsymbol{y} = \boldsymbol{b}$$

has been solved by three methods of *Algorithms* $\mathscr{T}1$ and $\mathscr{T}3$. First of them was the method of *Algorithm* $\mathscr{T}1$ with $j = 0$, i.e.,

$$\boldsymbol{x}_i = [y_{2i-1}, y_{2i}]^{\mathsf{T}}$$

and

$$\boldsymbol{H}_i = \boldsymbol{A}_i^{-1}\boldsymbol{B}_i = \begin{bmatrix} 1 & -2 \\ 0 & 1 \end{bmatrix}^{-1} \begin{bmatrix} 1 & 0 \\ -2 & 1 \end{bmatrix} = \begin{bmatrix} -3 & 2 \\ -2 & 1 \end{bmatrix},$$

$$\boldsymbol{h}_i = -2h \begin{bmatrix} 3 \\ 1 \end{bmatrix},$$

$$Z_i = \frac{-1}{3 + 2 \,.\, D_{i,2}}, \quad \boldsymbol{Z}_i = [Z_i] \,.$$

Thus

$$\begin{aligned} \boldsymbol{D}_{i+1} &= [D_{i+1,1}, D_{i+1,2}] = Z_i[-3D_{i,1} - 2D_{i,2}, 2D_{i,1} + D_{i,2}], \\ \boldsymbol{D}_1 &= [1, 0], \\ d_{i+1} &= Z_i(-d_i - 2h \,.\, (3 \,.\, D_{i,1} + D_{i,2})), \\ d_1 &= 0 \,. \end{aligned}$$

With our choice of matrices $\boldsymbol{Z}_i$ we have

$$D_{i,1} = 1 \quad \text{for every } i \,.$$

For the transfer from the right to the left we choose matrices $\boldsymbol{R}_i$ as in Remark 4:

$$\boldsymbol{R}_i = \boldsymbol{W}_i \boldsymbol{R}_{i+1} \boldsymbol{B}_i^{-1} \boldsymbol{A}_i \,,$$

i.e.,

$$\begin{aligned} \boldsymbol{R}_i &= [R_{i,1}, R_{i,2}] = W_i[R_{i+1,1} + 2 \,.\, R_{i+1,2}, -2R_{i+1,1} - 3R_{i+1,2}], \\ \boldsymbol{R}_N &= [0, 1], \\ r_i &= W_i(-r_{i+1} - 2h \,.\, (3 \,.\, R_{i+1,1} + 7 \,.\, R_{i+1,2})), \\ r_N &= 0 \,, \end{aligned}$$

where

$$W_i = \frac{-1}{3 + 2 \,.\, R_{i+1,1}}, \quad \boldsymbol{W}_i = [W_i],$$

i.e.,

$$R_{i,2} = 0 \quad \text{for every } i \,.$$

The second method has been the method of *Algorithm* $\mathscr{T}3$ with the same $\boldsymbol{Z}_i$ as in the first method which means that the matrices $\boldsymbol{D}_i$ are of the form $[1, D_{i,2}]$ and matrices $\boldsymbol{T}_i$ are

$$T_i = -2 \,.\, R_{i,2} \,,$$

i.e., matrices $\boldsymbol{R}_i$ have been of the form $[0, R_{i,2}]$. "Uncorrect" initial value $\boldsymbol{s}_1$ equals one. The third method was the usual Gaussian elimination that is also a method of *Algorithm* $\mathscr{T}1$ as is shown in [6]. The constant h was chosen to be 10^{-4} and 10^{-8} and the order N of the matrix $\boldsymbol{G}$ has been 100 and 1000, respectively. Calculations have been done on SIEMENS 4004 in double precision. Astonishingly, the Gaussian elimination gave the worst results. The other methods gave correct values within the rank of machine accuracy. Both the method of *Algorithm* $\mathscr{T}1$ and of $\mathscr{T}3$ appeared to be scarce sensitive to the growth of the order of the matrix $\boldsymbol{G}$. This is in full agreement with the stability analysis we hope to present in the next paper.

Table 1.

i	y_i	
	Gaussian elimination	methods of *Algorithm* $\mathscr{T}1$ and $\mathscr{T}3$ and the exact solution
10	·00999898	·00999897
100	·08264462	·08264462
400	·23875115	·23875114
700	·21120292	·21120293
950	·04795425	·04795429
990	·00999895	·00999897

Acknowledgement. The author would like to express his greatest gratitude to Dr. E. Vitásek for his constant support during the preparation of the paper and to Dr. J. Taufer for his turning the author's attention to the problem and many fruitful discussions on the topics.

References

[1] *Babuška I., Práger M.* and *Vitásek E.:* Numerical processes in differential equations, Interscience, New York (1966).

[2] *Buchberger B.* and *Emeljanenko G. A.:* Methods of inversion of tridiagonal matrices. (Russian), Ž. Vyčisl. Mat. i Mat. Fiz., 13 (1973), 546—554.

[3] *Samarskij A. A.:* Introduction into the theory of difference methods (Russian), Moscow (1971).

[4] *Taufer J.:* Lösung der Randwertprobleme für Systeme von Linearen Differentialgleichungen, Rozpravy ČSAV, 83 (1973).

[5] *Ting C. T. T.:* A method of solving a system of linear equations whose coefficients form a tridiagonal matrix, Quart. of Appl. Maths., XXII (1962).
[6] *Malina L.:* Methods of the transfer of conditions and conditions of "well conditionedness". (Russian), in Numerical methods of linear algebra, ed. G. I. Marčuk, Novosibirsk (1977), 87—96.

Souhrn

OBECNÁ TEORIE PŘÍMÝCH METOD ŘEŠENÍ SOUSTAV ROVNIC S PÁSOVOU MATICÍ SOUSTAVY

ĽUBOR MALINA

V práci je ukázána možnost konstrukce obecného algoritmu pro řešení soustav lineárních rovnic s pásovou maticí soustavy. V první části se pojednává o Gaussově eliminaci způsobem podstatně odlišným od postupů dřívějších. Jsou zde osvětleny základní myšlenky, které vedou k definici obecné třídy přímých metod řešení soustav s pásovými maticemi (tyto se pak nazývají metodami přesunu okrajových podmínek), která je popsána v druhé části. V třetí části je ukázáno na příkladech, jak lze volbou parametru obecného algoritmu dostat některé známé metody.

Author's address: Dr. *Ľubor Malina*, CSc., Ústav aplikovanej matematiky a výpočtovej techniky PF UK, Matematický pavilón, Mlynská dolina, 816 31 Bratislava.