Mathematical Problems in Engineering
Volume 7, pp. 283–297
Reprints available directly from the publisher
Photocopying permitted by license only

New Schemes for a Two-dimensional Inverse Problem with Temperature Overspecification

MEHDI DEHGHAN*

Department of Applied Mathematics, Faculty of Mathematics and Computer Science, Amirkabir University of Technology, No. 424, Hafez Ave. Tehran, Iran

(Received 3 October 2000; In final form 22 November 2000)

Two different finite difference schemes for solving the two-dimensional parabolic inverse problem with temperature overspecification are considered. These schemes are developed for indentifying the control parameter which produces, at any given time, a desired temperature distribution at a given point in the spatial domain. The numerical methods discussed, are based on the (3,3) alternating direction implicit (ADI) finite difference scheme and the (3,9) alternating direction implicit formula. These schemes are unconditionally stable. The basis of analysis of the finite difference equation considered here is the modified equivalent partial differential equation approach, developed from the 1974 work of Warming and Hyett [17]. This allows direct and simple comparison of the errors associated with the equations as well as providing a means to develop more accurate finite difference schemes. These schemes use less central processor times than the fully implicit schemes for two-dimensional diffusion with temperature overspecification. The alternating direction implicit schemes developed in this report use more CPU times than the fully explicit finite difference schemes, but their unconditional stability is significant. The results of numerical experiments are presented, and accuracy and the Central Processor (CPU) times needed for each of the methods are discussed. We also give error estimates in the maximum norm for each of these methods.

Keywords: Inverse problem; Parabolic equations; Temperature overspecification; Two-dimensional diffusion; Alternating direction implicit finite difference schemes- modified equivalent partial differential equations

AMS Subject Classifications: 65N06, Mxx, D25

*e-mail: mdehghan@cic.aku.ac.ir

1. INTRODUCTION

The main purpose of this paper is to construct two alternating direction implicit finite difference methods that have an acceptable accuracy and stability behaviour.

Recently, the study of parabolic inverse problems *i.e.*, the determination of some un-known function $p(t)$ in a parabolic equation, has been received much more attention. This work is aimed at producing two finite difference schemes for the numerical solution of the inverse problem of finding a control parameter $p=p(t)$ in the linear time dependent diffusion equation:

$$\frac{\partial u}{\partial t}=\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}+p(t)u+\phi(x,y,t), \quad 0<x,y<1, \ 0<t, \tag{1}$$

with initial condition

$$u(x,y,0)=f(x,y), \quad 0\leq x,y\leq 1, \tag{2}$$

and boundary conditions

$$u(0,y,t)=g_0(y,t), \quad 0\leq t\leq T, \ 0<y\leq 1, \tag{3}$$

$$u(1,y,t)=g_1(y,t), \quad 0\leq t\leq T, \ 0<y\leq 1, \tag{4}$$

$$u(x,0,t)=h_0(x,t), \quad 0\leq t\leq T, \ 0<x\leq 1, \tag{5}$$

$$u(x,1,t)=h_1(x,t), \quad 0\leq t\leq T, \ 0<x\leq 1, \tag{6}$$

subject to the overspecification at a point in the spatial domain

$$u(x_0,y_0,t)=E(t), \quad 0<t\leq T, \ 0\leq x_0,y_0\leq 1. \tag{7}$$

where f, g_0, g_1, h_0, h_1, ϕ and E are known functions, while the functions u and p are unknown.

The Eq. (1) can be used to describe a heat transfer process with a source parameter present. Equation (7) can then be interpreted as the temperature at a given point (x_0,y_0) in the spatial domain at time t. Thus the purpose of solving this inverse problem is to identify the source control parameter that produces at any given time, a desired temperature at a given point (x_0,y_0) in the spatial domain.

The inverse problem above and other similar problems have been studied in one dimension by many authors, and in two dimensions by [9, 16]. This kind of problem has many important applications. The existence and uniqueness and continuous dependence of the solutions to this problem and also the applications are discussed in [1–9, 13–15].

This paper consists of 5 parts.

Section 2 deals with the (3,3) alternating direction implicit finite difference method and the (3,9) ADI scheme. The method of evaluating $p(t)$ is also described in Section 3. The numerical results for the test used is given in Section 4. The comparison of both accuracy and efficiency between the methods developed is given in Sections 2 and 4. Conclusions are also presented in the last section.

2. THE FINITE DIFFERENCE METHODS

The domain $[0,1]^2 \times [0, T]$ is divided into an $M^2 \times N$ mesh with the spatial step size $h = 1/M$ in both x and y direction and the time step size $k = T/N$ respectively.

Grid points (x_i, y_j, t_n) are defined by

$$x_i = ih, \quad i = 0, 1, 2, \ldots, M, \tag{8}$$

$$y_j = jh, \quad j = 0, 1, 2, \ldots, M, \tag{9}$$

$$t_n = nk, \quad n = 0, 1, 2, \ldots, N, \tag{10}$$

in which M and N are integers. The notations u_{ij}^n and p^n are used for the finite difference approximations of $u(ih, jh, nk)$ and $p(nk)$, respectively.

The numerical methods suggested here are based on two approaches: first a numerical technique is used to approximate the solution of the two-dimensional diffusion equation, and second, a special procedure is used to evaluate $p(t)$ approximately using the temperature overspecification condition.

The finite difference formula described in this section and to be applied at interior gridpoints in the solution domain are the (3,3)

alternating direction implicit finite difference scheme or the (3,9) alternating direction implicit formula, which approximates the solution of the two-dimensional linear diffusion equation.

Using the initial condition

$$u(x,y,0) = f(x,y), \quad 0 \le x,y \le 1, \tag{11}$$

Equation (1) is solved approximately at the spatial points (x_i, y_j), commencing with initial values $u_{i,j}^0 = f(x_i, y_j)$, $i,j = 0,1,2,\ldots,M$, and boundary values

$$u_{0,j}^{n+1} = g_0(y_j, t_{n+1}), \tag{12}$$

$$u_{M,j}^{n+1} = g_1(y_j, t_{n+1}), \tag{13}$$

$$u_{i,0}^{n+1} = h_0(x_i, t_{n+1}), \tag{14}$$

$$u_{i,M}^{n+1} = h_1(x_i, t_{n+1}), \tag{15}$$

for $n = 0,1,2,\ldots,N-1$, where $h_0(x,t)$, $h_1(x,t)$, $g_0(y,t)$ and $g_1(y,t)$ are give in the boundary conditions and $p(t)$ will be found by the procedure described in Section 3.

In practical problems, (x_0, y_0) is a data point which can be always be chosen as a mesh point, *i.e.*, $x_0 = l_0 h$, $y_0 = k_0 h$, for some integers $1 \le l_0$, $k_0 \le M-1$. With this identification, the finite difference form of (7) can be written as [7].

$$u_{k_0,l_0}^n = E^n. \tag{16}$$

2.1. The (3,3) ADI Method

In the first half time interval of the (3,3) ADI procedure applied to our problem, the following formula is used:

$$\begin{aligned} &- s_x u_{i-1,j}^{n+1/2} + 2(1+s_x) u_{i,j}^{n+(1/2)} - s_x u_{i+1,j}^{n+(1/2)} \\ &\quad = s_y u_{i,j-1}^n + 2(1-s_y) u_{i,j}^n + s_y u_{i,j+1}^n \\ &\qquad + k(p^n u_{i,j}^n + \phi_{i,j}^n), \end{aligned} \tag{17}$$

with $i=1,2,\ldots,M-1$, for each $j=1,2,\ldots,M-1$, where

$$s_x = \frac{k}{(\Delta x)^2}, \tag{18}$$

and

$$s_y = \frac{k}{\Delta y^2}. \tag{19}$$

The resulting system of linear equations is strictly diagonally dominant, which guarantees that it is solvable. This system is tridiagonal and can be solved using the very fast Thomas algorithm.

This procedure is unconditionally von Neumann stable [12].

In the following the procedure using this formula will be referred to as the (3,3) method, because the computational molecule used for the x-sweep of this scheme involves three gridpoints at the new time level and three at the old level.

In the second half time interval the following formula is used with $j=1,2,\ldots,M-1$, for each $i=1,2,\ldots,M-1$,

$$\begin{aligned} &-s_y u_{i,j-1}^{n+1} + 2(1+s_y)u_{i,j}^{n+1} - s_y u_{i,j+1}^{n+1} \\ &\quad = s_x u_{i-1,j}^{n+(1/2)} + 2(1-s_x)u_{i,j}^{n+(1/2)} + s_x u_{i+1,j}^{n+(1/2)} \\ &\qquad + k(p^{n+(1/2)}u_{i,j}^{n+(1/2)} + \phi_{i,j}^{n+(1/2)}), \end{aligned} \tag{20}$$

for $i,j=1,2,\ldots,M-1$. The notation $u_{i,j}^{n+1/2}$ refers to values of $u_{i,j}$ computed at the intermediate stage, that is, at time $(t_n+k/2)$.

Values of $u_{i,j}^{n+1}$ on the boundaries $y=0$ and $y=1$, $x=0$ and $x=1$, are provided by the boundary conditions (3)–(6).

In the case $\Delta x=\Delta y=h$, we have

$$s_x = s_y = s = \frac{k}{h^2}, \tag{21}$$

and the formulae to be used in the two half-time steps of this time-split procedure are:

$$\begin{aligned} &-su_{i-1,j}^{n+(1/2)} + 2(1+s)u_{i,j}^{n+(1/2)} - su_{i+1,j}^{n+(1/2)} \\ &\quad = su_{i,j-1}^{n} + 2(1-s)u_{i,j}^{n} + su_{i,j+1}^{n} \\ &\qquad + k(p^n u_{i,j}^n + \phi_{i,j}^n), \end{aligned} \tag{22}$$

and

$$
\begin{aligned}
&-su_{i,j-1}^{n+1}+2(1+s)u_{i,j}^{n+1}-su_{i,j+1}^{n+1}\\
&\quad=su_{i-1,j}^{n+(1/2)}+2(1-s)u_{i,j}^{n+(1/2)}+su_{i+1,j}^{n+(1/2)}\\
&\qquad+k(p^{n+(1/2)}u_{i,j}^{n+(1/2)}+\phi_{i,j}^{n+(1/2)}).
\end{aligned}
\tag{23}
$$

Because each of these equations are consistent with the two dimensional diffusion equation values of $u_{0,j}^{n+1/2}$ and $u_{Mj}^{n+1/2}$, are provided by the boundary conditions (3)–(4).

In contrast to the locally one-dimensional (LOD) method, this ADI scheme is consistent with the full two-dimensional diffusion equation. Here there is no difficulty in splitting the time steps in half, because any boundary conditions specified are correct after each stage of the ADI process.

The sum of the modified equivalent partial differential equations corresponding to the formulae (17) and (20) at time $t_{n+1/2}$ is as follows [17]

$$
\begin{aligned}
&\frac{\partial u}{\partial t}-\frac{\partial^2 u}{\partial x^2}-\frac{\partial^2 u}{\partial y^2}-pu-\phi-\frac{(\Delta x)^2}{12}\frac{\partial^4 u}{\partial x^4}\\
&\quad-\frac{(\Delta y)^2}{12}\frac{\partial^4 u}{\partial y^4}+O\{(\Delta x)^4,(\Delta y)^4\}=0.
\end{aligned}
\tag{24}
$$

This equation shows that the (3,3) ADI scheme is only second-order accurate.

2.2. The (3,9) ADI Method

The x and y sweeps of the (3,9) ADI method for solving the two dimensional problem (1) are described in the following.

In the first half-time interval of the (3,9) ADI procedure, the following approximation will be used.

$$
\left.\frac{\partial u}{\partial t}\right|_{i,j}^{n}=\left.\frac{\partial^2 u}{\partial x^2}\right|_{i,j}^{n}+\left.\frac{\partial^2 u}{\partial y^2}\right|_{i,j}^{n}+(pu+\phi)|_{i,j}^{n}
\tag{25}
$$

$$u_t|_{i,j}^n = \frac{1}{6}\frac{(u_{i,j-1}^{(n+(1/2))} - u_{i,j-1}^n)}{k} + \frac{2}{3}\frac{(u_{i,j}^{(n+(1/2))} - u_{i,j}^n)}{k} + \frac{1}{6}\frac{(u_{i,j+1}^{(n+(1/2))} - u_{i,j+1}^n)}{k}, \tag{26}$$

$$u_{xx}|_{i,j}^n = \frac{1}{6}\frac{(u_{i-1,j-1}^n - 2u_{i,j-1}^n + u_{i+1,j-1}^n)}{(\Delta x)^2} + \frac{2}{3}\frac{(u_{i-1,j}^n - 2u_{i,j}^n + u_{i+1,j}^n)}{(\Delta x)^2} + \frac{1}{6}\frac{(u_{i-1,j+1}^n - 2u_{i,j+1}^n + u_{i+1,j+1}^n)}{(\Delta x)^2}, \tag{27}$$

$$u_{yy}|_{i,j}^n = \frac{1}{6}\frac{(u_{i-1,j-1}^n - 2u_{i-1,j}^n + u_{i-1,j+1}^n)}{(\Delta y)^2} + \frac{2}{3}\frac{(u_{i,j-1}^{n+(1/2)} - 2u_{i,j}^{n+(1/2)} + u_{i,j+1}^{n+(1/2)})}{(\Delta y)^2} + \frac{1}{6}\frac{(u_{i+1,j-1}^n - 2u_{i+1,j}^n + u_{i+1,j+1}^n)}{(\Delta y)^2}. \tag{28}$$

In the first half-time interval of the ADI procedure, with each $i = 1, 2, \ldots, M-1$, and for each $j = 1, 2 \ldots, M-1$ we have

$$\begin{aligned}(6s_y - 1)u_{i,j-1}^{n+(1/2)} - 4(1 + 3s_y)u_{i,j}^{n+(1/2)} + (6s_y - 1)u_{i,j+1}^{n+(1/2)} \\ = -s_x(u_{i-1,j-1}^n + u_{i-1,j+1}^n + u_{i+1,j-1}^n + u_{i+1,j+1}^n) \\ - 4s_x(u_{i-1,j}^n + u_{i+1,j}^n) \\ + 2(s_x - 1)(u_{i,j-1}^n + 4u_{i,j}^n + u_{i,j+1}^n) + k(p^n u_{i,j}^n + \phi_{i,j}^n),\end{aligned} \tag{29}$$

where the notation $u_{i,j}^{n+(1/2)}$ refers to values of $u_{i,j}$ computed at the intermediate stage.

As in the (3,3) ADI method, the resulting system of linear algebraic equations is strictly diagonally dominant, which ensures it is solvable. This system can be solved by using the very fast Thomas algorithm.

In the second half time interval the following approximations are used

$$\left.\frac{\partial u}{\partial t}\right|_{i,j}^{n+(1/2)} = \left.\frac{\partial^2 u}{\partial x^2}\right|_{i,j}^{n+(1/2)} + \left.\frac{\partial^2 u}{\partial y^2}\right|_{i,j}^{n+(1/2)} + (pu + \phi)|_{i,j}^{n+(1/2)}, \tag{30}$$

$$u_t|_{i,j}^{n+(1/2)} = \frac{1}{6}\frac{(u_{i-1,j}^{n+1} - u_{i-1,j}^{n+(1/2)})}{k} + \frac{2}{3}\frac{(u_{i,j}^{n+1} - u_{i,j}^{n+(1/2)})}{k} + \frac{1}{6}\frac{(u_{i+1,j}^{n+1} - u_{i+1,j}^{n+(1/2)})}{k}, \tag{31}$$

$$u_{xx}|_{i,j}^{n+(1/2)} = \frac{1}{6}\frac{(u_{i-1,j-1}^{n+(1/2)} - 2u_{i,j-1}^{n+(1/2)} + u_{i+1,j-1}^{n+(1/2)})}{(\Delta x)^2} + \frac{2}{3}\frac{(u_{i-1,j}^{n+1} - 2u_{i,j}^{n+1} + u_{i+1,j}^{n+1})}{(\Delta x)^2} + \frac{1}{6}\frac{(u_{i-1,j+1}^{n+(1/2)} - 2u_{i,j+1}^{n+(1/2)} + u_{i+1,j+1}^{n+(1/2)})}{(\Delta x)^2}, \tag{32}$$

$$u_{yy}|_{i,j}^{n+(1/2)} = \frac{1}{6}\frac{(u_{i-1,j-1}^{n+(1/2)} - 2u_{i-1,j}^{n+(1/2)} + u_{i-1,j+1}^{n+(1/2)})}{(\Delta y)^2} + \frac{2}{3}\frac{(u_{i,j-1}^{n+(1/2)} - 2u_{i,j}^{n+(1/2)} + u_{i,j+1}^{n+(1/2)})}{(\Delta y)^2} + \frac{1}{6}\frac{(u_{i+1,j-1}^{n+(1/2)} - 2u_{i+1,j}^{n+(1/2)} + u_{i+1,j+1}^{n+(1/2)})}{(\Delta y)^2}. \tag{33}$$

In the second half time interval the following formula is used with $j=1,2,\ldots,M-1$, for each $i=1,2,\ldots,M-1$,

$$\begin{aligned}(6s_x-1)u_{i-1,j}^{n+1} - 4(1+3s_x)u_{i,j}^{n+1} + (6s_x-1)u_{i+1,j}^{n+1} &= -s_y(u_{i-1,j-1}^{n+(1/2)} + u_{i-1,j+1}^{n+(1/2)} + u_{i+1,j-1}^{n+(1/2)} + u_{i+1,j+1}^{n+(1/2)}) \\ &\quad - 4s_y(u_{i,j-1}^{n+(1/2)} + u_{i,j+1}^{n+(1/2)}) \\ &\quad + 2(s_y-1)(u_{i-1,j}^{n+(1/2)} + 4u_{i,j}^{n+(1/2)} + u_{i+1,j}^{n+(1/2)}) \\ &\quad + k(p^{n+(1/2)}u_{i,j}^{n+(1/2)} + \phi_{i,j}^{n+(1/2)}).\end{aligned} \tag{34}$$

In the case $\Delta x=\Delta y=h$, we have $s_x=s_y=s$, and the formulae to be used in the two half-time steps of a time-split procedure are:

$$\begin{aligned}(6s-1)u_{i,j-1}^{n+(1/2)} - 4(1+3s)u_{i,j}^{n+(1/2)} + (6s-1)u_{i,j+1}^{n+(1/2)} &= -s(u_{i-1,j-1}^{n} + u_{i-1,j+1}^{n} + u_{i+1,j-1}^{n} + u_{i+1,j+1}^{n}) \\ &\quad - 4s(u_{i-1,j}^{n} + u_{i+1,j}^{n}) + 2(s-1)(u_{i,j-1}^{n} + 4u_{i,j}^{n} + u_{i,j+1}^{n}) \\ &\quad + k(p^n u_{i,j}^n + \phi_{i,j}^n),\end{aligned} \tag{35}$$

and

$$(6s-1)u_{i-1,j}^{n+1} - 4(1+3s)u_{i,j}^{n+1} + (6s-1)u_{i+1,j}^{n+1}$$
$$= -s(u_{i-1,j-1}^{n+(1/2)} + u_{i-1,j+1}^{n+(1/2)} + u_{i+1,j-1}^{n+(1/2)} + u_{i+1,j+1}^{n+(1/2)})$$
$$- 4s(u_{i,j-1}^{n+(1/2)} + u_{i,j+1}^{n+(1/2)}) + 2(s-1)(u_{i-1,j}^{n+(1/2)} + 4u_{i,j}^{n+(1/2)} + u_{i+1,j}^{n+(1/2)})$$
$$+ k(p^{n+(1/2)}u_{i,j}^{n+(1/2)} + \phi_{i,j}^{n+(1/2)}). \tag{36}$$

Values of $u_{i,j}^{n+1}$ on the boundaries $x=0,1$ and $y=0,1$ are provided by the boundary conditions (3)–(6).

This procedure is unconditionally von Neumann stable.

In the following the procedure using this formula will be referred to as the (3,9) technique, because the computational molecule used for the x-sweep of this method involves three gridpoints at the new time level and nine at the old level.

The modified equivalent equation of the double-sweep procedure of the (3,9) ADI formulae is as follows:

$$\frac{\partial u}{\partial t} - \frac{\partial^2 u}{\partial x^2} - \frac{\partial^2 u}{\partial y^2} - pu - \phi - \frac{(\Delta x)^4}{180}(120(s_x)^2 - 30s_x + 1)\frac{\partial^6 u}{\partial x^6}$$
$$- \frac{(\Delta y)^4}{180}(120(s_y)^2 - 30s_y + 1)\frac{\partial^6 u}{\partial y^6} + O\{(\Delta x)^6, (\Delta y)^6\} = 0, \tag{37}$$

which verifies its fourth-order convergence rate.

The choice of $s_x = s_y = (15 + \sqrt{105})/120$ or $s_x = s_y = (15 - \sqrt{105})/120$ makes the (3,9) ADI formula sixth-order convergent for the two-dimensional diffusion equation.

3. EVALUATING THE CONTROL PARAMETER P(t)

If $u(x, y, t)$ and $p(t)$ form a solution for (1)–(7), then

$$E'(t) = u_{xx}(x_0, y_0, t) + u_{yy}(x_0, y_0, t) + p(t)E(t) + \phi(x_0, y_0, t), \tag{38}$$

which gives:

$$p(t) = \frac{E'(t) - u_{xx}(x_0, y_0, t) - u_{yy}(x_0, y_0, t) - \phi(x_0, y_0, t)}{E(t)}. \tag{39}$$

Using (16) and (7), the finite difference form of (39) is

$$p^n = \frac{(E')^n - (1/h^2)(u^n_{k_0-1,l_0} - 2u^n_{k_0,l_0} + u^n_{k_0+1,l_0}) - (1/h^2)(u^n_{k_0,l_0-1} - 2u^n_{k_0,l_0} + u^n_{k_0,l_0+1}) - \phi^n_{k_0,l_0}}{E^n}. \tag{40}$$

Combining this with the compatibility conditions yields [16]

$$p^0 = \frac{E'(0) - f_{xx}(x_0, y_0) - f_{yy}(x_0, y_0) - \phi(x_0, y_0, 0)}{f(x_0, y_0)}. \tag{41}$$

This p^0, together with the values of $u(x, y, t)$ at $n = 0$ level given by the initial condition, provide a starting point for our computation.

Note that the presence of the temperature overspecification term can greatly complicate the application of standard numerical methods. The accuracy of the technique used to evaluate the control parameter must be compatible with that of the discretization of the diffusion equation [10]. It means that in the case of using a fourth order finite difference formula such as, the (3,9) ADI technique, a higher order of accuracy than that used for the (3,3) ADI scheme, is required. So in this case, the following fourth order scheme is employed for computing $p(t)$ approximately:

$$\begin{aligned} p^n = \frac{1}{E^n}\Bigg((E')^n &- \frac{1}{12h^2}(-u^n_{k_0-2,l_0} + 16u^n_{k_0-1,l_0} \\ &\quad - 30u^n_{k_0,l_0} + 16u^n_{k_0+1,l_0} - u^n_{k_0+2,l_0}) \\ &- \frac{1}{12h^2}(-u^n_{k_0,l_0-2} + 16u^n_{k_0,l_0-1} - 30u^n_{k_0,l_0} \\ &\quad + 16u^n_{k_0,l_0+1} - u^n_{k_0,l_0+2}) - \phi^n_{k_0,l_0}\Bigg). \end{aligned} \tag{42}$$

Since for practical computation, the time step size is small, it is reasonable to assume that p^{n+1} is not far from p^n. Thus, a good choice of the initial guess for p^{n+1}, denoted by $p^{(n+1)(0)}$, can be made as $p^{(n+1)(0)} = p^n$, for $n = 0, 1, \ldots, N$.

Substituting p^n and $p^{(n+1)(0)}$ into (22) for the (3,3) ADI finite difference scheme or into (35) for the (3,9) ADI technique, makes the related linear systems ready for solution.

Solving these linear systems, we obtain $u^{n+1(0)}_{i,j}$, $i, j = 1, 2, \ldots, M-1$ corresponding to $p^{n+1(0)}$. We use $p^{n+1(l)}$ to denote the lth guess for $p(t)$

at level $n+1$ and $u_{i,j}^{n+1(l)}$ to denote the corresponding values obtained by using $p^{(n+1)(l)}$, $n=0,1,\ldots,N-1$, $l=0,1,\ldots$.

For the corrections, we use (40) to generate $p^{n+1(l+1)}$ in the following way:

$$p^{n+1(l+1)} = \frac{1}{E^{n+1}}\Big((E^l)^{n+1} - (1/h^2)\Big(u_{k_0-1,l_0}^{n+1(l)} - 2u_{k_0,l_0}^{n+1(l)} + u_{k_0+1,l_0}^{n+1(l)}\Big) - (1/h^2)\Big(u_{k_0,l_0-1}^{n+1(l)} - 2u_{k_0,l_0}^{n+1(l)} + u_{k_0,l_0+1}^{n+1(l)}\Big) - \phi_{k_0,l_0}^{n+1}\Big), \quad (43)$$

for $l=0,1,\ldots,$.

Note with the (3,9) ADI finite difference scheme, which is a fourth-order technique, the following formula will be used to generate $p^{n+1(l+1)}$.

$$p^{n+1(l+1)} = \frac{1}{E^{n+1}}\Big((E^l)^{n+1} - \frac{1}{12h^2}\Big(-u_{k_0-2,l_0}^{n+1(l)} + 16u_{k_0-1,l_0}^{n+1(l)} - 30u_{k_0,l_0}^{n+1(l)} + 16u_{k_0+1,l_0}^{n+1(l)} - u_{k_0+2,l_0}^{n+1(l)}\Big) - \frac{1}{12h^2}\Big(-u_{k_0,l_0-2}^{n+1(l)} + 16u_{k_0,l_0-1}^{n+1(l)} - 30u_{k_0,l_0}^{n+1(l)} + 16u_{k_0,l_0+1}^{n+1(l)} - u_{k_0,l_0+2}^{n+1(l)}\Big) - \phi_{k_0,l_0}^{n+1}\Big), \quad (44)$$

for $l=0,1,\ldots$.

We will adjust $p^{n+1(l)}$ repeatedly until it converges, *i.e.*, satisfies a prescribed tolerance. Then we accept the corresponding values $u_{i,j}^{n+1(l)}$, $i,j=1,2,\ldots,M-1$ and $p^{n+1(l)}$ as $u_{i,j}^{n+1}$, $i,j=1,2,\ldots,M-1$ and p^{n+1} respectively. This strategy completes [7] the advancing from level n to level $n+1$.

4. NUMERICAL TEST

A problem for which exact solution is known is now used to test the methods described for solving the inverse problem with temperature overspecification. These are applied to solve (1)–(7) with $p(t)$ and u unknown.

Consider (1)–(7) with

$$\phi(x,y,t) = \left(\frac{5\pi^2}{16} - 5t\right)\exp(t)\sin\left(\frac{\pi}{4}(x+2y)\right), \tag{45}$$

$$g_0(0,y,t) = \exp(t)\sin\left(\frac{\pi y}{2}\right), \tag{46}$$

$$g_1(1,y,t) = \exp(t)\sin\left(\frac{\pi}{4}(1+2y)\right), \tag{47}$$

$$h_0(x,0,t) = \exp(t)\sin\left(\frac{\pi x}{4}\right), \tag{48}$$

$$h_1(x,1,t) = \exp(t)\sin\left(\frac{\pi}{4}(x+2)\right), \tag{49}$$

$$E(t) = \exp(t)\sin(0.2\pi), \tag{50}$$

$$f(x,y) = \sin\frac{\pi}{4}(x+2y), \tag{51}$$

for which the exact solution is

$$u(x,y,t) = \exp(t)\sin\frac{\pi}{4}(x+2y), \tag{52}$$

and

$$p(t) = 1 + 5t. \tag{53}$$

The results obtained for $u_{i,j}^N$ at $T=1.0$, computed for $h=0.02$, $s=1/4$ and $(x_0,y_0)=(0.4,\ 0.2)$, using both the (3,3) ADI finite difference method and the (3,9) ADI method are listed in Table I. These results reflect the fourth order convergence of the (3,9) ADI scheme.

The interesting feature of these results is that the error obtained when using the (3,9) ADI formula is about one thousand times smaller than those obtained when using the (3,3) ADI scheme. The results of the (3,3) ADI technique and the implicit scheme of [16] are about the same.

TABLE I Results for u with $h = 1/50$, $s = 1/4$ and $T = 1.0$

x	y	*Exact u*	(3,3) *ADI Error*	(3,9) *ADI Error*	*Implicit Method Error*
0.1	0.1	0.640478	3.5×10^{-3}	5.1×10^{-6}	7.7×10^{-3}
0.2	0.2	1.234072	3.6×10^{-3}	5.2×10^{-6}	7.4×10^{-3}
0.3	0.3	1.765382	3.6×10^{-3}	5.4×10^{-6}	8.1×10^{-3}
0.4	0.4	2.199132	3.7×10^{-3}	5.3×10^{-6}	9.5×10^{-3}
0.5	0.5	2.511363	3.8×10^{-3}	5.3×10^{-6}	9.6×10^{-3}
0.6	0.6	2.684813	3.9×10^{-3}	5.5×10^{-6}	8.7×10^{-3}
0.7	0.7	2.709900	4.0×10^{-3}	5.6×10^{-6}	8.5×10^{-3}
0.8	0.8	5.123874	4.0×10^{-3}	5.4×10^{-6}	7.9×10^{-3}
0.9	0.9	5.764316	3.8×10^{-3}	5.2×10^{-6}	7.3×10^{-3}

The results obtained for $p(t)$ with $h = 0.02$, $s = 1/4$, using both the (3,3) ADI finite difference scheme and the (3,9) ADI formula with $p(t)$ defined as in (53) and it was considered to be unknown and found by (43) for the (3,3) ADI scheme, and by (44) for the (3,9) ADI method are shown in Table II. This table shows that the results obtained using the (3,9) ADI method are much more accurate than using the (3,3) ADI finite difference method.

Note that the results obtained when using the (3,9) ADI scheme are about one thousand times more accurate than those obtained when using the implicit scheme of [16].

The CPU time for the (3,3) ADI finite difference method was 105.9 s and for the (3,9) ADI scheme was 108.6 s, while the time for the implicit method was 876.6 s.

The time needed using these ADI methods was about 8 times shorter than using the implicit method of [16].

TABLE II Results for p with $h = 1/50$, $s = 1/4$

t	*Exact p*	(3,3) *ADI Error*	(3,9) *ADI Error*	*Implicit Method Error*
0.1	1.500000	9.5×10^{-3}	6.2×10^{-5}	7.9×10^{-2}
0.2	2.000000	9.6×10^{-3}	6.3×10^{-5}	6.0×10^{-2}
0.3	2.500000	9.7×10^{-3}	6.5×10^{-5}	8.0×10^{-2}
0.4	3.000000	9.7×10^{-3}	6.4×10^{-5}	9.0×10^{-2}
0.5	3.500000	9.9×10^{-3}	6.5×10^{-5}	9.5×10^{-2}
0.6	4.000000	9.9×10^{-3}	6.6×10^{-5}	8.8×10^{-2}
0.7	4.500000	9.8×10^{-3}	6.3×10^{-5}	8.1×10^{-2}
0.8	5.000000	9.6×10^{-3}	6.0×10^{-5}	7.2×10^{-2}
0.9	5.500000	9.3×10^{-3}	5.9×10^{-5}	6.9×10^{-2}
1.0	6.000000	9.1×10^{-3}	5.8×10^{-5}	5.4×10^{-2}

5. CONCLUSIONS

In this paper the (3,3) ADI finite difference method and the (3,9) ADI scheme were applied to a two-dimensional inverse problem with temperature overspecification. The proposed numerical schemes solved this model quite satisfactory. Using the (3,3) ADI finite difference scheme or the (3,9) ADI technique for the two-dimensional linear diffusion problem with appropriate treatment on the control parameter describe our model well. These procedures are very simple to implement and economical to use. They are very efficient and they need less CPU time than the implicit methods. A comparison with results from the implicit scheme of [16] for the model problem used clearly demonstrates that the ADI techniques are computationally superior. A common feature of the explicit finite difference methods is the restriction of the size of the time step due to stability requirements. This restriction necessitates extremely small values for k. This limitation is removed when the implicit finite difference schemes are used. However, a disadvantage of these techniques is the extensive amount of CPU times utilized in determining the numerical solution compared to the explicit methods for the same selection of values s and h. So these schemes are impractical for higher dimensional problems. So the need to develop the ADI techniques is clear. These ADI finite difference schemes are very easy to implement for similar 3-dimensional inverse problems, but it may be more difficult when dealing with the implicit schemes or the explicit methods. Another extension of these results might include two-dimensional parabolic inverse problems with energy overspecification.

References

[1] Cannon, J. R. and Yin, H. M. (1989). On a class of non-classical parabolic problems, *J. Differential Equations*, **79**, 266–288.

[2] Cannon, J. R. and Lin, Y. (1988). Determination of parameter $p(t)$ in Holder classes for some semilinear parabolic equations, *Inverse Problems*, **4**, 595–606.

[3] Cannon, J. R. and Lin, Y. (1990). An inverse problem of finding a parameter in a semi-linear heat equation, *J. Math. Anal. Appl.*, **145**(2), 470–484.

[4] Cannon, J. R. and Yin, H. M. (1990). Numerical solution of some parabolic inverse problems, *Numerical Methods for Partial Differential Equations*, **2**, 177–191.

[5] Cannon, J. R. and Lin, Y. (1988). Determination of parameter $p(t)$ in some quasi-linear parabolic differential equations, *Inverse Problems*, **4**, 35–45.

[6] Cannon, J. R., Lin, Y. and Wang, S., Determination of a control function in a parabolic partial differential equations, *Research report*, 89-10, Department of Mathematics and Statistics, Mcgill University.
[7] Cannon, J. R., Lin, Y. and Wang, S. (1992). Determination of source parameter in parabolic equations, *Meccanica*, **27**, 85–94.
[8] Day, W. A. (1982). Extension of a property of the heat equation to linear thermoelasticity and other theories, *Quart. Appl. Math.*, **40**, 319–330.
[9] Dehghan, M. (1999). Fully implicit finite difference methods for two-dimensional diffusion with a non-local boundary condition, *Journal of computational and Applied Mathematics*, **106**, 255–269.
[10] Fairweather, G. and Saylor, R. D. (1991). The reformulation and numerical solution of certain nonclassical initial-boundary value problems, *SIAM, J. Sci. Stat. Comput.*, **12**(1), 127–144.
[11] Lapidus, L. and Pinder, G. F. (1982). *Numerical solution of partial differential equations in science and engineering*. Wiley, New York.
[12] Mitchell, A. R. and Griffiths, D. F. (1980). *The finite difference methods in partial differential equations*, T.Wiley.
[13] Prilepko, A. I. and Orlovskii, D. G. (1985). Determination of the evolution parameter of an equation and inverse problems of mathematical physics, I, *J. Differential Equations*, **21**, 119–129 and part II, **21**, 649–701.
[14] Prilepko, A. I. and Soloev, V. V. (1987). Solvability of the inverse boundary value problem of finding a coefficient of a lower order term in a parabolic equation, *Differential Equations*, **23**(1), 136–143.
[15] Rundell, W. (1980). Determination of an unknown non-homogenous term in a linear partial differential equation from overspecified boundary data, *Appl. Anal.*, **10**, 231–242.
[16] Wang, S. (1992). Numerical solutions of two inverse problems for identifying control parameters in 2-dimensional parabolic partial differential equations. *Modern developements in numerical simulation of flow and heat transfer*, HTP-Vol. 194, pp. 11–16.
[17] Warming, R. F. and Hyett, B. J. (1974). The modified equation approach to the stability and accuracy analysis of finite-difference methods, *Journal of Computational Physics*, **14**(2), 159–179.