1. INTRODUCTION

Diffusion processes have to be taken into account in the numerical simulation of many problems in mathematical physics, for example, problems in gas and fluid dynamics. Allowance for diffusion processes leads to the necessity of solving parabolic equations. When they are approximated by an explicit scheme, a severe restriction is imposed on the time step in order to preserve the stability of the scheme [1, 2]. As a result, the computation time increases considerably. This restriction is eliminated in the case of implicit schemes, but they may also be unsuitable since the arising systems of linear algebraic equations are too difficult to solve. Note that, in some cases, classical multigrid methods [3] can also be associated with much computational work and do not provide considerable advantages in comparison with explicit schemes. Therefore, it is necessary to develop new algorithms for solving parabolic equations.

In [4, 5], an effective algorithm based on the multigrid method was proposed for parabolic equations. The algorithm involves only one iteration step of a two-grid cycle at every time level, and one or several iterations are performed at the smoothing stage. A theoretical and numerical study of model problems for the heat equation with continuous and discontinuous coefficients showed that the algorithm preserves the advantages of implicit schemes, such as stability and accuracy. However, the solution scheme in [4, 5] was not conservative. Moreover, for problems with a discontinuous thermal conductivity, one smoothing iteration step is usually insufficient to achieve acceptable accuracy of the solution and there is no stopping criterion for the iterative process at the smoothing stage.

In this paper, we propose a new effective multigrid algorithm that is free of the shortcomings inherent in the method of [4, 5]. Every time step of the algorithm involves one iteration of a two-grid cycle and one smoothing iteration. The algorithm is analyzed using an initial–boundary value problem for the heat equation. The conservativeness of the scheme is proved theoretically for a two-dimensional model problem. A one-dimensional model problem is used to theoretically prove the absolute stability of the algorithm with respect to the initial data and to examine the boundedness of the difference derivatives of the solution. The convergence and accuracy of the method are analyzed theoretically using two- and one-dimensional model problems. It is shown that the amount of arithmetic work at every time step of the algorithm is considerably less than in implicit schemes used on a fine grid. The good accuracy of the algorithm is demonstrated by computing two-dimensional model problems, including ones with discontinuous coefficients.
2. ALGORITHM FOR SOLVING DIFFUSION-TYPE EQUATIONS BASED ON A TWO-GRID METHOD

An algorithm for solving diffusion-type equations is constructed and analyzed using the following initial–boundary value problem for the two-dimensional heat equation:

\[\rho C_v \frac{\partial T}{\partial t} = \text{div}(\kappa \text{grad } T) + f, \quad (x, y, t) \in G = \{0 < x < l_1, 0 < y < l_2, 0 < t \leq T\},\]

\[T(x, y, t) = g(x, y, t) \quad \text{for} \quad (x, y) \in \gamma,\]

\[T(x, y, 0) = T_0(x, y),\]

where \(C_v\) is the heat capacity at constant volume, \(\rho\) is the density, \(\kappa\) is the thermal conductivity, \(T\) is the temperature at the point \((x, y)\) at the time \(t\), \(f\) is the density of the heat sources, \(\gamma\) is the boundary of the computational domain, and \(g(x, y, t)\) and \(T_0(x, y)\) are given functions. Problem (2.1) is approximated by the purely implicit difference scheme

\[
(C_v \rho)_{ij} \frac{u_{ij}^{n+1} - u_{ij}^n}{\tau} = \kappa_{i+0.5,j} \frac{u_{i+1,j}^n - u_{ij}^n}{h_x^2} - \kappa_{i-0.5,j} \frac{u_{ij}^n - u_{i-1,j}^n}{h_x^2} + \kappa_{ij+0.5} \frac{u_{ij}^n - u_{ij+1}^n}{h_y^2} - \kappa_{ij-0.5} \frac{u_{ij}^n - u_{ij-1}^n}{h_y^2} + \Phi_{ij},
\]

\[0 < i \leq N_1, \quad 0 < j \leq N_2,\]

\[u_{ij}^{n+1} = T_0(x_i, y_j), \quad 0 \leq i \leq N_1, \quad 0 \leq j \leq N_2,\]

\[u_{i0}^n = u_1(t_{n+1}, y_j), \quad u_{N_2}^n = u_2(t_{n+1}, y_j),\]

\[u_{io}^n = u_3(t_{n+1}, x_i), \quad u_{iN_2}^n = u_4(t_{n+1}, x_i), \quad 0 < i < N_1, \quad 0 < j < N_2,\]

where \(u_{ij}^n\) and \(u_{ij}^{n+1}\) are the solutions at the \((n+1)\)th and \(n\)th time levels, \(h_x\) and \(h_y\) are the constant mesh sizes in \(x\) and \(y\), and \(\tau\) is the step in \(t\). Difference scheme (2.2) is a system of linear algebraic equations for the unknown values of the solution at the \((n+1)\)th time level:

\[A_h u^{n+1} = f_h.\]

In the method proposed in this work, the grid function values at the next time level are determined according to the following algorithm. At Stage I, a single smoothing simple iteration for Eqs. (2.2) or (2.3) is performed using the formula

\[
(u_{ij})^{s+1} = \sigma \left[ \frac{(C_v \rho)_{ij} u_{ij}^n}{\tau} + \kappa_{i+0.5,j} \frac{(u_{i+1,j}^n)^s + \kappa_{i-0.5,j} (u_{i-1,j}^n)^s}{h_x^2} + \kappa_{ij+0.5} \frac{(u_{ij}^n + 0.5(u_{ij+1}^n))^s + \kappa_{ij-0.5} (u_{ij-1}^n)^s}{h_y^2} + \Phi_{ij} \right]
\times \left( \frac{(C_v \rho)_{ij}}{\tau} + \kappa_{i+0.5,j} + \kappa_{i-0.5,j} \right) + (1 - \sigma)(u_{ij})^s,
\]

where \(i = 1, 2, \ldots, N_1 - 1; j = 1, 2, \ldots, N_2 - 1; 0 < \sigma \leq 1\) is a weighting factor; \(s = 0\); and \((u_{ij})^0 = u_{ij}^n\). The resulting grid function is denoted by \(u_{ij}^s\). Next, we calculate the residual

\[r = A_h u^s - f_h.\]

At Stage II, the residual is projected or restricted to a coarse grid by using the formulas

\[
R_{lp} = \frac{1}{4} r_{2i,2j} + \frac{3}{8} (r_{2i,1,2j} + r_{2i-1,2j} + r_{2i,2j+1} + r_{2i,2j-1})
\]

\[+ \frac{1}{16} (r_{2i+1,2j+1} + r_{2i-1,2j-1} + r_{2i-1,2j+1} + r_{2i+1,2j-1}),\]

where \(l = 1, 2, \ldots, N_l - 1\) and \(p = 1, 2, \ldots, N_p - 1\).