

ON THE SOLUTION OF SYSTEMS OF DIFFERENCE EQUATIONS

1961

SovietRxiv

View the original and related papers at <https://sovietrxiv.org/items/ru-196101.15981>

Source: Math-Net.Ru and CyberLeninka. Machine translation. Verify with the original.

Abstract

Full Text

MATHEMATICS

V. V. RUSANOV

ON THE SOLUTION OF SYSTEMS OF DIFFERENCE EQUATIONS

(Presented by Academician M. V. Keldysh, 7 VII 1960)

§ 1. In solving various problems for partial differential equations by the grid method, it is often necessary to solve systems of a large number of linear algebraic equations for the values of the unknown functions at the grid points. Usually it proves possible to write such systems in the following general form:

$$g_1 \mathbf{w}^{(1)} = \vec{\gamma}_1,$$

$$a_{i+1} \mathbf{w}^{(i+1)} + b_i \mathbf{w}^{(i)} = \vec{\pi}_{i+1/2}, \quad (1)$$

$$g_M \mathbf{w}^{(M)} = \vec{\gamma}_M.$$

Here $\mathbf{w}^{(i)}$ ($i = 1, 2, \dots, M$) are vectors of dimension L , whose components $w_k^{(i)}$ ($k = 1, 2, \dots, L$) are the sought values of the functions; a_{i+1} , b_i ($i = 1, 2, \dots, M-1$) are square matrices of size $L \times L$; g_1 is a matrix of size $L_1 \times L$; g_M is a matrix of size $L_M \times L$; $L_1 + L_M = L$; $\vec{\gamma}_1$ and $\vec{\gamma}_M$ are vectors of dimensions L_1 and L_M , respectively.

In passing from differential equations to difference equations, the first and third groups of equations in (1) arise from the boundary conditions, and the second—from the differential equations themselves.

Introduce the vector $\mathbf{W} = \{\mathbf{w}^{(1)}, \mathbf{w}^{(2)}, \dots, \mathbf{w}^{(M)}\}$, of dimension ML , composed of all the unknowns. \mathbf{W} is the solution of the system

$$A\mathbf{W} = \mathbf{P}, \quad (2)$$

where A is a matrix of size $ML \times ML$:

$$A = \begin{Bmatrix} g_1 & 0 & 0 & \dots & 0 & 0 \\ b_1 & a_2 & 0 & \dots & 0 & 0 \\ 0 & b_2 & a_3 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & 0 & \dots & b_{M-1} & a_M \\ 0 & 0 & 0 & \dots & 0 & g_M \end{Bmatrix}, \quad \mathbf{P} = \begin{Bmatrix} \vec{\gamma}_1 \\ \vec{\pi}_{1/2} \\ \vec{\pi}_{3/2} \\ \vdots \\ \vec{\pi}_{M-1/2} \\ \vec{\gamma}_M \end{Bmatrix}. \quad (3)$$

The number ML is usually of the order of several hundreds, and the direct solution of system (2) without using the special form of the matrix A is very laborious and uneconomical. One of the methods that use the specificity of the matrix A is the method of matrix elimination^(1,2), but its application is limited by the need to satisfy a number of stability conditions, the verification of which is difficult. Moreover, these conditions may not even be necessary for the possibility of solving system (2) by some other method. The essence of the matter is that in the elimination method the order of operations is strictly determined, and it may happen that one of those matrices that must be inverted in the course of the solution, turns out to be singular. The situation is quite analogous to that which may occur when solving a system by the elimination method without pivoting, when a previously specified minor of the matrix by which one must divide accidentally becomes zero. It is therefore natural to try to combine the sweep method with the elimination method with pivoting. One such combined method is described below.

§ 2. As in the ordinary elimination method, the solution of the system is carried out in two stages. Stage I, or the forward sweep, consists in eliminating unknowns and reducing the matrix A to triangular form (up to a permutation of rows and columns). Stage II, or the back substitution, consists in finding the solution by back substitution.

The choice of pivots and the elimination of unknowns are performed in the following order:

- 1) From the matrix g_1 , L_1 pivots are selected successively. Each unknown among the L_1 selected ones remains in only one of the rows g_1 with coefficient one, and is eliminated from the remaining rows. In addition, all L_1 selected unknowns are eliminated from the subsequent L equations, so that in the matrix b_1 the pivot columns become zero. The remaining nonzero elements of g_1 , with the numbers of the selected pivot rows and columns indicated, are stored for the backward sweep.
- 2) From the remaining L_M nonzero columns of the matrix b_1 , L_M pivots are selected successively, and the corresponding unknowns are eliminated from the remaining equations. At the same time the rows of the matrix a_2 are combined in accordance with the combination of the rows of b_1 . After this step is completed, only ones and zeros remain in the matrix b_1 . The number of ones is L_M , and they stand one in each of the L_M pivot rows.

- 3) From L_1 rows of the matrix a_2 which were not pivots in the preceding step, L_1 pivots are selected successively, and the selected unknowns are eliminated both from all rows of a_2 and from all rows of b_2 (except for the one row of a_2 in which the unknown remains with coefficient one). After the step is completed, the remaining nonzero elements of the matrix a_2 are stored for the backward sweep together with the numbers of the selected pivots.

Then steps 2) and 3) are repeated alternately as applied to the matrices $b_2, a_3, b_3,$ etc., up to and including the matrix a_M .

- 4) From the remaining L_M nonzero columns and rows of g_M , L_M pivots are selected, i.e., a square matrix of size $L_M \times L_M$ is inverted. Thus L_M unknowns are found, more precisely L_M components of the vector $\mathbf{w}^{(M)}$, from the total number L of components.

This completes the forward sweep. In the backward sweep the computation proceeds in the same way as in the elimination and sweep methods, namely, with the aid of the previously stored parts of the transformed matrices a_i , the vectors $\mathbf{w}^{(M)}, \mathbf{w}^{(M-1)}, \dots, \mathbf{w}^{(1)}$ are computed successively.

§ 3. The main difference between the proposed method and the sweep method is as follows. In the sweep method, the successively computed relations between the components of $\mathbf{w}^{(i)}$ are written in the form

$$\mathbf{u}^{(i)} = X_{12}^{(i)} \mathbf{v}^{(i)} + Y_1^{(i)}, \quad (4)$$

where $\mathbf{u}^{(i)}$ and $\mathbf{v}^{(i)}$ are vectors consisting of L_1 and L_M components of the vector $\mathbf{w}^{(i)}$, respectively, and $X_{12}^{(i)}$ is a matrix of size $L_1 \times L_M$. In the method described above, relations of the form (4) are essentially also computed, but the partition of $\mathbf{w}^{(i)}$ into $\mathbf{u}^{(i)}$ and $\mathbf{v}^{(i)}$ is not prescribed in the same way for all $\mathbf{w}^{(i)}$, but is chosen automatically for each $\mathbf{w}^{(i)}$ during the course of the solution so that the matrix $X_{12}^{(i)}$ has as few elements as possible.

It is not difficult to show, analogously to how this is done for the method of pivots, that in the method described, under the condition that the matrix A is nonsingular, the possibility is excluded of choosing as a pivot a number,

equal to zero. As applied to the numerical solution of a system of equations, this means that the proposed method will give a good result for a well-conditioned matrix A .

It is also easy to verify that, in terms of the number of operations performed, the algorithm described belongs to the class of optimal ones. Namely, if a_i and b_i are of general form, then solving system (1) by any other direct method (including the sweep method) will require no smaller number of operations.

In conclusion, we note that the proposed method has been tested repeatedly in solving various problems on the "Strela" electronic computer. The standard

program for solving systems of the form (1) contains 220 commands and uses $2L^2 + 4L$ working cells. In the forward pass, the matrices a_i and b_i are entered into the main memory successively from tape or drum, where the information for the backward pass is then recorded. The total number of machine cycles for $L > 4$ and any L_1 does not exceed $M(6L^3 + 28L^2 + 76L)$, which is approximately twice the number of arithmetic operations.

Received
2 VI 1960

References

1. G. I. Marchuk, *Numerical methods for calculating nuclear reactors*, Moscow, 1958.
2. V. V. Rusanov, *Computational Mathematics*, collection 6, Moscow, 1950.

Note: Figure translations are in progress. See original paper for figures.

Source: Math-Net.Ru and CyberLeninka. Machine translation. Verify with the original.