Discrete & Continuous Models
& Applied Computational Science
ISSN 2658-7149 (online), 2658-4670 (print)

2022, 30 (1) 62–78
http://journals.rudn.ru/miph

Review article

UDC 535:535.3:681.7
DOI: 10.22363/2658-4670-2022-30-1-62-78

Finite-difference methods for solving 1D Poisson problem

Serge Ndayisenga[1],
Leonid A. Sevastianov[1, 2], **Konstantin P. Lovetskiy**[1]

[1] *Peoples' Friendship University of Russia (RUDN University)*
6, Miklukho-Maklaya St., Moscow, 117198, Russian Federation
[2] *Bogoliubov Laboratory of Theoretical Physics*
Joint Institute for Nuclear Research
6, Joliot-Curie St., Dubna, Moscow Region, 141980, Russian Federation

(received: January 21, 2022; revised: February 2, 2022; accepted: February 18, 2022)

Abstract. The paper discusses the formulation and analysis of methods for solving the one-dimensional Poisson equation based on finite-difference approximations — an important and very useful tool for the numerical study of differential equations. In fact, this is a classical approximation method based on the expansion of the solution in a Taylor series, based on which the recent progress of theoretical and practical studies allowed increasing the accuracy, stability, and convergence of methods for solving differential equations. Some of the features of this analysis include interesting extensions to classical numerical analysis of initial and boundary value problems. In the first part, a numerical method for solving the one-dimensional Poisson equation is presented, which reduces to solving a system of linear algebraic equations (SLAE) with a banded symmetric positive definite matrix. The well-known tridiagonal matrix algorithm, also known as the Thomas algorithm, is used to solve the SLAEs. The second part presents a solution method based on an analytical representation of the exact inverse matrix of a discretized version of the Poisson equation. Expressions for inverse matrices essentially depend on the types of boundary conditions in the original setting. Variants of inverse matrices for the Poisson equation with different boundary conditions at the ends of the interval under study are presented — the Dirichlet conditions at both ends of the interval, the Dirichlet conditions at one of the ends and Neumann conditions at the other. In all three cases, the coefficients of the inverse matrices are easily found and the algorithm for solving the problem is practically reduced to multiplying the matrix by the vector of the right-hand side.

Key words and phrases: 1D Poisson equation, finite difference method, tridiagonal matrix inversion, Thomas algorithm, Gaussian elimination

1. Introduction

Applied mathematical models are mainly based on the use of partial differential equations [1]. The solution must satisfy a given equation of mathematical physics and some additional relations, which are, first, boundary and initial conditions. The most important for applications [2] are second-order equations — elliptic, parabolic, and hyperbolic. Currently for equations of mathematical physics, methods of numerical solution and the appropriate software [3], [4], as well as computer algebra systems (CASs) such as Sage, Mathematica, Maxima and Maple are actively developed to implement these methods. Many features of stationary problems of mathematical physics described by elliptic equations of the second order can be illustrated by considering the simplest boundary value problems for an ordinary differential equation of the second order. Perhaps the simplest second-order elliptic equation is the Poisson equation.

Let us consider some methods for the numerical solution of this equation and compare the investigated methods.

The Poisson equation [1] is a special case of the heat conduction equation describing the dependence of the temperature of a medium on spatial coordinates and time, and the heat capacity and thermal conductivity of the medium (in the general case, inhomogeneous) are considered to be given. We will consider the problem of finding the steady-state distribution of density or temperature (e.g., when the distribution of sources does not depend on time). In this case, terms with time derivatives are eliminated from the non-stationary equation and a stationary heat equation is obtained, which belongs to the class of elliptic equations. A two-point boundary value problem is the problem of finding a solution to an ordinary differential equation or second-order systems in the interval $a \leqslant x \leqslant b$. Additional conditions are imposed on the solution at any two points of the interval, e.g., a and b — the 'boundaries' of the segment (hence the name of the problem).

Consider a second-order differential equation

$$-\frac{d}{dx}\left(k\left(x\right)\frac{du}{dx}\right)+p\left(x\right)u\left(x\right)=f\left(x\right), \quad a \leqslant x \leqslant b. \tag{1}$$

It is called the one-dimensional stationary heat conduction equation and arises in the mathematical modeling of many important processes. For example, this equation describes the steady-state temperature distribution $u\left(x\right)$ in a heat-conducting rod of length $l = b - a$. In this case, $k\left(x\right)$ is the thermal conductivity coefficient; $w\left(x\right) = -k\left(x\right)\dfrac{du}{dx}$ is the heat flux density, $p\left(x\right)$ is the heat transfer coefficient (pu is the heat sink power proportional to the temperature u); $f\left(x\right)$ is the density of heat sources (at $f \leqslant 0$ it is the density of heat sinks).

The boundary value problem is much harder to solve than the Cauchy problem, and various approaches are used for this purpose. The most common are various sampling methods that allow replacing the original problem with a certain discrete analog. The resulting discrete boundary value problem is a system of equations (possibly nonlinear) with a finite number of unknowns and can be numerically solved using special direct or iterative methods. One of the simplest discretization algorithms often used in applied scientific and technical calculations is the method of finite differences [5].

The most commonly used method for solving difference equations arising in the approximation of boundary value problems for equations of mathematical physics is the sweep method [6], [7], or the Thomas method [8].

Below we will show how the difference method is applied to solve the boundary value problem (1), restricting ourselves, for simplicity, to an equation with a constant coefficient $k(x) \equiv 1$. In this case, the boundary value problem with Dirichlet boundary conditions takes the form

$$u''(x) - p(x)\,u(x) = f(x)\,, \quad a < x < b, \tag{2}$$

$$u(a) = \alpha, \quad u(b) = \beta. \tag{3}$$

Introduce on $[a, b]$ a grid $a = x_0 < x_1 < x_2 < ... < x_n = b$, which for simplicity is assumed uniform. Let us approximately express the second derivative of the solution in terms of the values of the future solution at the grid nodes $u_n = u(x_n)$. We use the simplest symmetric difference approximation

$$u''(x_n) \approx \frac{1}{h^2}\left(u_{n-1} - 2u_n + u_{n+1}\right), \quad h = x_{n+1} - x_n = \text{const.} \tag{4}$$

Using such an approximation at each internal grid node x_n, $1 \leqslant n \leqslant N-1$ and substituting it into the differential equation (2), we transform the differential equation (1) into a system of finite-difference equations, i.e., into a system of approximate linear algebraic equations, the solution of which will be an approximate solution $y_n \approx u(x_n)$. Finite-difference equations cannot be written at the boundary nodes $n = 0$, $n = N$, otherwise the indices of the nodes will go beyond the permissible limits [5]. Denoting $p_n = p(x_n)$ and $f_n = f(x_n)$, we get a system of $(N-1)$ linear equations with respect to the approximate values of the solution at grid nodes

$$y_{n-1} - \left(2 + h^2 p_n\right) y_n + y_{n+1} = h^2 f_n, \quad 1 \leqslant n \leqslant N-1. \tag{5}$$

The number of unknowns y_n, $0 \leqslant n \leqslant N$ equals $(N+1)$, i.e., it is greater than the number of equations (5). The lacking two equations are to be obtained from the boundary conditions (3)

$$y_0 = \alpha, \quad y_N = \beta. \tag{6}$$

Solving the algebraic system (5), (6) we get an approximate solution of the boundary value problem (2), (3).

Further analysis of the described algorithm of solving the boundary value problem is to answer three important questions.

— What are the conditions for the existence of a solution to the system of algebraic equations?
— Does the solution of the system of algebraic equations tend to the exact solution of the boundary value problem upon reducing the grid step?
— Is it possible to develop an algorithm (procedure) for finding the solution with given accuracy by reducing the grid step?

It is known [5, P. 66], that for a rather wide class of the boundary value problem coefficients it is possible to prove the existence of a finite-difference solution and its convergence to the exact solution. The following theorem takes place.

Theorem 1. *Let $p(x)$, $f(x)$ are twice continuously differentiable on $[a, b]$, $p(x) \geqslant m$, where the constant $m \geqslant 0$. Also let the step h be small enough, so that $h \leqslant 2$. Then the finite-difference solution exists, its difference from the exact solution by the norm c being of the order of $O(h^2)$.*

Remark 1. The matrix of the system (5), (6) is tridiagonal. It is not difficult to solve the system by the Gaussian method for a strip matrix or by sweep method. These are direct methods. They allow finding a solution, executing about nine arithmetic operations for each node. By virtue of the conditions of the theorem, the solution of the system of equations by the sweep method exists, is unique and found without accumulating round-off errors.

Remark 2. The conditions of the theorem are sufficient, but not necessary. Even if the conditions are not met, in most cases the finite-difference solution exists and converges to the exact one. Under additional assumptions, it is possible to construct an asymptotically accurate estimate of the error. Then it is possible to apply the grid refinement and Richardson's method to find the posterior estimate of the error and calculations with control of the accuracy.

2. Finite-difference scheme

The problem in matrix form can be represented as

$$\begin{pmatrix} -2+p_1 & 1 & 0 & \cdots & 0 \\ 1 & -2+p_2 & 1 & \cdots & 0 \\ 0 & 1 & -2+p_3 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & 0 \\ 0 & 0 & 0 & \ddots & 1 \\ 0 & 0 & 0 & 1 & -2+p_N \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ \vdots \\ \vdots \\ u_N \end{pmatrix} = \begin{pmatrix} h^2 f_1 - u_a \\ h^2 f_2 \\ h^2 f_3 \\ h^2 f_4 \\ \vdots \\ \vdots \\ h^2 f_N - u_b \end{pmatrix}. \quad (7)$$

When applying the sweep method to systems of the form (7), during a forward sweep, both the coefficients of the matrix and the elements of the vector on the right-hand side are recalculated. The matrix is thus reduced to two-diagonal form. During the backward sweep, the components of the solution are calculated at the second stage. Tridiagonal matrices, which are inverted using the simple sweep method, often arise when solving differential equations of two independent variables by the finite-difference method, e.g., when solving a linear one-dimensional heat equation.

For such systems, the solution can be obtained in operations instead of required by the Gaussian elimination method. The first sweep of the method calculates the sweep coefficients, based on which the inverse substitution yields the solution. Examples of such matrices usually arise from discretization of the one-dimensional Poisson equation and interpolation by the natural cubic spline.

For the simplest one-dimensional Poisson equation in the case when $p(x) \equiv 0$, the authors of Refs. [9], [10] proposed a solution based on the analytical (exact) representation of the inverse matrix coefficients.

3. The exact formulation of the inverse of the tridiagonal matrix for solving the 1D Poisson equation with the finite difference method

Consider a method for solving the one-dimensional Poisson equation using the finite difference method based on exact formulas for the inverse of the Laplacian tridiagonal matrix. In the method proposed in Ref. [11], formulas for the coefficients of the inverse matrix are directly derived. Thus, the procedure of solving the one-dimensional Poisson equation becomes very accurate and very fast. This method is a very important tool for solving many physical and technical problems, where the Poisson equation often appears when describing (modeling) various physical phenomena.

3.1. The finite difference method for solving the Poisson equation with Dirichlet–Dirichlet boundary conditions

Consider a function $u(x)$, that satisfies the Poisson equation $u''(x) = f(x)$ on the interval $]a, b[$, where $f(x)$ is a given function. We require that the function $u(x)$ satisfy the Dirichlet–Dirichlet boundary conditions: $u(a) = \alpha$, $u(b) = \beta$. On the considered interval $[a, b]$ we specify a one-dimensional grid $x_i = a + i \cdot \Delta x$, $i = 0, \dots, N+1$, where the uniform step of the grid is calculated as $\Delta x = \dfrac{b-a}{N+1} = h$. We denote by $u_i = u(x_i)$ and $f_i = f(x_i)$, $i = 0, \dots, N+1$ the values of the approximate solution and the function in the right-hand side.

Replacing the second derivative by symmetric difference expressions, we obtain the following system for internal nodes:

$$u_{i-1} - 2u_i + u_{i+1} = h^2 f_i, \quad i = 1, \dots, N. \tag{8}$$

In matrix form, the system of linear algebraic equations (8), taking into account the boundary conditions, can be written in the form $Au = F$, where $F = \left(h^2 f_1 - u_a, h^2 f_2, \dots, h^2 f_{N-1}, h^2 f_N - u_b\right)^T$, or

$$\begin{pmatrix} -2 & 1 & 0 & 0 & 0 & \cdots & \cdots & 0 \\ 1 & -2 & 1 & 0 & 0 & \cdots & \cdots & 0 \\ 0 & 1 & -2 & 1 & 0 & \cdots & \cdots & 0 \\ 0 & 0 & 1 & -2 & 1 & \ddots & \cdots & 0 \\ 0 & 0 & 0 & 1 & -2 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & 0 & 0 & \ddots & \ddots & \ddots & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -2 \end{pmatrix} \times \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ \vdots \\ u_{N-1} \\ u_N \end{pmatrix} = \begin{pmatrix} h^2 f_1 - u_a \\ h^2 f_2 \\ h^2 f_3 \\ h^2 f_4 \\ h^2 f_5 \\ \vdots \\ h^2 f_{N-1} \\ h^2 f_N - u_b \end{pmatrix}. \tag{9}$$

Thus, the solution of the one-dimensional Poisson equation is reduced to the inversion of the tridiagonal symmetric negative definite matrix

$$A = \left(a_{ij}\right), \quad i, j = 1, \ldots, N.$$

The inverse matrix which we denote by

$$B = \left(b_{ij}\right), \quad i, j = 1, \ldots, N,$$

is also symmetric.

The elements of matrix A may be briefly written as

$$a_{ij} = \begin{cases} -2, & i = j, \\ 1, & |i-j| = 1, \quad i = 1, \ldots, N \\ 0, & |i-j| > 1, \end{cases} \tag{10}$$

and the elements of matrix B are related by the following formulas:

$$\begin{cases} -2b_{i1} + b_{i2} = \delta_i^1, \\ b_{ij-1} - 2b_{ij} + b_{ij+1} = \delta_i^j, \quad 1 < i, \quad j < N, \\ b_{iN-1} - 2b_{iN} = \delta_i^N, \end{cases} \tag{11}$$

where δ_i^j is the Kronecker symbol.

3.2. Calculating the inverse matrix

Relations (11) allow deriving the following interesting dependencies

$$b_{ij+1} = b_{ij} + b_{i1}, \quad b_{ij} = jb_{i1} + (j-1). \tag{12}$$

From relations (12) it follows that the elements of inverse matrix B are unambiguously determined by the value of the element b_{11}. This coefficient can be determined based on the behavior of matrix B at different dimensionalities N:

$$b_{11} = -N/(N+1). \tag{13}$$

From relations (12) and (13), it is easy to express the elements of the first row and the first column of the inverse matrix

$$\begin{cases} b_{1j} = -(N-(j-1))/(N+1), \\ b_{i1} = -(N-(i-1))/(N+1). \end{cases} \tag{14}$$

These relations allow completing the accurate and full determination of the coefficients of the inverse matrix $B = \left(b_{ij}\right)$, $i, j = 1, \ldots, N$:

$$b_{ij} = \begin{cases} -j(N-(i-1))/(N+1), & i \geqslant j, \\ -i(N-(j-1))/(N+1), & i < j; \end{cases} \tag{15}$$

$$B = -\frac{1}{(N+1)}\times$$
$$\times\begin{pmatrix} N & N-1 & \cdots & N-(j-1) & \cdots & 2 & 1 \\ N-1 & 2\,(N-1) & \cdots & 2\,[N-(j-1)] & \cdots & 4 & 2 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ N-(i-1) & 2\,[N-(i-1)] & \cdots & i\,[N-(j-1)] & \cdots & 2i & i \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 2 & 4 & \cdots & 2j & \cdots & 2\,(N-1) & N-1 \\ 1 & 2 & \cdots & j & \cdots & N-1 & N \end{pmatrix}.$$

With the inverse matrix elements known, it is easy to get the solution of the one-dimensional Poisson equation by mere multiplication of the matrix by the right-hand side vector $u = BF$.

3.3. Classification of media

Taking into account the specific form of the inverse matrix and its persymmetry makes it easy to express the solution u_N at the point x_n

$$u_N = -\,(N+1)^{-1}\sum_{i=1}^{N} i\cdot F_i. \tag{16}$$

The direct search for the solution u_{N-1} at the point x_{N-1} leads to the expression

$$u_{N-1} = -\,(N+1)^{-1}\left[\left[\sum_{i=1}^{N-1} 2i\cdot F_i\right] + (N-1)\,F_N\right]. \tag{17}$$

In a similar way, it is possible to derive the expressions for calculating the rest components of the solution in the form

$$u_{N-k} = -\,(N+1)^{-1}\times$$
$$\times\left[(k+1)\left[\sum_{i=1}^{N-k} iF_i\right] + (N-k)\left[\sum_{i=N-k+1}^{N}(N-(i-1))\,F_i\right]\right],$$
$$k = 0,1,\ldots,N-1 \tag{18}$$

or in the form

$$u_k = -\,(N+1)^{-1}\times$$
$$\times\left[(N-k+1)\left[\sum_{i=1}^{k} iF_i\right] + k\left[\sum_{i=k+1}^{N}(N-(i-1))\,F_i\right]\right],$$
$$k = 1,\ldots,N. \tag{19}$$

From the computational point of view, it is preferable to use Eqs. (15), when programming the procedure of calculating the solution.

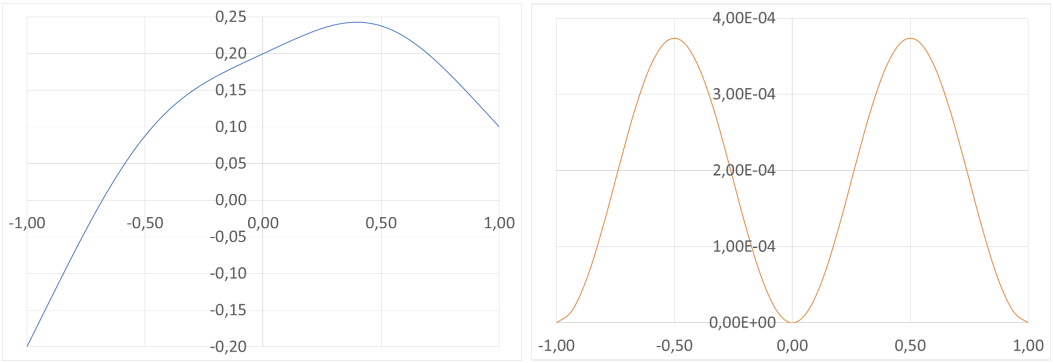
Let us consider the numerical solution of the problem of finding a scalar potential given on the interval $[-1, 1]$ and satisfying the Poisson equation $\Delta\Phi\left(x\right) = \dfrac{\partial^2\Phi\left(x\right)}{\partial x^2} = f\left(x\right) = -\cos^2\left(\pi\left(x-0.5\right)\right)$ and the Dirichlet–Dirichlet boundary conditions: $\Phi\left(-1\right) = -0.2$, $\Phi\left(1\right) = 0.1$.

The exact solution is expressed by the formula

$$\Phi_{\text{exact}} = -\frac{x^2}{4} + \left[\frac{\cos\left(\pi\left(x-0.5\right)\right)}{2\pi}\right]^2 + \frac{x}{4} - 0.1\left(x+1\right) + 0.3. \tag{20}$$

The software implementation of the algorithm consists of several lines, namely, filling the vector on the right-hand side of Eq. (9) and multiplying the inverse matrix B by this vector using Eqs. (14).

Figure 1 illustrates the results of the numerical experiment.



(a) Exact solution

(b) Calculation error

Figure 1. The maximal error at points $x = \pm 0.5$ is 0.36 at $N = 30$ and decreases to 0.036 at $N = 300$

4. Solving the 1D Poisson equation with the Neumann–Dirichlet and Dirichlet–Neumann boundary conditions

The problem is to determine the scalar potential $u\left(x\right)$ satisfying the one-dimensional Poisson equation $\Delta u\left(x\right) = f\left(x\right)$ on the interval $]a, b[$, where $f\left(x\right)$ is a given function. It is necessary to find the solution satisfying the Neumann–Dirichlet boundary conditions $u'\left(a\right) = u'_a$ and $u\left(b\right) = u_b$. Let us consider a special uniform grid for the finite difference method with the step $\Delta x = \dfrac{b-a}{N} = h$, consisting of $N+1$ points. The coordinates of the grid nodes $\left(x_i\right)$ are determined by the expression $x_i = a + \left(i-1\right)\cdot h$, $i = 0, 1, \dots N+1$. We denote by u_i the approximate values of the desired solution at point

x_i : $u_i = u(x_i)$, and by f_i the value of the given function in the right-hand side at the same point. In addition, let us denote by $u'_i = u'(x_i)$ and $u''_i = u''(x_i)$ the values of the first and second derivatives of the sought solution at the grid node at the same point. Replacing the derivatives with symmetric finite-difference expressions [12], we arrive at the approximation formulas of the second order of accuracy for the first derivatives

$$u'_i = \frac{u_{i+1} - u_{i-1}}{2h} + O\left(h^2\right), \quad i = 1, 2, 3, ..., N \tag{21}$$

and for the second derivatives

$$u''_i = \frac{u_{i-1} - 2u_i + u_{i+1}}{h^2} + O\left(h^2\right), \quad i = 1, 2, ..., N. \tag{22}$$

The system of linear equations for the internal nodes of the interval looks as

$$u_{i-1} - 2u_i + u_{i+1} = h^2 f_i, \quad i = 1, ..., N. \tag{23}$$

4.1. The Neumann–Dirichlet boundary conditions

Let us derive equations complementing the system with the boundary conditions at the left and right ends of the interval taken into account. Assuming the use of Eqs. (21) and (23) possible and combining them at $i = 0$, we eliminate u_{-1} from the system of equations.

$$-u_1 + u_2 = h^2 \frac{f_i}{2} + hu'_a, \quad i = 1, ..., N. \tag{24}$$

Thus, introducing into consideration an additional virtual point $x_0 = a - h$ allows using the central differences with the order of approximation $O\left(h^2\right)$ for the sought solution even at the boundary point of the interval.

We introduce the vector F with the components expressed as

$$F_1 = h^2 \frac{f_i}{2} + hu'_a, \quad F_N = h^2 f_N - u_b, \quad F_i = h^2 f_i, \quad i = 2, 3, ..., N-1. \tag{25}$$

As a result, the system of equations that determines the solution components reduces to the form

$$\begin{pmatrix} -1 & 1 & 0 & 0 & 0 & \cdots & \cdots & 0 \\ 1 & -2 & 1 & 0 & 0 & \cdots & \cdots & 0 \\ 0 & 1 & -2 & 1 & 0 & \cdots & \cdots & 0 \\ 0 & 0 & 1 & -2 & 1 & \ddots & \cdots & 0 \\ 0 & 0 & 0 & 1 & -2 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & 0 & 0 & \ddots & \ddots & \ddots & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -2 \end{pmatrix} \times \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ \vdots \\ u_{N-1} \\ u_N \end{pmatrix} = \begin{pmatrix} \frac{h^2}{2} f_1 + hu'_a \\ h^2 f_2 \\ h^2 f_3 \\ h^2 f_4 \\ h^2 f_5 \\ \vdots \\ h^2 f_{N-1} \\ h^2 f_N - u_b \end{pmatrix}, \tag{26}$$

where the matrix $A = \{a_{ij}\}$, $i, j = 1, \dots, N$ of system (8) is symmetric tridiagonal negative definite and possesses the property of diagonal transformation. The presence of diagonal dominance in the coefficient matrix guarantees the stability of the sweep method; however, in this case, there is a way to calculate the elements of the inverse matrix.

4.2. Calculation of the inverse matrix elements

Let us write down the properties of the inverse matrix $B = \{b_{ij}\}$, $i, j = 1, \dots, N$, $B = A^{-1}$, following directly from its definition. It must be symmetrical and its elements must satisfy the following relations:

$$\begin{cases} -b_{1j} + b_{2j} = \delta_j^1, \quad 1 < j < N, \\ b_{i1} - 2b_{i2} + b_{i3} = \delta_i^2, \quad 1 < i < N, \\ b_{i-1j} - 2b_{ij} + b_{i+1j} = \delta_i^j, \quad 1 < i, j < N, \\ b_{iN-1} - 2b_{iN} = \delta_i^N, \quad 1 < i < N, \end{cases} \tag{27}$$

where δ_i^j is the Kronecker symbol.

The elements of the inverse matrix also satisfy the relations

$$b_{ij} = \begin{cases} b_{11} + (j-1), & i \leqslant j, \\ b_{11} + (i-1), & i > j. \end{cases} \tag{28}$$

The analysis of behavior of the system determinant allows deriving the expressions

$$\det(B) = (-1)^N,$$
$$b_{11} = \frac{N \cdot (-1)^{N-1}}{(-1)^N} = -N, \text{ and } b_{NN} = \frac{(-1)^{N-1}}{(-1)^N} = -1 = b_{1N}. \tag{29}$$

Using Eqs. (27)–(29), we can exactly determine the elements of the inverse matrix, which is related to the search for the approximate solution in the case of the Neumann–Dirichlet boundary conditions. Thus, the elements of matrix B are determined by the expressions

$$b_{ij} = \begin{cases} -[N - (j-1)], & i \leqslant j, \\ -[N - (i-1)], & i > j. \end{cases} \tag{30}$$

The elements of the inverse matrix can be alternatively expressed as

$$b_{ij} = -[N - [\max(i,j) - 1]] = -\left[N - \left[\frac{(i+j) + |i-j|}{2} - 1\right]\right]. \tag{31}$$

The expressions (30) and (31) are equivalent. However, for software implementation, the first one is preferable.

As a result of the transformations carried out, explicit expressions for the elements of the inverse matrix are obtained, and the solution of the Poisson problem with Neumann–Dirichlet boundary conditions can be obtained using a simple multiplication of the inverse matrix by the vector of the right-hand side: $U = BF$, where

$$B = -\begin{pmatrix} N & N-1 & N-2 & \cdots & \cdots & 2 & 1 \\ N-1 & N-1 & N-2 & \cdots & \cdots & 2 & 1 \\ N-2 & N-2 & N-2 & \cdots & \cdots & 2 & 1 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 2 & 2 & 2 & \cdots & \cdots & 2 & 1 \\ 1 & 1 & 1 & \cdots & \cdots & 1 & 1 \end{pmatrix}.$$

Each solution component can be expressed directly using the formula

$$u_k = -\left[(N-k+1)\left[\sum_{i=1}^{k} F_i\right] + \left[\sum_{i=k+1}^{N} (N-(i-1)) \cdot F_i\right]\right], \qquad k = 1, 2, \ldots, N. \quad (32)$$

Formula (32) gives a simple analytical expression for the solution of the Poisson equation with Neumann–Dirichlet boundary conditions. It is very easy to program it either directly or based on Eq. (30). One double loop will be enough to compute the entire solution.

4.3. Example

Consider a numerical solution of the problem of finding a scalar potential defined on the interval $[-a, b]$ and satisfying the Poisson equation

$$\Delta\Phi(x) = \frac{\partial^2 \Phi(x)}{\partial x^2} = f(x) = V_0 \cos(kx + \varphi_0),$$

where a, b, V_0, k and φ_0 are given constants, and the Neumann–Dirichlet boundary conditions $\dfrac{d\Phi}{dx}(a) = \Phi'_a$ and $\Phi(b) = \Phi_b$.

The known exact solution is expressed as

$$\Phi_{\text{exact}}(x) = \left[\Phi'_a - \frac{V_0}{k}\sin(ka + \varphi_0)\right](x-b) - \\ - \frac{V_0}{k^2}\left[\cos(kx+\varphi_0) - \cos(kb+\varphi_0)\right] + \Phi_b. \quad (33)$$

Let us consider the finite-difference solution at $a = -\dfrac{\pi}{2}$, $b = \dfrac{\pi}{4}$, $V_0 = 1$, $k = \dfrac{\pi}{2}$ and $\varphi_0 = \dfrac{\pi}{4}$.

We define the computational grid with the following parameters:

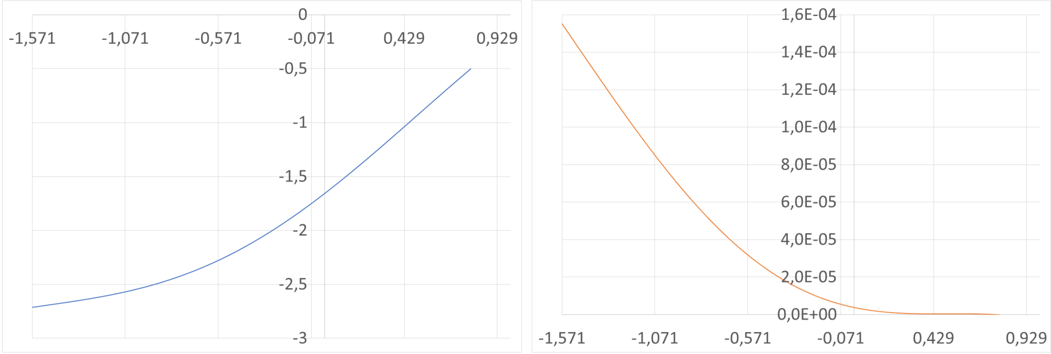
$$N = 100, \quad \Delta x = h = \frac{b-a}{N}, \quad x_i = (i-1)\,\Delta x, \quad \Phi_i = \Phi\,(x_i)$$

and $f_i = f\,(x_i) = \cos\,(kx_i + \varphi_0)$. The solution is assumed to satisfy the Neumann–Dirichlet conditions specified as: $\Phi'_a = 1/4$ and $\Phi_b = -1/2$.

We calculate the solution of the Poisson problem multiplying the inverse matrix with the elements determined by expressions (30) by the right-hand side vector, corrected using Eqs. (26).

The software implementation of the algorithm consists of a few lines: filling the right-hand side vector (26) and multiplying the inverse matrix B by this vector using Eqs. (30).

Figure 2 illustrates the results of the numerical experiment.



(a) Exact solution

(b) Calculation error

Figure 2. The maximal error at $x = -1.571$ is $1.55E-04$ for $N = 100$ and decreases to $1.55E-06$ for $N = 1000$

5. Dirichlet–Neumann boundary conditions

5.1. Discretization and matrix equation

By analogy with the case of the Neumann–Dirichlet boundary conditions, we consider the symmetric case with the Dirichlet–Neumann boundary conditions. Let us first define a suitable sampling grid on the interval $[a, b]$. Grid points $\{x_i,\ i = 0, 1, \ldots, N+1\}$ are specified as $x_i = a + ih$. The boundary conditions u_a and u'_b, complementing the Poisson equation redefine the system of finite-difference equations (23). The solution value u_{N+1} at the 'virtual' point x_{N+1} is expressed using the boundary condition for the derivative, approximating the latter by symmetric central differences. As the last equation of the system, we get

$$u_{N-1} - u_N = h^2 \frac{f_N}{2} - hu'_b. \tag{34}$$

The transformed right-hand side vector F is presented as

$$F_N = h^2\frac{f_N}{2} - hu_b', \quad F_1 = h^2 f_1 - u_a, \quad F_i = h^2 f_i, \quad i = 2, \dots, N-1. \tag{35}$$

Like in the previous case of Neumann–Dirichlet boundary conditions, the resulting matrix of the equation

$$\begin{pmatrix} -2 & 1 & 0 & 0 & 0 & \cdots & \cdots & 0 \\ 1 & -2 & 1 & 0 & 0 & \cdots & \cdots & 0 \\ 0 & 1 & -2 & 1 & 0 & \cdots & \cdots & 0 \\ 0 & 0 & 1 & -2 & 1 & \ddots & \cdots & 0 \\ 0 & 0 & 0 & 1 & -2 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & 0 & 0 & \ddots & \ddots & \ddots & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{pmatrix} \times \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ \vdots \\ u_{N-1} \\ u_N \end{pmatrix} = \begin{pmatrix} h^2 f_1 - u_a \\ h^2 f_2 \\ h^2 f_3 \\ h^2 f_4 \\ h^2 f_5 \\ \vdots \\ h^2 f_{N-1} \\ h^2 \frac{f_N}{2} - hu_b' \end{pmatrix} \tag{36}$$

is symmetric three-diagonal negative definite, with the dominant main diagonal.

With respect to the antidiagonal, this matrix is symmetric to the matrix used in the solution of the Poisson problem with the Neumann–Dirichlet boundary conditions. The system is definite and has a unique solution for any right-hand side.

Using the antidiagonal symmetry with respect to the Neumann–Dirichlet problem, we construct the inverse matrix for the Dirichlet–Neumann case:

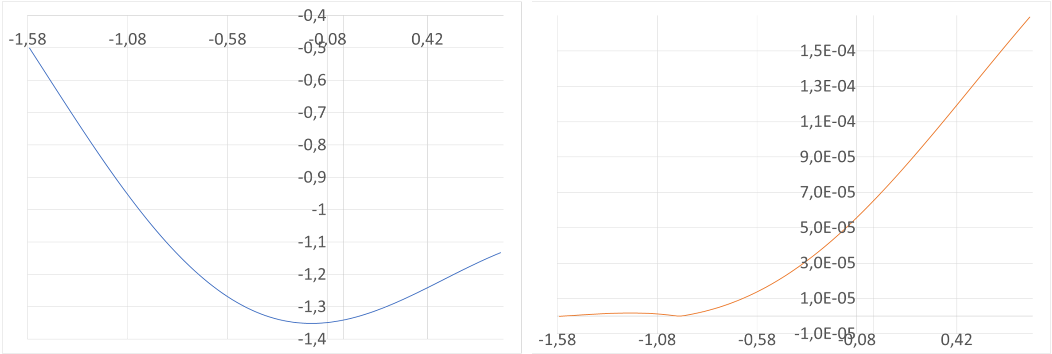
$$B = -\begin{pmatrix} 1 & 1 & 1 & 1 & \cdots & 1 & 1 \\ 1 & 2 & 2 & 2 & \cdots & 2 & 2 \\ 1 & 2 & \ddots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \ddots & \cdots & \cdots & \cdots \\ 1 & 2 & \cdots & \cdots & N-2 & N-2 & N-2 \\ 1 & 2 & \cdots & \cdots & N-2 & N-1 & N-1 \\ 1 & 2 & \cdots & \cdots & N-2 & N-1 & N \end{pmatrix}.$$

Therefore, the exact solution of the system of equations (36) can be written very simply (in a single line)

$$u_k = -\left[\left[\sum_{i=1}^{k} i \cdot F_i\right] + k \cdot \left[\sum_{i=k+1}^{N} F_i\right]\right], \quad k = 1, 2, \dots, N. \tag{37}$$

The software implementation of the method reduces to simple multiplication of the inverse matrix by the right-hand side vector.

Figure 3 illustrates the results of the numerical experiment.



(a) Exact solution

(b) Calculation error

Figure 3. The maximal error in this case at point $x = 0.785$ is $1.69E - 04$ for $N = 100$ and reduces to $1.69E - 6$ for $N = 1000$

5.2. Example

An example of the previous section is considered, which differs only in that the boundary conditions set earlier at the left end of the interval are transferred to the right and vice versa. The software implementation of the algorithm consists of several lines: filling in the vector of the right-hand side (34) and multiplying the inverse matrix B by this vector using Eq. (37).

6. Conclusion

The paper gives examples of practical problems, in the simulation of which it is necessary to solve second-order elliptic equations with different boundary conditions. The case of the one-dimensional Poisson equation and its finite-difference solution are described in detail. Estimates of the complexity of the sweep algorithm in the case of a uniform grid are given. An approach to solving the one-dimensional Poisson equation using explicitly calculated coefficients of inverse matrices for various types of boundary conditions is also described. The Dirichlet and Neumann boundary conditions in various combinations are considered.

A comparative analysis of the computational complexity of methods for solving the one-dimensional Poisson equation, based on the use of the sweep method and methods using an explicit representation of inverse matrices is presented.

Direct calculation shows that to implement calculations by right-sweep formulas, approximately $8N$ arithmetic operations are required, whereas in the Gauss method for fully filled matrices this number is approximately $(2/3)\,N^3$. It is also important that the tridiagonal structure of the matrix of the system makes it possible to use for its storage only an array of real variables of dimension $3N - 2$.

The assertion of the author of Ref. [4] that the method he proposed using the explicit form of inverse matrices allows solving the Poisson equations with different boundary conditions faster and more accurately is, to put it mildly, incorrect. Provided that the stability conditions of the sweep method are met, the speed of solving the problem by the sweep (Thomas) method is an order of magnitude higher due to a much smaller number of required operations.

However, unlike the sweep method [13], the practical implementation of the proposed method does not imply the allocation of additional arrays for software implementation, since the elements of the inverse matrix have a very simple form and their calculation within the loop determining the components of the solution is not difficult.

References

[1] A. N. Tikhonov and A. A. Samarskii, *Equations of Mathematical Physics [Uravneniya matematicheskoy fiziki]*, 7th ed. Moscow: Moscow State University, Nauka, 2004, in Russian.

[2] D. A. Yakovlev, V. G. Chigrinov, and H. S. Kwok, *Modeling and optimization of LCD optical performance*. New York: Wiley, 2015. DOI: 10.1002/9781118706749.

[3] L. N. Trefethen, *Approximation theory and approximation practice*. Philadelphia: SIAM – Society for Industrial and Applied Mathematics, 2019.

[4] M. Planitz *et al.*, *Numerical Recipes: The Art of Scientific Computing*, 3rd ed. New York: Cambridge University Press, 2007.

[5] N. N. Kalitkin and P. V. Koryakin, “Numerical methods [Chislennyye metody],” in *Methods of Mathematical Physics*, 1st ed. Moscow: Academia, 2013, vol. 2, in Russian.

[6] A. A. Abramov and V. B. Andreyev, “On the application of the method of successive substitution to the determination of periodic solutions of differential and difference equations,” *USSR Computational Mathematics and Mathematical Physics*, vol. 3, no. 2, pp. 498–504, 1963. DOI: 10.1016/0041-5553(63)90034-x.

[7] A. A. Samarskiy and A. V. Gulin, *Numerical Methods of Mathematical Physics [Chislennyye metody matematicheskoy fiziki]*. Moscow: Scientific world, 2003, in Russian.

[8] L. H. Thomas, *Elliptic problems in linear difference equations over a network*. New York: Waston Sci. Comput. Lab. Rept., Columbia University, 1949.

[9] S. B. Gueye, K. Talla, and C. Mbow, “Generalization of the exact solution of 1D Poisson equation with robin boundary conditions, using the finite difference method,” *Journal of Electromagnetic Analysis and Applications*, vol. 6, no. 12, pp. 372–381, 2014. DOI: 10.4236/jemaa.2014.612038.

[10] S. B. Gueye, K. Talla, and C. Mbow, “Solution of 1D Poisson equation with Neumann–Dirichlet and Dirichlet–Neumann boundary conditions, using the finite difference method,” *Journal of Electromagnetic Analysis and Applications*, vol. 6, no. 10, pp. 309–318, 2014. DOI: 10.4236/jemaa.2014.610031.

[11] S. B. Gueye, “The exact formulation of the inverse of the tridiagonal matrix for solving the 1D Poisson equation with the finite difference method,” *Journal of Electromagnetic Analysis and Applications*, vol. 6, no. 10, pp. 303–308, 2014. DOI: 10.4236/jemaa.2014.610031.

[12] N. N. Kalitkin and E. A. Alshina, “Numerical Methods [Chislennyye metody],” in *Numerical analysis*. Moscow: Academia, 2013, vol. 1, in Russian.

[13] A. Amosov, Y. Dubinsky, and N. Kopchenova, *Computational Methods [Vychislitel’nyye metody]*, 4th ed. St. Petersburg: Lan’, 2021, in Russian.

For citation:

S. Ndayisenga, L. A. Sevastianov, K. P. Lovetskiy, Finite-difference methods for solving 1D Poisson problem, Discrete and Continuous Models and Applied Computational Science 30 (1) (2022) 62–78. DOI: 10.22363/2658-4670-2022-30-1-62-78.

Information about the authors:

Sevastianov, Leonid A. — Doctor of Physical and Mathematical Sciences, Professor of Department of Applied Probability and Informatics of Peoples’ Friendship University of Russia (RUDN University), Leading Researcher of Bogoliubov Laboratory of Theoretical Physics, JINR (e-mail: sevastianov-la@rudn.ru, phone: +7(495)9522572, ORCID: https://orcid.org/0000-0002-1856-4643, ResearcherID: B-8497-2016, Scopus Author ID: 8783969400)

Lovetskiy, Konstantin P. — Candidate of Physical and Mathematical Sciences, Associate Professor of Department of Applied Probability and Informatics of Peoples’ Friendship University of Russia (RUDN University) (e-mail: lovetskiy-kp@rudn.ru, phone: +7(495)9522572, ORCID: https://orcid.org/0000-0002-3645-1060 , ResearcherID: A-5725-2017, Scopus Author ID: 18634692900)

Ndayisenga, Serge — Student of Department of Applied Probability and Informatics of Peoples’ Friendship University of Russia (RUDN University) (e-mail: 1032195775@rudn.ru, phone: +7(977)9086946, ORCID: https://orcid.org/0000-0002-9297-9839, ResearcherID: AAC-3303-2022)

УДК 535:535.3:681.7
DOI: 10.22363/2658-4670-2022-30-1-62-78

Конечно-разностные методы решения 1D задачи Пуассона

С. Ндайисенга[1], Л. А. Севастьянов[1,2], К. П. Ловецкий[1]

[1] *Российский университет дружбы народов*
ул. Миклухо-Маклая, д. 6, Москва, 117198, Россия
[2] *Лаборатория теоретической физики им. Н. Н. Боголюбова*
Объединённый институт ядерных исследований
ул. Жолио-Кюри, д. 6, Дубна, Московская область, 141980, Россия

Аннотация. В статье обсуждается постановка и анализ методов решения одномерного уравнения Пуассона на основе конечно-разностных аппроксимаций — важного и очень полезного инструмента численного исследования дифференциальных уравнений. По сути, это классический метод аппроксимации, основанный на разложении решения в ряд Тейлора. Развитие теоретических и практических результатов на базе этого метода в последние годы позволили повысить точность, стабильность и сходимость методов решения дифференциальных уравнений. Некоторые особенности этого анализа включают интересные расширения классического численного анализа начальных и граничных задач. В первой части излагается численный метод решения одномерного уравнения Пуассона, сводящийся к решению системы линейных алгебраических уравнений (СЛАУ) с ленточной симметричной положительно определённой матрицей. В качестве метода решения СЛАУ используется широко известный метод прогонки (метод Томаса). Во второй части представлен метод решения, основанный на аналитическом представлении точной обратной матрицы дискретизированного варианта уравнения Пуассона. Выражения для обратных матриц существенно зависят от типов граничных условий в исходной постановке. Представлены варианты обратных матриц для уравнения Пуассона с различными граничными условиями на концах исследуемого интервала — условиями Дирихле на обоих концах интервала, условиями Дирихле на одном из концов и Неймана на другом. Во всех трёх случаях коэффициенты обратных матриц легко вычисляются (выписываются) и алгоритм решения задачи практически сводится к умножению матрицы на вектор правой части.

Ключевые слова: 1D уравнение Пуассона, метод конечных разностей, обращение трехдиагональной матрицы, алгоритм Томаса, исключение Гаусса