Method Article

Parallel multigrid method for solving inverse problems

H.K. Al-Mahdawi\textsuperscript{a,b}, A. I Sidikova\textsuperscript{b}, Hussein Alkattan\textsuperscript{b}, Mostafa Abotaleb\textsuperscript{b,*}, Ammar Kadi\textsuperscript{c}, El-Sayed M El-kenawy\textsuperscript{d}

\textsuperscript{a}Electronic Computer Centre, University of Diyala, Diyala 132001, Iraq
\textsuperscript{b}Department of System Programming, South Ural State University, Chelyabinsk 454080, Russia
\textsuperscript{c}Department of Food and Biotechnology, South Ural State University, 454080 Chelyabinsk, Russia
\textsuperscript{d}Department of Communications and Electronics, Delta Higher Institute of Engineering and Technology, Mansoura, 35111, Egypt

\textbf{A B S T R A C T}

We considered in this work the linear operator equation and used the Landweber iterative method as an iterative solver. After that, we used the multigrid method as an optimization method for obtaining an approximation solution with a highly accurate and fast process. A new parallel algorithm for the multigrid process has been developed. The proposed algorithm is based on a V-cycle mixed with the two-grid method. This modification of the V-cycle provides for parallel computing for each level. A coarse grid operator with a residual right-hand side vector for each coarse grid is provided. This parallel algorithm is used to accelerate and enhance computation for the solution of the iteration method in solving the inverse ill-posed problems. The necessary cost-time computation for all stages and processes for the parallel V-cycle algorithm has been done. A numerical experiment on solving the IVP (initial value problem) for the heat equation showed that the new parallel algorithm is much more efficient than the sequential method.

- The study of iteration algorithms and mathematical experiments reveals a slow rate of convergence.
- The Multigrid method is often used to speed up the rate of convergence of iterative methods, which is an effective method of solving large systems of linear algebra equations.
- The approximation solution for the linear algebra equations was found by using the parallel method with the multigrid method.

© 2022 The Author(s). Published by Elsevier B.V.

This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/)

\textbf{A R T I C L E  I N F O}

Method name: Parallel Multigrid Method
Keywords: Multigrid, Inverse problem, Iteration method, Parallel
Article history: Received 22 July 2022; Accepted 15 October 2022; Available online 1 November 2022

* Corresponding author.
\textit{E-mail address: abotalebmostafa@bk.ru} (M. Abotaleb).

\url{https://doi.org/10.1016/j.mex.2022.101887}

2215-0161/© 2022 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/)
Introduction

We defined a parallel multigrid algorithm for solving IP (Inverse Problems). In this article, we consider the initial value problem of the one-dimensional (1-D) heat equation as a model of an ill-posed inverse problem. In the early twentieth century, Hadamard [1] labeled situations as well-posed problems, stating that a problem was well-posed when it fulfilled the following points:

1 A solution exists.
2 uniqueness. This solution is unique.
3 Stability (the given data is continuously dependent on the solution).

If at least one of the above points or conditions is not fulfilled in the problem, the problem is considered as an ill-posed problem. The violations of 1 and 2 can often be improved with a small re-formulation of the problem. Violations of stability are much harder to remedy because they imply that a small disturbance in the data leads to a large disturbance in the estimated solution [2–5]. Various methods and algorithms for solving IP “inverse problems” have been explained and used in [6–16]. The success of these methods and algorithms is largely based on understanding and analyzing the mathematical problems related to the declarations of the properties of these IP “inverse problems” and identifying specific difficulties in solving them [17–22].

One of the many known methods of solving a linear operator equation is the iteration method. This method has several shortcomings, including initial guessing, complexity, and convergence. The discretization process for the integral equation primes to a huge, sparse, and high-dimensional system of linear equations or linear operator equation. Inappropriately, theoretic convergence studies and mathematical experimentation show that the convergence rate is slow for these iterative algorithms, leading to an amplified cost of iteration. The direct iteration method suffers from some restricting boundaries. Multigrid methods advanced from efforts to overcome these limitations. Multigrid settings are largely successful when used in conjunction with relaxation or iteration methods, and they lead to fast and direct points to estimated solutions to solve the IP “inverse problems”.

Multigrid methods [23–25] are frequently used to accelerate the convergence rate of iterative methods [26]. The main idea of this work is to apply parallel computing to the classical multigrid method and consider the cost of parallel and equational computing. All these steps have been applied through the following sections. In Section 2 we defined the inverse problem for linear operator equations with the necessary mathematical notions. In section 3, the classical multigrid method has been defined with some classical algorithms. The new parallel multigrid method has been defined with the cost of computing. In the last section, the numerical examples have been applied and many results have been obtained.

Problem statement

Let’s define the operator $A \in H$ and the vectors $u, f \in H$ then let’s consider the linear system of equations:

$$Au = f.$$  (1)
A is linear a compact operator $A : H \to H$. We are interested in cases in which the operator $A$ is obtained from the discretization of an ill-posedness, such as the integral equation of first kind. Express the ill-posed problem of finding an approximate solution to the Eq. (1) as shown in the following. Suppose that $f = f_0$ there exists an real solution $u_0$ of problem (1) and this solution is related to the set $M_r$ which is defined as the class of correctness of (1),

$$M_r = \{ u_0 : u_0 \in H, \| u_0 \| \leq r \},$$

but the vector $f_0$ is unknown. Instead $f_\delta$ and error level or measurement noise $\delta$ are given such that

$$\| f_\delta - f_0 \| \leq \delta.$$  

(3)

Using the given information $M_r, f_\delta$, and $\delta$, we go to find the estimated solution $u_\delta$ of (1) and compute the deviation from the real solution $\| u_\delta - u_0 \|$. The iterative regularization method Landweber [27] used to find the estimated solution $u_\delta$ defined by next formula

$$u_{\delta}^{k+1} = u_\delta^k + \alpha A^* E(f_\delta - Au_\delta^k), \quad k = 0, 1, \ldots,$$

where $\alpha$ is defined as the regularization parameter and $E$ is the identity operator. The classical Landweber iterative method (CLI) can be defined by the following algorithm.

Algorithm 2 CLI (iteration number)

1. $u^0 = \vec{0}$, zeros vector.
2. $u^k = u^0$
3. for $k = 1, 2, 3, \ldots$ iteration number
   a. $u^k = u^k + \alpha A^T (f_\delta - Au^k)$
   b. variational $= \| Au^k - f_\delta \|^2$
4. End for loop

There are two significant measures of $u_\delta$ as an approximation of $u_0$ the error and residual. An iterative method for the solution of (1) can be applied directly to the Eq. (1) or to an equation of the error. It is called residual equation [28]. Let $u_\delta$ be an approximation of $u_0$, then the error $e = u_0 - u_\delta$ satisfies the equation

$$Ae = f_\delta - Au_\delta = r.$$  

(5)

where $r$ is the residual, e.g. the volume by which the estimate $u_\delta$ fails to satisfy the original problem (1).

**Multigrid method**

Multigrid method (MG) may be observed as a class of iterative solvers with the fast property of linear systems of equations with large sparse matrices ascending from PDE or integral equation discretization. The iterative solvers have nice numerical properties, such as low memory demand of $O(N)$. However, these methods have a serious drawback which slows down convergence with the increase of discretization resolution the smoothing property.

Often, iterative solutions reduce high-frequency error components, which are related to the toughest variations of the error spread on the grid or domain. Low frequency (low varying) parts aren’t taken out, and the error doesn’t go away, but instead becomes smooth.

Moreover, the grid size increases or domain diminution leads to the oscillation of the slowly rotting smooth error, making it accordingly extra vulnerable. Coarsening the grid to accelerate the convergence is very effective, but it spoils the discretization accuracy. From this point, we need to define the grid with diminution of that grid $h$ and the equivalent linear system of equations or linear operator equation as the next equation

$$A^h u^h = f^h.$$  

(6)

The method for improving the performance of iterative methods, at least at the beginning of the iteration, starts with a good initial iteration. For the typical problem, we can find a good initial
iteration by explaining the nearly solution for the problem on the coarse grid and by using a few iterations, which is called smoothing.

Let us study a grid $\Omega^{2h}$. In practice, uniform refinement consists in dividing in halves all intervals of $\Omega^{2h}$, leading to the grid $\Omega^h$, see Fig. 1.

The nested iteration is the strategy of using coarse grid problems to improve the iterative method to solve (6):

- solve $A^h u^h = f^h$ on a very coarse grid approximately by applying a smoother (iteration).
- smooth $A^{2h} u^{2h} = f^{2h}$ on $\Omega^{2h}$.
- solve $A^h u^h = f^h$ on $\Omega^h$ by the iterative method with the initial iteration coming from the coarser grids.

Coarse grid correction (two-level method) is the second strategy and also uses the residual Eq. (5) (see Fig. 2):

- Smooth $A^h u^h = f^h$ on $\Omega^h$. This step gives an approximation $u_0^h$ of the solution which still has to be updated appropriately. Compute the residual $r^h = f^h - A^h u_0^h$.
- Project by “restrict” residual vector to $\Omega^{2h}$. The result is $R(r^h)$.
- Solve $A^{2h} e^{2h} = R(r^h)$ on $\Omega^{2h}$. With this step, we obtain an approximation of the error, $e^{2h}$.
- Project “prolongate” $e^{2h}$ to $\Omega^h$. The result is denoted by $P(e^{2h})$.
- Update the approximation of the solution on $\Omega^h$ by $u^h = u_0^h + P(e^{2h})$

The transfer from a coarse grid to a fine grid is called prolongation or interpolation. In many situations, we can use simplest approach, which is the linear interpolation. Let $\Omega^{2h}$ be divided into $N = 2$ intervals and $\Omega^h$ into $N$ intervals. The node $j$ on $\Omega^{2h}$ corresponds to the node $2j$ on $\Omega^h$, $0 \leq j \leq N/2$, Let $u^{2h}$ be given on $\Omega^{2h}$. Then, the linear interpolation

$$f_{2h}^h : \mathbb{R}^{N/2-1} \rightarrow \mathbb{R}^{N-1}, u^h = f_{2h}^h u^{2h},$$

is given by
\[ u^h_{2j} = u^{2h}_j, \quad j = 1, \ldots, N/2 - 1, \]
\[ u^h_{2j+1} = \frac{1}{2} \left( u^{2h}_j + u^{2h}_{j+1} \right), \quad j = 0, \ldots, N/2 - 1. \]  

(see Fig. 3). For even points or nodes in grid \( \Omega^h \), takes the value of the equivalent point in grid \( \Omega^{2h} \). About the odd nodes in grid \( \Omega^h \), the values of the neighbor points is computed.
It can be represented as operator. Using the standard basis of \( \mathbb{R}^{N/2-1} \) and \( \mathbb{R}^{N-1} \), then,

\[
I_h^2 u^{2h} = \frac{1}{2} \begin{bmatrix}
1 & 2 & 1 \\
2 & 1 & \ddots \\
\ddots & \ddots & \ddots \\
1 & 2 & 1
\end{bmatrix} \begin{bmatrix}
u_1 \\
u_2 \\
\vdots \\
u_{(n-1)\times(\frac{J}{2}-1)}
\end{bmatrix} = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_{\frac{J}{2}-1} \\ u_{n-1} \end{bmatrix} = u^h. \tag{8}
\]

Suppose that error (unknown) is a smooth vector on the grid \( \Omega^h \). In addition, the coarse grid estimates on \( \Omega^{2h} \) is calculated, and it should be exact in the coarse grid. The interpolation of this coarse grid evaluation is a smooth vector on the fine grid. For this reason, the smooth error on the fine grid has supposed.

If there is a fault on the fine grid, then each interpolation of a coarse grid estimate to the fine grid is a smooth function and we cannot expect that the error on the fine grid will come close to being fixed. See Fig. 4.

Perturbation provides good results if the fault on the fine grid is smooth. Hence, prolongation is an appropriate match for the smoother, which works best professionally if the error is wavering.

For the two-level method, we must do the transmission action for the residual vector from \( \Omega^h \) to \( \Omega^{2h} \) before the coarse grid equation can be solved. This transmission is called restriction. The simplest restriction is injection. It is defined by

\[
i_h^2 : \mathbb{R}^{N-1} \rightarrow \mathbb{R}^{N/2-1}, u^{2h} = i_h^2 u^h
\]

\[
u_{2j}^h = u_{2j}^h, j = 1, \ldots, N/2 - 1,
\]

(see Fig. 5). In this method, for each node on the coarse grid, we take the value of the grid function at the corresponding node on the fine grid.

Injection is an efficient method. If we ignore every other node on \( h \), then the values of the residual in these nodes (and the error in these nodes) have no impact on the system on the coarse grid. Consequently, these errors will generally not be corrected.

There is a weighted restriction for finite difference schemes. The weighted restriction uses all the nodes on the fine grid. It is defined by an appropriate average

\[
i_h^2 : \mathbb{R}^{N-1} \rightarrow \mathbb{R}^{N/2-1}, \]

\[
u^{2h} = i_h^2 u^h, \quad u_{2j}^{2h} = \frac{1}{4} \left( u_{2j-1}^h + 2u_{2j}^h + u_{2j+1}^h \right), j = 1, \ldots, \frac{N}{2} - 1. \tag{9}
\]

If the spaces \( \mathbb{R}^{N-1} \) and \( \mathbb{R}^{N/2-1} \) are equipped with the standard bases, the matrix representation of the weighted restriction operator appears as follows:

\[
i_h^2 u^h = \frac{1}{4} \begin{bmatrix}
1 & 2 & 1 \\
2 & 1 & \ddots \\
\ddots & \ddots & \ddots \\
1 & 2 & 1 \\
1 & 2 & 1
\end{bmatrix} \begin{bmatrix}
u_1 \\
u_2 \\
\vdots \\
u_{\frac{J}{2}-1} \\ u_{n-1}
\end{bmatrix} = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_{\frac{J}{2}-1} \\ u_{n-1} \end{bmatrix} = u^{2h}. \tag{10}
\]

With this representation, we can see the important connection between weighted restriction \( i_h^2 \) and interpolation \( i_h^2 : i_h^2 \rightarrow 2(i_h^2)^T \).
Fig. 4. Oscillating error and interpolation.
The operator on a coarse grid is defined as the Galerkin projection operation. The operation is started from the derivation of an appropriate coarse grid operator, by the Galerkin operator which is used by the residual formula as next

$$A^h v^h = r^h.$$  \hfill (11)

Let us assume that $e^h$ is founded in the range of the prolongation operator $f_{2h}^h$. Then, there is a vector $e^{2h}$ defined on the next coarse grid as following

$$e^h = f_{2h}^h (e^{2h}).$$  \hfill (12)

Replacing (12) in (11), we obtain

$$A^{h} f_{2h}^h (e^{2h}) = r^h.$$  \hfill (13)

By running the restriction action to both sides of (13) returns

$$l_{h}^{2h} A^{h} f_{2h}^h (e^{2h}) = l_{h}^{2h} r^h,$$  \hfill (14)

leading to the definition

$$A^{2h} (e^{2h}) = r^{2h},$$  \hfill (15)

where

$$A^{2h} = l_{h}^{2h} A^{h} f_{2h}^h.$$  \hfill (16)

The derivation of (16) was depended on the error $e^h$ is in the range of the prolongation. This property in general is not specified. If it true, then a real solution in coarse grid provide a solution to $A^h u^h = f^h$ with correction step on the coarse grid. However, this derivation gives a motivation for defining $A^{2h}$ as (16).

To make simpler notations, the residual equation for the right-hand side vector will be signified by $f^{2h}$ instead of $r^{2h}$. The solution on the finest grid will represented as $u^h$ and the present iterate as $v^h$. Instead of representing the solution on the coarse grid as $e^{2h}$, it will be represented as $u^{2h}$.

**Multigrid method V-cycle** is applied by imbedding the two-level method into itself. Let us assume that there some $l + 1$ grids, $l \geq 0$, where the finest grid has the size $h$ and this size increased by the factor 2 for each next or coarser grid. Let $L = 2^l$.

**Algorithm V-cycle**

- Pre smoothing: apply the smoother $V_1$ times to $A^h u^h = f^h$ with the initial guess $v^h$. The result is denoted by $v^h$.
- Compute $f^{2h} = l_{h}^{2h} (f^h - A^h v^h)$.
  - Apply the smoother $V_1$ times to $A^{2h} u^{2h} = f^{2h}$ with the initial guess $v^{2h} = 0$. Denote the result by $v^{2h}$. 

\[Fig. 5.\] Injection.
H.K. Al-Mahdawi, A.I. Sidikova and H. Alkattan et al. / MethodsX 9 (2022) 101887

Fig. 6. Multigrid V-cycle.

Fig. 7. Multigrid W-cycle. $s$ smoothing, $r$ restriction, $p$ prolongated, $e$ exact solver.

- Compute $f^{4h} = I^{4h}_{2h} (f^{2h} - A^{2h} u^{2h})$.
- Solve $A^{2h} u^{2h} = f^{2h}$.
- Correct $v^{2h} := v^{2h} + I^{2h}_{4h} u^{4h}$.
- Apply smoother V2 times to $A^{2h} u^{2h} = f^{2h}$ with the initial guess $v^{2h}$.
- Correct $v^{h} := v^{h} + P_{2h}^{h} u^{2h}$
- Post smoothing: apply smoother V2 times to $A^{h} u^{h} = f^{h}$ with the initial guess $v^{h}$.

The name V-cycle comes from their movement through the hierarchy of grids (see Fig. 6).

The V-cycle method is based on the applied recursion of the two-level algorithm for solving the coarse grid. The V-cycle algorithm does not provide an answer on how to solve the initial value in the pre-smooth section. There are some other methods, such as the W-cycle (see Fig. 7).

If we use on each grid (besides the finest grid) one multigrid V-cycle for smoothing, a full multigrid V-cycle is performed (see Fig. 8).

The above methods run sequentially, which requires a great deal of computation time, especially when the fine grid is the highest level. For this reason, we define parallel computing for the V-cycle method as shown in the next section.
Parallel multigrid method

The two-grid procedure defined above can be comprehensive to an arrangement or next grids through recursion, which leads to the modified FMG with V-cycle as shown in Fig. 6.

We suggest a parallel method which applies the two-grid method during pre-smoothing for the V-cycle method. In this method, we can compute the residual for each coarse grid independently prior to smoothing. This means that we can apply all pre-smooth computing in parallel as shown in Fig. 9.

The parallel multigrid method V-cycle is applied by imbedding the two-level method for each coarse grid prior to smoothing.

Algorithm PV-cycle

- Pre smoothing:
  - Apply two-grid to $A^h u^h = f^h$, the result is denoted by $v^h$ and $v^{2h}$
  - Apply two-grid to $A^{4h} u^{4h} = f^{4h}$, the result is denoted by $v^{4h}$ and $v^{8h}$
  - ...
  - Apply two-grid to $A^{(L)} u^{(L)} = f^{(L)}$, the result is denoted by $v^{Lh}$
  - Correct $v^{2h} := v^{2h} + I_{4h} v^{4h}$
- Correct $v^{h} := v^{h} + I_{2h} v^{2h}$
- Post smoothing: apply smoother $V_2$ periods to $A^h u^h = f^h$ with the initial vector $v^h$.

The parallel $V$-cycle becomes a parallel move through the hierarchy of grids, see Fig. 9

To compute the coarse grid operator for each domain independently for each two-grid in the coarse grid, we used the following formula.

$$A^{2^m h} = \prod_{j=0}^{m} A^{2^j h} \prod_{j=0}^{m} (I + \frac{1}{2^{j+1}}) \quad m = 1, 2, 3, \ldots \quad (17)$$

The same method is used to compute the $f$ right-hand side vector,

$$f^{2^m h} = \prod_{j=0}^{m} f^{2^j h} \quad m = 0, 1, 2, \ldots \quad (18)$$

The above-mentioned multigrid algorithms were used in a general-object to solve linear operator equations. In order to compute the computation time for running a parallel $V$-cycle, we divided the parallel $V$-cycle into two parts: post-smooth, which is implemented in parallel, and pre-smooth, which is implemented sequentially (see Fig. 10).
Fig. 12. Approximation solutions in $\Omega^1$.

(a) $\Omega^h = 1024$ with $\delta = 0.04$

(b) $\Omega^h = 2048$ with $\delta = 0.05$
The time for running each process \( P_{\text{finegrid,coarsegrid}} \) in the pre-smooth \( (T_{\text{pre-smooth}}) \) for each two-grid process is calculated as:

\[
T_{\text{pre-smooth}} = \max \left\{ P_{2h,2^{j+1}h} \right\}, \quad j = 0, 1, \ldots, m.
\] (19)

We defined the next equation to compute the time for each two-grid process \( (S_h \text{smoothing in a fine grid, } S_{2h} \text{ smoothing in a coarse grid, restriction, and prolongation})\):

\[
P_{2h,2^{j+1}h} = 2T(S_{2h}) + T(S_{2^{j+1}h}) + T(2^{j+1}h) + T(2^{j+1}2^{j+1}h), \quad j = 0, 1, \ldots
\] (20)

where \( T(S_{2h}) \), returned the time for smoothing, and \( T(2^{j+1}h) \) returned the time for computing the restriction and prolongation operations, respectively.

Regarding post-smoothing computation time \( T_{\text{post-smooth}} \), the time necessary to complete prolongation with last smooth step can be calculated as:

\[
T_{\text{post-smooth}} = m \sum_{j=0}^{m} T(P_{2h,2^{j+1}h}) + S_h.
\] (21)

**Numerical experiment and conclusion**

The IVP “initial value problem” for heat problem was studied in work [2, 4]. The mathematical form of IVP has been labeled by the following linear PDE system:

\[
\frac{\partial u(x, t)}{\partial t} = \frac{\partial^2 u(x, t)}{\partial x^2} \quad 0 \leq x \leq l, \quad t \in (0, T],
\] (22)

\[
u(x, t) = 0, \quad t \in (0, T],
\] (23)

\[
u(l, t) = 0, \quad t \in (0, T].
\] (24)
where \( u(0, t) \) and \( u(l, t) \) are boundary conditions and \( u_0(x) \) is the IC “initial condition” which is needing to be found. In this work, we apply other numerical algorithms to obtain a more accurate solution and fast implementation for a high-scale problem. The integral form for this equation will be:

\[
Au(x) = \int_0^l K(x, y) u(y) \, dy = f(x),
\]

where the kernel \( K(x, y) \) is an infinite series. Since we cannot hold the infinite sum, we do 10 times for the sum of series:

\[
K(x, y) = \frac{2}{\pi} \sum_{n=1}^{10} e^{-\frac{(n\pi)^2}{l^2}} \sin\left(\frac{n\pi x}{l}\right) \sin\left(\frac{n\pi y}{l}\right), \quad T > 0.
\]

To obtain an estimated solution to \( u(x) \), we can reformulate the problem as a linear operator equation \( Au = f \). After applying the discretization algorithm described in [5] we obtain

\[
A \begin{bmatrix} u(y_0) \\ u(y_2) \\ \vdots \\ u(y_{n-1}) \end{bmatrix} = \begin{bmatrix} f(x_0) \\ f(x_2) \\ \vdots \\ f(x_{n-1}) \end{bmatrix},
\]

where

\[
A = \frac{1}{n} \begin{bmatrix}
K(x_0, y_0) & K(x_0, y_1) & \cdots & K(x_0, y_{n-1}) \\
K(x_1, y_0) & K(x_1, y_1) & \cdots & K(x_1, y_{n-1}) \\
\vdots & \vdots & \ddots & \vdots \\
K(x_{n-1}, y_0) & K(x_{n-1}, y_1) & \cdots & K(x_{n-1}, y_{n-1})
\end{bmatrix}.
\]

The bounded and injective operator \( A \) is ill-conditioned, so any numerical attempt to directly solve (28) will be fail.

Considering the inverse IVP (22)–(25) for the heat equation, we must find \( u(x) \in L_2[0, 1] \). The exact solution will be \( u(x) = \sin \pi x, 0 \leq x \leq 1 \). We created the input vector or right-hand side for IP “inverse problem” \( u(x, T) = f(x), 0 \leq x \leq 1 \) and \( T = 0.01 \).

In this numerical example we will use the Landweber method and the same iterative method as in the V-cycle multigrid. We compared the 3 algorithms with high scale of domains \( \Omega^h \); the result is shown in Table 1 for a domain of \( \Omega^h = 1024 \) and Table 2 for domain size \( \Omega^h = 2048 \).

The V-cycle algorithm has been successfully applied to obtain a good approximating solution with low time in high scale domains \( \Omega^h \), see Fig. 12.

The Landweber iteration method has been successfully applied to obtain an estimated solution. The V-cycle multigrid method has been applied to reduce computation costs for the iterative method. The parallel technique has been successfully applied to the V-cycle multigrid method to obtain the approximation solution for the IVP for a heat equation.

**Conflicts of Interest**

The authors declare no conflict of interest.

**Data Availability**

No data was used for the research described in the article.
Acknowledgement

The work was supported by Act 211 Government of the Russian Federation, contract No. 02.A03.21.0011. The work was supported by the Ministry of Science and Higher Education of the Russian Federation (government order FENU-2020-0022).

References

[1] P.J. Daniell, J. Hadamard, Lectures on Cauchy's problem in linear partial differential equations, Math. Gaz. 12 (171) (1924) 173, doi: 10.2307/3603014.
[2] H.K. Al-Mahdawi, Studying the Picard's method for solving the inverse cauchy problem for heat conductivity equations, Bull. South Ural State Univ. Ser. “Computational Math. Softw. Eng. 8 (4) (2019) 5–14, doi: 10.14529/cmse190401.
[3] H.K. Al-Mahdawi, Development the regularization computing method for solving boundary value problem to heat equation in the composite materials, Journal of Physics: Conference Series 1999 (1) (2021) 12136.
[4] H.K. Al-Mahdawi, Development the numerical method to solve the inverse initial value problem for the thermal conductivity equation of composite materials, J. Phys. Conf. Ser. 1879 (3) (May 2021) 32016, doi: 10.1088/1742-6596/1879/3/032016.
[5] H.K. Al-Mahdawi, A.I. Sidikova. An approximate solution of fredholm integral equation of the first kind by the regularization method with parallel computing, Turkish J. Comput. Math. Educ. 12 (12) (2021) 4582–4591.
[6] V.P. Tanana, Completeness of the system of eigenfunctions of the Sturm–Liouville problem with the singularity, Vestn. Udmurt. Univ. Mat. Mekhanika. Komp’yuternye Nauk. 30 (1) (2020) 59–63.
[7] V.P. Tanana, A.I. Sidikova, Optimal Methods for Ill-Posed Problems: With Applications to Heat Conduction, 62, Walter de Gruyter GmbH & Co KG, 2018.
[8] V.P. Tanana, B.A. Markov, Uniqueness, existence and solution of the direct boundary heat exchange problem for a weakly non-linear temperature conductivity coefficient, Journal of Physics: Conference Series 1715 (1) (2021) 12050.
[9] V.P. Tanana, Optimization of methods for solving inverse problems, in: 2018 International Russian Automation Conference (RussAutoCon), 2018, pp. 1–7.
[10] V.P. Tanana, B.A. Markov, The control problem for the heat equation in the case of a composite material, Journal of Physics: Conference Series 1715 (1) (2021) 12049.
[11] V.P. Tanana, A.I. Sidikova, A.A. Ershova, A numerical solution to a problem of crystal energy spectrum determination by the heat capacity dependent on a temperature, Eurasian J. Math. Comput. Appl. 5 (1) (2017) 87–94.
[12] V.P. Tanana and T.N. Rudakova, “The optimum of the MM Lavrent’ev method,” 2011.
[13] V.K. Ivanov, V.V. Vasin, V.P. Tanana, Theory of Linear Ill-Posed Problems and Its Applications, 36, Walter de Gruyter, 2013.
[14] V.P. Tanana, A comparison of error estimates at a point and on a set when solving ill-posed problems, J. Inverse Ill-posed Probl. 26 (4) (2018) 541–550.
[15] V.P. Tanana, On Reducing an Inverse Boundary-Value Problem to the Synthesis of Two Ill-Posed Problems and Their Solution, Numer. Anal. Appl. 13 (2) (2020) 180–192.
[16] V.P. Tanana, E.Y. Vishnyakov, A.I. Sidikova, An approximate solution of a Fredholm integral equation of the first kind by the residual method, Numer. Anal. Appl. 9 (1) (2016) 74–81, doi: 10.1134/S1995423916010080.
[17] V.B. Glasko, N.I. Kulik, N.I. Shiklyarov, A.N. Tikhonov, An inverse problem of heat conductivity, Zhurnal Vychislitel’noi Mat. i Mat. Fiz. 19 (3) (1979) 768–774.
[18] A.S. Belonosov, M.A. Shishlenin, Continuation problem for the parabolic equation with the data on the part of the boundary, Siber. Electron. Math. Rep. 11 (2014) 22–34.
[19] S.I. Kabanikhin, A. Hasanov, A.V. Penenko, A gradient descent method for solving an inverse coefficient heat conduction problem, Numer. Anal. Appl. 1 (1) (2008) 34–45.
[20] A.G. Yagola, I.E. Stepanova, Y. Van, V.N. Titarenko, Obratnye zadachi i metody ikh resheniya. Prilozheniya k geofizike, Inverse Probl. Methods their Solut. Appl. to Geophys. Moscow Binom. Lab. znannii (2014).
[21] S.I. Kabanikhin, O.I. Krivorot’ko, M.A. Shishlenin, A numerical method for solving an inverse thermoacoustic problem, Numer. Anal. Appl. 6 (1) (2013) 34–39.
[22] V.P. Tanana, On the order-optimality of the projection regularization method in solving inverse problems, Sib. Zhurnal Ind. Mat. 7 (2) (2004) 117–132.
[23] K. Stüben and U. Trottenberg, “Multigrid methods: fundamental algorithms, model problem analysis and applications,” pp. 1–176, 1982, doi: 10.1007/bfb009928.
[24] P. Wesseling, Introduction To Multigrid Methods, Institute for Computer Applications in Science and Engineering Hampton VA, 1995.
[25] W.L. Briggs, V.E. Henson, and S.F. McCormick, A multigrid tutorial. SIAM, 2000.
[26] H.K.I. Al-Mahdawi, M. Abotaleb, H. Alkattan, A.-M.Z. Tareq, A. Badr, A. Kadi, Multigrid method for solving inverse problems for heat equation, Mathematics 10 (15) (2022) 2802.
[27] L. Landweber, An iteration formula for fredholm integral equations of the first kind, Am. J. Math. 73 (3) (1951) 615, doi: 10.2307/2372313.
[28] H.K.I. Al-Mahdawi, H. Alkattan, M. Abotaleb, A. Kadi, E.-S.M. El-kenawy, Updating the landweber iteration method for solving inverse problems, Mathematics 10 (15) (2022) 2798.