Research Article

Fourth-Order Deferred Correction Scheme for Solving Heat Conduction Problem

D. Yambangwai¹ and N. P. Moshkin²

¹ Department of Mathematics, School of Science, University of Phayao, Phayao 56000, Thailand

² School of Mathematics, Institute of Science, Suranaree University of Technology, Nakhon Ratchasima 30000, Thailand

Correspondence should be addressed to D. Yambangwai; damrong.sut@gmail.com

Received 3 December 2012; Accepted 27 February 2013

Academic Editor: Safa Bozkurt Coskun

Copyright © 2013 D. Yambangwai and N. P. Moshkin. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A deferred correction method is utilized to increase the order of spatial accuracy of the Crank-Nicolson scheme for the numerical solution of the one-dimensional heat equation. The fourth-order methods proposed are the easier development and can be solved by using Thomas algorithms. The stability analysis and numerical experiments have been limited to one-dimensional heat-conducting problems with Dirichlet boundary conditions and initial data.

1. Introduction

The desired properties of finite difference schemes are stability, accuracy, and efficiency. These requirements are in conflict with each other. In many applications a high-order accuracy is required in the spatial discretization. To reach better stability, implicit approximation is desired. For a high-order method of traditional type (not a high-order compact (HOC)), the stencil becomes wider with increasing order of accuracy. For a standard centered discretization of order p , the stencil is $p + 1$ points wide. This inflicts problems at the fictional boundaries, and using an implicit method results in the solution of an algebraic system of equations with large bandwidth. In light of conflict requirements of stability, accuracy, and computational efficiency, it is desired to develop schemes that have a wide range of stability and high order of accuracy and lead to the solution of a system of linear equations with a tri-diagonal matrix, that is, the system of linear equations arising from a standard second-order discretization of heat equation.

The development of high-order compact (HOC) schemes [1–18] is one approach to overcome the antagonism between stability, accuracy, and computational cost. However, the HOC becomes complicated when applied to multidimensional problems or to non-Cartesian coordinate cases.

Another way of preserving a compact stencil at higher time level and reaching high-order spatial accuracy is the deferred correction approach [11]. A classical deferred correction procedure is developed in [19, 20].

In this paper we use the deferred correction technique to obtain fourth-order accurate schemes in space for the one-dimensional heat-conducting problem with Dirichlet boundary conditions. The linear system that needs to be solved at each time step is similar to the standard Crank-Nicolson method of second order which is solved by using Thomas algorithms. The fourth-order deferred (FOD) correction schemes are compared with the fourth-order semi-implicit (FOS) schemes and fourth-order compact (FOC) schemes for the Dirichlet boundary value problems.

A set of schemes are constructed for the one-dimensional heat-conducting problem with Dirichlet boundary conditions and initial data:

$$u_t = \beta u_{xx} + f(x, t), \quad 0 < x < l, \quad t > 0, \quad (1)$$

$$u(x, 0) = u_0(x), \quad 0 < x < l, \quad (2)$$

$$\text{Dirichlet BC : } u(0, t) = \gamma_1(t), \quad u(l, t) = \gamma_2(t), \quad t > 0, \quad (3)$$

where the diffusion coefficient β is positive, $u(x, t)$ represents the temperature at point (x, t) , and $f(x, t)$, $\gamma_1(t)$, $\gamma_2(t)$ are sufficiently smooth functions.

The rest of this paper is organized as follows. Section 2 presents an FOD scheme which we use to compare performance of proposed scheme with FOS and FOC schemes. Section 3 provides examples of comparisons. Although FOD schemes have a higher computational cost than FOS and FOC schemes, it is evident from these examples that the FOD schemes have the advantage of accuracy in the uniform norm, robustness, and the ability to be extended easily to the multidimensional case. We conclude the paper in Section 4.

2. The Fourth-Order Schemes

Let Δt denote the temporal mesh size. For simplicity, we consider a uniform mesh consisting of N points: $x_1, x_2, \dots, x_N$ where $x_i = (i - 1)\Delta x$ and the mesh size is $\Delta x = l/(N - 1)$. Below we use the notations u_i^m and $(u_{xx})_i^m$ to represent the numerical approximations of $u(x_i, t^m)$ and $u_{xx}(x_i, t^m)$ where $t^m = m\Delta t$ and $u^{(p)}$ is the value of the p th derivative of the given function u .

2.1. Fourth-Order Semi-Implicit Scheme. The application to the well-known Crank-Nikolson scheme to (1) results in the following expression:

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \frac{\beta}{2} \left[(u_{xx})_i^{n+1} + (u_{xx})_i^n \right] + f_i^{n+1/2}, \quad (4)$$

where $f_i^{n+1/2} = (f_i^{n+1} + f_i^n)/2$, $i = 2, \dots, N - 1$. The Dirichlet boundary conditions

$$\begin{aligned} u(0, m\Delta t) &= \gamma_1(t^m) = u_1^m, \\ u(l, m\Delta t) &= \gamma_2(t^m) = u_N^m \end{aligned} \quad (5)$$

are used to derive the following fourth-order approximation of second derivative terms:

$$\begin{aligned} (u_{xx})_2^m &= \frac{1}{12\Delta x^2} (a_1 u_1^m + a_2 u_2^m + a_3 u_3^m + a_4 u_4^m \\ &\quad + a_5 u_5^m + a_6 u_6^m) \\ &= \frac{a_1}{12\Delta x^2} \gamma_1(t^m) + \frac{1}{12\Delta x^2} \\ &\quad \times (a_2 u_2^m + a_3 u_3^m + a_4 u_4^m + a_5 u_5^m + a_6 u_6^m), \end{aligned}$$

$$\begin{aligned} (u_{xx})_i^m &= \frac{1}{12\Delta x^2} \\ &\quad \times (-u_{i-2}^m + 16u_{i-1}^m - 30u_i^m + 16u_{i+1}^m - u_{i+2}^m), \\ &\quad i = 3, \dots, N - 2, \\ (u_{xx})_{N-1}^m &= \frac{1}{12\Delta x^2} (a_1 u_N^m + a_2 u_{N-1}^m \\ &\quad + a_3 u_{N-2}^m + a_4 u_{N-3}^m \\ &\quad + a_5 u_{N-4}^m + a_6 u_{N-5}^m) \\ &= \frac{a_1}{12\Delta x^2} \gamma_2(t^m) + \frac{1}{12\Delta x^2} \\ &\quad \times (a_2 u_{N-1}^m + a_3 u_{N-2}^m \\ &\quad + a_4 u_{N-3}^m + a_5 u_{N-4}^m + a_6 u_{N-5}^m), \end{aligned} \quad (6)$$

where the coefficients can be found by matching the Taylor series expansion of left-hand-side terms up to order $O(\Delta x^4)u^{(6)}$ which gives the following values of coefficients:

$$\begin{aligned} a_1 &= 10, & a_2 &= -15, & a_3 &= -4, \\ a_4 &= 14, & a_5 &= -6, & a_6 &= 1. \end{aligned} \quad (7)$$

Schemes (6) can be combined and expressed in the following matrix form:

$$\mathbf{u}_{xx}^m = \frac{1}{\Delta x^2} \Lambda_h \mathbf{u}^m + \gamma(t^m), \quad (8)$$

where Λ_h is the corresponding triangular and sparse $(N - 2) \times (N - 2)$ matrix,

$$\begin{aligned} \mathbf{u}_{xx}^m &= ((u_{xx})_2^m, (u_{xx})_3^m, \dots, (u_{xx})_{N-1}^m)^T, \\ \mathbf{u}^m &= (u_2^m, u_3^m, \dots, u_{N-1}^m)^T, \\ \gamma(t^m) &= (\gamma_1(t^m), 0, \dots, 0, \gamma_2(t^m))^T. \end{aligned} \quad (9)$$

Substituting (6) into (4) gives us the following matrix form:

$$\begin{aligned} (E - \alpha \Lambda_h) \mathbf{u}^{n+1} &= (E + \alpha \Lambda_h) \mathbf{u}^n \\ &\quad + \Delta t [\gamma(t^{n+1}) + \gamma(t^n)] \\ &\quad + \Delta t \mathbf{f}^{n+1/2}, \end{aligned} \quad (10)$$

where $\alpha = \beta \Delta t / (2\Delta x^2)$, $\mathbf{f}^{n+1/2} = (f_2^{n+1/2}, f_3^{n+1/2}, \dots, f_{N-1}^{n+1/2})^T$, and E denote the $(N - 2) \times (N - 2)$ identity matrix. The scheme (10) is FOSs for the heat-conducting problem with Dirichlet boundary condition. The order of approximation is $O(\Delta t^2, \Delta x^4)$ in the uniform norm. The triangular and sparse $(N - 2) \times (N - 2)$ coefficient matrix in FOSs are time independent; hence, we have to store the inverse of the coefficient matrix $E - \alpha \Lambda_h$ before the time marching in the implementation for computational efficiency.

2.2. Fourth-Order Deferred Correction Schemes. A set of fourth-order deferred correction schemes is based on the well-known Crank-Nikolson type of scheme in the following form:

$$\frac{u_i^{n+1,s+1} - u_i^n}{\Delta t} = \frac{\beta}{2} \left[(u_{xx})_i^{n+1,s+1} + (u_{xx})_i^n \right] + f_i^{n+1/2}, \quad (11)$$

where $f_i^{n+1/2} = (f_i^{n+1} + f_i^n)/2$ and the second superscript “ s ” denotes the number of iterations $s = 0, \dots, \hat{S}$ and $i = 2, \dots, N-1$.

The deferred correction technique [11] is utilized to approximate the second-order derivatives at higher time levels $(u_{xx})_i^{n+1,s+1}$, $i = 2, \dots, N-1$ by the iterative method

$$(u_{xx})_i^{n+1,s+1} = (u_{xx})_i^{n+1,s} + \left[(u_{xx}^h)_i^{n+1,s} - (u_{xx}^l)_i^{n+1,s} \right], \quad (12)$$

where $(u_{xx}^h)_i^{n+1,s}$, $i = 2, \dots, N-1$, $s = 0, \dots, \hat{S}$, is high-order approximation on wide stencil and $(u_{xx}^l)_i^{n+1,k}$, $k = s, s+1$, $i = 2, \dots, N-1$, is the lower-order approximation on compact stencil (usually three-point stencil). The expression in the square brackets of (12) is evaluated explicitly using the values known from the previous iteration. When $s = 0$ we use the solution from the time level n (so $u^{n+1,0} = u^n$ and $(u_{xx})_i^{n+1,0} = (u_{xx})_i^n$). Once the iterations converge, the lower-order approximation terms drop out and the approximation of $(u_{xx})_i^{n+1,s+1}$ obtained has the same order of approximation as $(u_{xx}^h)_i^{n+1,\hat{S}}$. There are no difficulties to construct high-order approximation for interior points.

To preserve a compact three using wide stencil in the finite difference scheme at higher time level $(n+1, s+1)$, we use the central second-order finite difference approximation to approximate the lower-order term in (12):

$$(u_{xx}^l)_i^{n+1,k} = \frac{1}{\Delta x^2} \Lambda_l u_i^{n+1,k}, \quad k = s, s+1, \quad i = 3, \dots, N-2, \quad (13)$$

$$\Lambda_l u_i^{n+1,k} = u_{i-1}^{n+1,k} - 2u_i^{n+1,k} + u_{i+1}^{n+1,k}.$$

For the high-order approximation term in (12), we use a symmetric five-point wide stencil for the inner points to reach the fourth order of approximation:

$$(u_{xx}^h)_i^{n+1,s} = \frac{1}{\Delta x^2} \Lambda_h u_i^{n+1,s}, \quad i = 3, \dots, N-2, \quad (14)$$

$$\Lambda_h u_i^{n+1,s} = \frac{1}{12} \left(-u_{i-2}^{n+1,s} + 16u_{i-1}^{n+1,s} - 30u_i^{n+1,s} + 16u_{i+1}^{n+1,s} - u_{i+2}^{n+1,s} \right).$$

Case $s = 0$ in (13) gives the fourth order of approximation to approximate the second-order derivatives at the time level n .

2.2.1. Stability Analysis. To study the stability of scheme (11)–(14), we use the Von-Neumann stability analysis. For simplicity, we assume that $f_i^{n+1/2} \equiv 0$ in (11) and u is periodic in x .

Let us recast scheme (11) in the following form:

$$(E + \alpha \Lambda_l) u_i^{n+1,s+1} = \alpha (\Lambda_l - \Lambda_h) u_i^{n+1,s} + (E - \alpha \Lambda_l) u_i^n, \quad (15)$$

where $\alpha = \beta \Delta t / (2 \Delta x^2)$. If we define the following operators $A = E + \alpha \Lambda_l$, $B = E - \alpha \Lambda_h$, and $C = E + \alpha \Lambda_h$, where E is the identity operator, then (15) can be rewritten as follows:

$$A u_i^{n+1,s+1} = (A - C) u_i^{n+1,s} + B u_i^n. \quad (16)$$

Assuming that the operators commute, $(A - C)A = A(A - C)$ (e.g., in the case of uniform grid), it is easy to demonstrate that if $u_i^{n+1,\hat{S}+1} = u_i^{n+1}$ and $u_i^{n+1,0} = u_i^n$ we get

$$A^{\hat{S}+1} u_i^{n+1} = \left(\sum_{k=0}^{\hat{S}} A^{\hat{S}-k} (A - C)^k \right) B u_i^n + (A - C)^{\hat{S}+1} u_i^n. \quad (17)$$

Let $u_i^n = \xi^n e^{I \Theta i}$, $I = \sqrt{-1}$, be the solution of (11)–(14), where $\Theta = 2\pi \Delta x / l$ is the phase angle with wavelength l . From (17), we can derive an equation for the amplification factor in the form

$$|\xi| = |\varphi(\Theta, \hat{S}, \alpha)|, \quad (18)$$

where $\hat{S}$ is the number of iterations, and

$$|\varphi(\Theta, \hat{S}, \alpha)| = \frac{\left| \left[\left(\sum_{k=0}^{\hat{S}} A^{\hat{S}-k} (A - C)^k \right) B + (A - C)^{\hat{S}+1} \right] e^{I \Theta i} \right|}{|A^{\hat{S}+1} e^{I \Theta i}|}. \quad (19)$$

For stability of the method it is necessary that the absolute values of the amplification factor are less than one; that is,

$$|\xi| < 1. \quad (20)$$

Calculations are tedious and almost impossible to do by hand without mistake. We have therefore automate all calculations in a computer algebra environment based on REDUCE to obtain an explicit form of $|\varphi(\Theta, \hat{S}, \alpha)|$. Figure 1 shows the values of $|\xi|^2$ in the polar coordinate system $(|\xi|^2, \Theta)$ for $\hat{S} = 1, 3$, and 5 . If only one iteration is executed in (11), $\hat{S} = 1$, inequality (20) holds if $\alpha < 1.5$, as can be seen from Figure 1(a)). If 3 iterations are done in (11) (Figure 1(b)), $\hat{S} = 3$, the amplification factor remains bounded by one at least for $\alpha \leq 10$. In case of $\hat{S} = 5$, the stability criteria hold up to $\alpha = 30$ as can be seen from Figure 1(c)). It can be seen that increasing the number of internal iterations results in increasing the range of α needed for stability. This tendency allows to assume that as $\hat{S} \rightarrow \infty$, our method becomes the unconditionally stable Crank-Nikolson method for the heat equation.

FIGURE 1: Variation of amplification factor with Θ . (a) $\hat{S} = 1$, dashed line $\alpha = 2.0$, solid line $\alpha = 1.5$, dash-dotted line $\alpha = 1.0$, (b) $\hat{S} = 3$, dashed line $\alpha = 10.0$, solid line $\alpha = 5.0$, dash-dotted line $\alpha = 1.0$, and (c) $\hat{S} = 5$, dashed line $\alpha = 30.0$, solid line $\alpha = 15$, dash-dotted line $\alpha = 5.0$, dotted line $\alpha = 1.0$.

2.2.2. Fourth-Order Deferred Correction Scheme. Let us first consider the one-dimensional heat conduction problem with initial data and Dirichlet boundary conditions (1)–(3):

$$u_1^{n+1,k} = \gamma_1(t^{n+1}), \quad u_N^{n+1,k} = \gamma_2(t^{n+1}). \quad (21)$$

The finite difference approximations at x_2 and x_{N-1} , which are the points next to the left and right boundaries, are straightforward:

$$\begin{aligned} (u_{xx}^l)_2^{n+1,k} &= \frac{1}{\Delta x^2} (\gamma_1(t^{n+1}) - 2u_2^{n+1,k} + u_3^{n+1,k}), \\ k &= s, s+1, \\ (u_{xx}^h)_2^{n+1,s} &= \frac{1}{12\Delta x^2} (10\gamma_1(t^{n+1}) - 15u_2^{n+1,s} - 4u_3^{n+1,s} \\ &\quad + 14u_4^{n+1,s} - 6u_5^{n+1,s} + u_6^{n+1,s}), \\ (u_{xx}^h)_{N-1}^{n+1,s} &= \frac{1}{12\Delta x^2} (10\gamma_2(t^{n+1}) - 15u_{N-1}^{n+1,s} - 4u_{N-2}^{n+1,s} \\ &\quad + 14u_{N-3}^{n+1,s} - 6u_{N-4}^{n+1,s} + u_{N-5}^{n+1,s}), \\ (u_{xx}^l)_{N-1}^{n+1,k} &= \frac{1}{\Delta x^2} (u_{N-2}^{n+1,k} - 2u_{N-1}^{n+1,k} + \gamma_2(t^{n+1})), \\ k &= s, s+1. \end{aligned} \quad (22)$$

Cases $s = 0$ or $k = 0$ give formulae to approximate $(u_{xx}^l)_i^n$ and $(u_{xx}^h)_i^n$. Substituting (13), (14), and (22) into (12) the

following fourth-order deferred correction approximations of $(u_{xx})_i^{n+1,s+1}$, $i = 2, \dots, N-1$, are

$$\begin{aligned} (u_{xx})_2^{n+1,s+1} &= \frac{5}{6\Delta x^2} \gamma_1(t^{n+1}) \\ &\quad + \frac{1}{\Delta x^2} (-2u_2^{n+1,s+1} + u_3^{n+1,s+1}) \\ &\quad + \frac{1}{12\Delta x^2} (9u_2^{n+1,s} - 16u_3^{n+1,s} + 14u_4^{n+1,s} \\ &\quad - 6u_5^{n+1,s} + u_6^{n+1,s}), \\ (u_{xx})_i^{n+1,s+1} &= \frac{1}{\Delta x^2} (u_{i-1}^{n+1,s+1} - 2u_i^{n+1,s+1} + u_{i+1}^{n+1,s+1}) \\ &\quad + \frac{1}{12\Delta x^2} (-u_{i-2}^{n+1,s} + 4u_{i-1}^{n+1,s} - 6u_i^{n+1,s} \\ &\quad + 4u_{i+1}^{n+1,s} - u_{i+2}^{n+1,s}), \\ i &= 3, \dots, N-2, \\ (u_{xx})_{N-1}^{n+1,s+1} &= \frac{5}{6\Delta x^2} \gamma_2(t^{n+1}) \\ &\quad + \frac{1}{\Delta x^2} (-2u_{N-1}^{n+1,s+1} + u_N^{n+1,s+1}) \\ &\quad + \frac{1}{12\Delta x^2} (9u_{N-1}^{n+1,s} - 16u_{N-2}^{n+1,s} + 14u_{N-3}^{n+1,s} \\ &\quad - 6u_{N-4}^{n+1,s} + u_{N-5}^{n+1,s}). \end{aligned} \quad (23)$$

Schemes (23) can be combined and expressed in the following matrix form:

$$\mathbf{u}_{xx}^{n+1,s+1} = \frac{1}{\Delta x^2} \Lambda_t \mathbf{u}^{n+1,s+1} + \frac{1}{\Delta x^2} (\Lambda_h - \Lambda_t) \mathbf{u}^{n+1,s} + \gamma(t^{n+1}), \quad (24)$$

where Λ_l is a tridiagonal $(N - 2) \times (N - 2)$ matrix and Λ_h is the corresponding triangular and sparse $(N - 2) \times (N - 2)$ matrix,

$$\mathbf{u}^{n+1,k} = (u_2^{n+1,k}, u_3^{n+1,k}, \dots, u_{N-1}^{n+1,k})^T, \quad k = s, s + 1, \quad (25)$$

$$\mathbf{u}_{xx}^{n+1,s+1} = ((u_{xx})_2^{n+1,s+1}, (u_{xx})_3^{n+1,s+1}, \dots, (u_{xx})_{N-1}^{n+1,s+1})^T. \quad (26)$$

Substituting (6), (23) into (11), the formulae can be written into matrix form

$$\begin{aligned} (E - \alpha\Lambda_l) \mathbf{u}^{n+1,s+1} \\ = \alpha(\Lambda_h - \Lambda_l) \mathbf{u}^{n+1,s} + (E + \alpha\Lambda_h) \mathbf{u}^n \\ + \Delta t [\gamma(t^{n+1}) + \gamma(t^n)] + \Delta t \mathbf{f}^{n+1/2}. \end{aligned} \quad (27)$$

The above matrix form is called FODs for Dirichlet boundary value problem (1)–(3). Thomas algorithms can be used to compute the solutions of FODs. At each step of time t^n and the initial stage, the convergence of FODs requires more iterations to converge to the solution of the FOSs. The order of approximation of FODs is $O(\Delta t^2, \Delta x^4)$ which is the same as FOSs in the uniform norm.

2.3. Fourth-Order Compact Scheme. Let us briefly represent the main idea and final formulae of compact schemes. Spatial derivatives in the Crank-Nikolson scheme (4) are evaluated by the fourth-order compact finite differences implicit scheme [5, 7, 8, 13, 14, 17].

In [8, 14], the Dirichlet boundary conditions

$$u(0, m\Delta t) = \gamma_1(t^m) = u_1^m, \quad u(l, m\Delta t) = \gamma_2(t^m) = u_N^m \quad (28)$$

are used to derive the following fourth-order schemes

$$\begin{aligned} (u_{xx})_2^m + \sigma(u_{xx})_3^m \\ = \frac{1}{24\Delta x^2} (a_1 u_1^m + a_2 u_2^m + a_3 u_3^m + a_4 u_4^m \\ + a_5 u_5^m + a_6 u_6^m) \\ = \frac{a_1}{24\Delta x^2} \gamma_1(t^m) \\ + \frac{1}{24\Delta x^2} (a_2 u_2^m + a_3 u_3^m + a_4 u_4^m + a_5 u_5^m \\ + a_6 u_6^m), \end{aligned}$$

$$\begin{aligned} (u_{xx})_{i-1}^m + 10(u_{xx})_i^m + (u_{xx})_{i+1}^m \\ = \frac{2}{\Delta x^2} (6u_{i-1}^m - 12u_i^m + 6u_{i+1}^m), \\ i = 2, \dots, N - 1, \\ (u_{xx})_{N-1}^m + \sigma(u_{xx})_{N-2}^m \\ = \frac{1}{24\Delta x^2} (a_1 u_N^m + a_2 u_{N-1}^m + a_3 u_{N-2}^m + a_4 u_{N-3}^m \\ + a_5 u_{N-4}^m + a_6 u_{N-5}^m) \\ = \frac{a_1}{24\Delta x^2} \gamma_2(t^m) \\ + \frac{1}{24\Delta x^2} (a_2 u_{N-1}^m + a_3 u_{N-2}^m + a_4 u_{N-3}^m + a_5 u_{N-4}^m \\ + a_6 u_{N-5}^m), \end{aligned} \quad (29)$$

where the coefficients can be found by matching the Taylor series expansion of left-hand-side terms up to order $O(\Delta x^4)u^{(6)}$ which gives the following values of coefficients [8]:

$$\begin{aligned} \sigma = \frac{1}{2}, \quad a_1 = 19, \quad a_2 = -14, \quad a_3 = -38, \\ a_4 = 44, \quad a_5 = -13, \quad a_6 = 2. \end{aligned} \quad (30)$$

Then all derivatives in (4) are approximated by the fourth-order compact formula; we can write

$$A\mathbf{u}_{xx}^m = \frac{1}{\Delta x^2} B\mathbf{u}^m + \gamma^m, \quad m = n, n + 1, \quad (31)$$

where A and B are the corresponding triangular and sparse $(N - 2) \times (N - 2)$ matrices, $\mathbf{u}_{xx}^m = ((u_{xx})_2^m, (u_{xx})_3^m, \dots, (u_{xx})_{N-1}^m)^T$, $\mathbf{u}^m = (u_2^m, u_3^m, \dots, u_{N-1}^m)^T$ and $\gamma^m = (\gamma_1(t^m), 0, \dots, 0, \gamma_2(t^m))^T$, $m = n, n + 1$. Schemes (4) and (29) can be combined and expressed in the following matrix form:

$$\begin{aligned} (A - \alpha B) \mathbf{u}^{n+1} = (A + \alpha B) \mathbf{u}^n \\ + \Delta t [\gamma(t^{n+1}) + \gamma(t^n)] \\ + \Delta t \mathbf{f}^{n+1/2}. \end{aligned} \quad (32)$$

This scheme is called FOCs for Dirichlet boundary value problem (1)–(3). We like to mention that the above scheme has truncation error $O(\Delta t^2, \Delta x^4)$. Note that the triangular and sparse $(N - 2) \times (N - 2)$ coefficient matrices in FOCs are time independent; hence, we have to store the inverse of the coefficient matrix $A - \alpha B$ before the time marching in the implementation of computational efficiency.

3. Numerical Examples

In this section, three numerical examples are carried out. The first two are linear heat-conducting problem, with Dirichlet

boundary conditions, which are used to confirm our theoretical analysis. Then we apply the FODs to the Burgers equation. For simplicity, we fix our problem domain $\Omega = \{x \mid 0 \leq x \leq 1\}$. In all computations, we used $\Delta t = \Delta x^2/4$ and $\epsilon = 10^{-10}$. The following stopping criterion is used:

$$\max_{1 \leq i \leq N} |u_i^{n+1, \hat{S}+1} - u_i^{n+1, \hat{S}}| < \epsilon, \quad s = 0, \dots, \hat{S}, \quad (33)$$

where " $\hat{S}$ " denotes the number of the last iteration.

The computations are performed using uniform grids of 11, 21, 41, 81, and 161 nodes. The initial and boundary conditions are obtained based on the exact solutions. For the testing purpose only, all computations are performed for $0 \leq t \leq 1$.

Example 1 (the homogeneous heat equation with the homogeneous Dirichlet boundary conditions). One has

$$\begin{aligned} u_t &= u_{xx}, \quad 0 \leq x \leq 1, \quad t > 0, \\ u(x, 0) &= \sin(\pi x), \quad u(0, t) = 0, \quad u(1, t) = 0. \end{aligned} \quad (34)$$

The exact solution is $u(x, t) = e^{-\pi^2 t} \sin(\pi x)$. The results of performance over the time interval $t \in [0, 1]$ for the FOCs, FODs, and FOSs are represented in Table 1, where the maximum error and the rate of convergence at time instant $t = 1$ are shown.

Example 2 (the nonhomogeneous heat equation with non-homogeneous Dirichlet boundary conditions). One has

$$\begin{aligned} u_t &= u_{xx} + (\pi^2 - 1)e^{-t} \cos(\pi x) \\ &\quad + 4x - 2, \quad 0 \leq x \leq 1, \quad t > 0, \\ u(x, 0) &= \cos(\pi x) + x^2, \quad u(0, t) = e^{-t}, \\ u(1, t) &= -e^{-t} + 4t + 1. \end{aligned} \quad (35)$$

The exact solution is $u(x, t) = e^{-t} \cos(\pi x) + x^2 + 4xt$. The results of performance over the time domain $t \in [0, 1]$ for the FOC, FOD, and FOS schemes are represented in Table 2, where the maximum error and the rate of convergence at time instant $t = 1$ are shown.

The last two columns of Tables 1 and 2 demonstrate the average number of iterations in FODs at one time step and the CPU time required to obtain the solution at time instant $t = 1$. The average number of iterations means the total number of iterations divided by the number of time steps. As a rule, at the initial stage the convergence of deferred correction requires more iterations. For larger instants of time, the convergence occurs after 2~7 iterations as can be seen from Tables 1 and 2. All of schemes are seen to be the fourth order of accuracy, as the error is reduced approximately by factor four when the mesh is refined by half. The maximum error of the FODs and FOCs is almost the same, since the iterative scheme FODs is constructed by applying the deferred correction technique on the FOSs. It can be stated that when the iterations converge,

the solution of FODs, therefore, converges to the solution of FOSs in each step of time. As shown in Tables 1 and 2, there is hardly a difference in the computational efficiency between FODs and FOSs. Both schemes are more efficient than FOCs. An explanation is due to the iteration needed for the convergence of solutions on each step of time.

Although the FODs use more computational time as compared with FOCs and FOSs, it is recommended that the construction of FODs can be easily implemented. Moreover, the scheme does not need to store the inverse of coefficient matrices as required in FOCs and FOSs. Therefore, the method is easily extended to multidimensional cases.

It is suggested that the differed correction technique can solve problems which need high accuracy of computational methods. Also this technique can be easily implemented and extended for solving problem with Neumann boundary conditions. In addition, such technique can be easily used to create standard code and applied in case of nonuniform grids.

Considering Burgers equation

$$u_t = \beta u_{xx} - uu_x, \quad 0 \leq x \leq 1, \quad t > 0, \quad (36)$$

with the exact solution [21] is given by

$$u(x, t) = \frac{\xi + \eta + (\eta - \xi)e^\rho}{1 + e^\rho}, \quad (37)$$

where $\rho = \xi(x - \eta t - v)/\beta$. The initial and Dirichlet boundary conditions are considered to be in agreement with the exact solution proposed here. For Burgers equation (36), we solve it by the following fourth-order deferred correction scheme:

$$\frac{u_i^{n+1, s+1} - u_i^n}{\Delta t} = \frac{\beta}{2} [(u_{xx})_i^{n+1, s+1} + (u_{xx})_i^n] + f_i^n, \quad (38)$$

where $f_i^n = -[(u^2/2)_x]_i^n$. The nonlinear term f_i^n is approximated with the fourth-order approximation and all the second-derivative terms in (38) are approximated by the fourth-order formula (6) and the fourth-order deferred correction schemes (23). The scheme (38) can be combined and expressed in the following matrix form:

$$\begin{aligned} (E - \alpha \Lambda_l) \mathbf{u}^{n+1, s+1} &= \alpha (\Lambda_h - \Lambda_l) \mathbf{u}^{n+1, s} + (E + \alpha \Lambda_h) \mathbf{u}^n \\ &\quad + \Delta t [\gamma(t^{n+1}) + \gamma(t^n)] + \Delta t \mathbf{f}^n, \end{aligned} \quad (39)$$

where E is identity matrix, Λ_l is tridiagonal $(N-2) \times (N-2)$ matrix, and Λ_h is the corresponding triangular and sparse $(N-2) \times (N-2)$ matrix and can be solved by using Thomas algorithm.

Example 3 (the Burgers equation (36) and the constant values $\nu = 0.125$, $\xi = 0.6$, $\eta = 0.4$, and $\beta = 1$ with appropriate initial and Dirichlet boundary condition in agreement with exact solution (37)). This problem was solved using different time step and mesh sizes over the time interval $0 < t \leq 1$. The results of performance over the time interval $t \in [0, 1]$ for the FODs are represented in Tables 3 and 4, where the maximum error and the rate of convergence at time instant $t = 1$ are shown.

TABLE 1: Maximum absolute error, order of convergence, and CPU time in seconds of the FOCs, FODs, and FOSs for test problem (34) at time instant $t = 1$.

Types of scheme	Grid points	Maximum error	Rate of convergence	Aver. number of iteration	CPU time in sec.
FOCs	11	3.8687×10^{-8}	—	1	0.006
	21	6.0426×10^{-10}	6.0005	1	0.042
	41	2.2454×10^{-11}	4.7501	1	0.326
	81	1.2821×10^{-12}	4.1304	1	2.564
	161	8.0164×10^{-14}	3.9994	1	20.437
FODs	11	9.9767×10^{-9}	—	4	0.015
	21	1.4996×10^{-9}	2.7361	3	0.085
	41	1.1193×10^{-10}	3.7438	2	0.438
	81	7.1438×10^{-12}	3.9698	2	3.450
	161	4.4797×10^{-13}	4.1875	2	27.495
FOSs	11	9.9763×10^{-9}	—	1	0.006
	21	1.4996×10^{-9}	2.7361	1	0.043
	41	1.1193×10^{-10}	3.7438	1	0.334
	81	7.1440×10^{-12}	3.9698	1	2.623
	161	4.4854×10^{-13}	4.1875	1	20.907

TABLE 2: Absolute error, the rate of convergence, and CPU time in seconds of the FOCs, FODs, and FOSs for the test problem (35) at time instant $t = 1$.

Types of scheme	Grid points	Maximum error	Rate of convergence	Aver. number of iteration	CPU time in sec.
FOCs	11	1.8470×10^{-5}	—	1	0.006
	21	3.6901×10^{-7}	5.6454	1	0.046
	41	7.5595×10^{-9}	5.6092	1	0.353
	81	6.6458×10^{-10}	3.5077	1	2.778
	161	4.8841×10^{-11}	3.7663	1	22.141
FODs	11	1.7132×10^{-5}	—	7	0.016
	21	2.6914×10^{-7}	5.9922	7	0.128
	41	2.7910×10^{-8}	3.2655	6	0.851
	81	2.0112×10^{-9}	3.7941	5	5.568
	161	1.3116×10^{-10}	3.9415	5	44.375
FOSs	11	1.2895×10^{-5}	—	1	0.006
	21	2.8544×10^{-7}	5.9922	1	0.046
	41	2.7306×10^{-8}	3.2655	1	0.359
	81	1.9590×10^{-9}	3.7941	1	2.821
	161	1.3130×10^{-10}	3.9415	1	22.484

TABLE 3: Maximum absolute error, order of convergence, and CPU time in seconds for Example 3 at time instant $t = 1$ with fixed mesh size $\Delta x = 0.05$.

Types of scheme	Time step sizes	Maximum error	Rate of convergence	Aver. number of iteration	CPU time in sec.
FODs	10^{-2}	8.0945×10^{-6}	—	10	0.015
	10^{-3}	8.0942×10^{-7}	1.0000	6	0.109
	10^{-4}	8.1144×10^{-8}	0.9989	3	0.656
	10^{-5}	8.2989×10^{-9}	0.9902	3	6.281

TABLE 4: Maximum absolute error, order of convergence, and CPU time in seconds for Example 3 at time instant $t = 1$ with time step size $\Delta t = \Delta x^4$.

Types of scheme	Grid points	Maximum error	Rate of convergence	Aver. number of iteration	CPU time in sec.
FODs	11	8.3375×10^{-8}	—	3	0.516
	21	5.2632×10^{-9}	3.9812	2	8.594
	41	3.3491×10^{-10}	3.9774	2	200.015

In order to analyze the results found in application to the Burgers equation (36), Table 3 demonstrates rate of convergence, average number of iteration at each time step, and CPU time required to obtain the solution of Example 3 by using FODs at time instant $t = 1$ when $\Delta x = 0.05$ with various time step sizes. Table 4 shows the rate of convergence, average number of iteration at each time step, and CPU time required to obtain the solution of Example 3 at time instant $t = 1$ and using uniform grids of 11, 21, and 41 with time step sizes $\Delta t = \Delta x^4$ and $\epsilon = 10^{-10}$.

It can be seen from Tables 3 and 4 that numerical results are in good agreement with the exact solution. We only observe $O(\Delta t)$ convergence rate and the error is dominated by time error. An explanation for this phenomenon is due to the nonlinear term, which is approximated at time level n , instead of at time level $n + 1/2$ for the FODs (38).

4. Conclusion

In this paper, a new set of fourth-order schemes for the one-dimensional heat conduction problem with Dirichlet boundary conditions is constructed using a deferred correction technique. The construction of high-order deferred correction schemes requires only a regular three-point stencil at higher time level which is similar to the standard second-order Crank-Nikolson method. The greatest significance of FODs, compared with FOCs and FOSs, is the easier development and that it can be solved by using Thomas algorithms. Numerical examples confirm the order of accuracy. We also implement our algorithms to nonlinear problems. However, theoretical analysis for nonlinear problems needs further investigation. Posterior idea for this project is to use another way to make uu_x term as follows [21, 22]:

$$u^{n+1}(u_x)_i^{n+1} \approx u^{n+1}(u_x)_i^n + u^n(u_x)_i^{n+1} - u^n(u_x)_i^n, \quad (40)$$

where better results are expected to be found. The first two terms on the right-hand side of above equation make the coefficient matrices of FOCs, FODs, and FOSs vary with time. That is, the inverse coefficient matrices of FOCs and FOSs have to be stored on each step of time while FODs have no need. For this reason, the FODs is simple to implement although FODs need more iterations for the convergence of solution on each step of time.

Acknowledgments

This work is financially supported by the Commission on Higher Education (CHE), the Thailand Research Fund (TRF), the University of Phayao (UP), Project MRG5580014. The author would like to express their deep appreciation to Professor Sergey Meleshko for the kind assistance and valuable advice on the REDUCE calculations.

References

- [1] Y. Adam, "Highly accurate compact implicit methods and boundary conditions," *Journal of Computational Physics*, vol. 24, no. 1, pp. 10–22, 1977.
- [2] G. F. Carey and W. F. Spitz, "Higher-order compact mixed methods," *Communications in Numerical Methods in Engineering with Biomedical Applications*, vol. 13, no. 7, pp. 553–564, 1997.
- [3] M. H. Carpenter, D. Gottlieb, and S. Abarbanel, "Stable and accurate boundary treatments for compact, high-order finite-difference schemes," *Applied Numerical Mathematics*, vol. 12, no. 1–3, pp. 55–87, 1993.
- [4] I. Christie, "Upwind compact finite difference schemes," *Journal of Computational Physics*, vol. 59, no. 3, pp. 353–368, 1985.
- [5] P. C. Chu and C. Fan, "A three-point combined compact difference scheme," *Journal of Computational Physics*, vol. 140, no. 2, pp. 370–399, 1998.
- [6] P. C. Chu and C. Fan, "A three-point sixth-order nonuniform combined compact difference scheme," *Journal of Computational Physics*, vol. 148, no. 2, pp. 663–674, 1999.
- [7] W. Dai and R. Nassar, "A compact finite difference scheme for solving a three-dimensional heat transport equation in a thin film," *Numerical Methods for Partial Differential Equations*, vol. 16, no. 5, pp. 441–458, 2000.
- [8] W. Dai and R. Nassar, "Compact ADI method for solving parabolic differential equations," *Numerical Methods for Partial Differential Equations*, vol. 18, no. 2, pp. 129–142, 2002.
- [9] W. Dai, "A new accurate finite difference scheme for Neumann (insulated) boundary condition of heat conduction," *International Journal of Thermal Sciences*, vol. 49, no. 3, pp. 571–579, 2010.
- [10] X. Deng and H. Maekawa, "Compact high-order accurate nonlinear schemes," *Journal of Computational Physics*, vol. 130, no. 1, pp. 77–91, 1997.
- [11] J. H. Ferziger and M. Peric, *Computational Methods in Fluid Dynamics*, Springer, Berlin, Germany, 2002.
- [12] J. C. Kalita, D. C. Dalal, and A. K. Dass, "A class of higher order compact schemes for the unsteady two-dimensional convection-diffusion equation with variable convection coefficients," *International Journal for Numerical Methods in Fluids*, vol. 38, no. 12, pp. 1111–1131, 2002.
- [13] S. Karaa and J. Zhang, "High order ADI method for solving unsteady convection-diffusion problems," *Journal of Computational Physics*, vol. 198, no. 1, pp. 1–9, 2004.
- [14] S. K. Lele, "Compact finite difference schemes with spectral-like resolution," *Journal of Computational Physics*, vol. 103, no. 1, pp. 16–42, 1992.
- [15] J. Li, Y. Chen, and G. Liu, "High-order compact ADI methods for parabolic equations," *Computers & Mathematics with Applications*, vol. 52, no. 8–9, pp. 1343–1356, 2006.
- [16] I. M. Navon and H. A. Riphagen, "An implicit compact fourth-order algorithms for solving the shallow-water equations in conservation-law form," *Monthly Weather Review*, vol. 107, no. 9, pp. 1107–1127, 1979.
- [17] J. Zhao, W. Dai, and T. Niu, "Fourth-order compact schemes of a heat conduction problem with Neumann boundary conditions," *Numerical Methods for Partial Differential Equations*, vol. 23, no. 5, pp. 949–959, 2007.
- [18] J. Zhao, W. Dai, and S. Zhang, "Fourth-order compact schemes for solving multidimensional heat problems with Neumann boundary conditions," *Numerical Methods for Partial Differential Equations*, vol. 24, no. 1, pp. 165–178, 2008.
- [19] V. Pereyra, "On improving an approximate solution of a functional equation by deferred corrections," *Numerische Mathematik*, vol. 8, pp. 376–391, 1966.

- [20] V. Pereyra, "Iterated deferred corrections for nonlinear boundary value problems," *Numerische Mathematik*, vol. 11, pp. 111–125, 1968.
- [21] M. C. Miriane, M. G. Paulo, and C. R. Estaner, "Error analysis in the solution of unsteady nonlinear convection-diffusion problems," *Research Journal of Physical and Applied Science*, vol. 1, no. 1, pp. 20–22, 2012.
- [22] B. Jiang, *The Least-Squares Finite Element Method: Theory and Applications in Computational Fluid Dynamics and Electromagnetics*, Scientific Computation, Springer, Berlin, Germany, 1998.