Pré-Publicações do Departamento de Matemática
Universidade de Coimbra
Preprint Number 05–07

# DEVELOPMENT OF FINITE DIFFERENCE SCHEMES NEAR AN INFLOW BOUNDARY

ERCÍLIA SOUSA

ABSTRACT: Numerical schemes for a convection-diffusion problem defined on the whole real line have been derived by Morton and Sobey [1] using the exact evolution operator through one time step. In this paper we derive new numerical schemes by using the exact evolution operator for a convection-diffusion problem defined on the half-line. We obtain a third order method that requires the use of a numerical boundary condition which is also derived using the same evolution operator. We determine whether there are advantages from the point of view of stability and accuracy in using these new schemes, when compared with similar methods obtained for the whole line. We conclude that the third order scheme provides gains in terms of stability and although it does not improve the practical accuracy of existing methods faraway from the inflow boundary, it does improve the accuracy next to the inflow boundary.

KEYWORDS: finite differences, convection-diffusion, stability, accuracy.

## 1. Introduction

The mechanism of convection diffusion appears in many physical applications and accurate modeling of the interaction between convective and diffusive processes can be a difficult task. Although the majority of physical experiments are performed in the presence of boundaries if we consider the approximation of the unsteady convection-diffusion problem, we can observe that much of the literature is concerned with choices for the whole real line.

It is very common that the approximate solutions we derive for the whole line present some difficulties when we need to deal with the presence of a physical boundary. This difficulty is more obvious if we are interested in simulations next to the boundary and at short times. Even if they perform efficiently far away from the physical boundary, next to it they can have a poor performance. In this paper, we present new finite difference schemes derived taking into account the existence of an inflow physical boundary.

Finite difference schemes typically consist of replacement of the individual derivative terms in the partial differential equation by a set of discretised approximations (see e.g. Smith [2]). However, recently different techniques

Received April 27, 2005.

were suggested for deriving finite differences for the unsteady convection-diffusion equation (see e.g. Morton and Sobey [1] and Xu *et al* [3]). In the next section, we use the framework described in Morton and Sobey [1] to obtain finite difference schemes taking into account the presence of a physical boundary.

Related with the convergence of a finite difference scheme we encounter questions about stability and accuracy and the presence of a boundary most likely will affect stability and accuracy of the overall numerical scheme. In the third section we study the stability and accuracy of the numerical schemes and in the fourth section, to analyse the performance of the third order scheme, we present two test problems.

## 2. The finite difference schemes

Consider the one-dimensional problem of convection with constant velocity $V$ in the positive $x$ direction and constant diffusion $D > 0$:

$$\frac{\partial u}{\partial t} + V\frac{\partial u}{\partial x} = D\frac{\partial^2 u}{\partial x^2}, \quad t > 0, \quad x > 0, \tag{1}$$

with the initial condition

$$u(x,0) = f(x), \quad x \geq 0, \tag{2}$$

and subject to the boundary conditions

$$u(x,t) \to 0,\ x \to \infty \qquad \text{and} \qquad u(0,t) = g(t),\ t \geq 0. \tag{3}$$

The exact solution of the system (1), (2) and (3) can be found using Laplace transforms in $t$:

$$\begin{aligned} u(x,t) &= \frac{1}{\sqrt{\pi}}\int_0^t g(t-\tau)G^*(x,\tau)d\tau \\ &+ \frac{1}{\sqrt{\pi}}\int_{\frac{Vt-x}{2\sqrt{Dt}}}^{+\infty} f(x - Vt + 2\sqrt{Dt}\xi)\mathrm{e}^{-\xi^2}d\xi \\ &- \frac{1}{\sqrt{\pi}}\int_{\frac{Vt+x}{2\sqrt{Dt}}}^{+\infty} f(-x - Vt + 2\sqrt{Dt}\xi)\mathrm{e}^{Vx/D}\mathrm{e}^{-\xi^2}d\xi \end{aligned} \tag{4}$$

where the function $G^*(x,\tau)$ is given by

$$G^*(x,\tau) = \frac{x}{2\sqrt{D}\tau^{2/3}}\mathrm{e}^{-(x-V\tau)^2/4D\tau}. \tag{5}$$

Applying the result to evolution over one time step, we write,

$$u(x,t_n+\Delta t)=\frac{1}{\sqrt{\pi}}\int_0^{\Delta t} g(t_n+\Delta t-\tau)G^*(x,\tau)d\tau$$

$$+\frac{1}{\sqrt{\pi}}\int_{\frac{V\Delta t-x}{2\sqrt{D\Delta t}}}^{+\infty} u(x-V\Delta t+2\sqrt{D\Delta t}\xi,t_n)\mathrm{e}^{-\xi^2}d\xi \tag{6}$$

$$-\frac{1}{\sqrt{\pi}}\int_{\frac{V\Delta t+x}{2\sqrt{D\Delta t}}}^{+\infty} u(-x-V\Delta t+2\sqrt{D\Delta t}\xi,t_n)\mathrm{e}^{Vx/D}\mathrm{e}^{-\xi^2}d\xi.$$

The exact solution for this model problem differs from the solution of a convection-diffusion problem on the whole real line. This is the fundamental solution we shall use, to derive approximation schemes by allowing a local solution to evolve and then restricting the evolved solution to an approximation space.

We can rewrite the evolution operator over one time step, given by (6), in terms of a Green's function:

$$\begin{aligned} u(x,t_n+\Delta t) &= \frac{1}{\sqrt{\pi}}\int_0^{\Delta t} g(t_n+\Delta t-\tau)G^*(x,\tau)d\tau \\ &\quad +\frac{1}{\sqrt{\pi}}\int_0^{+\infty} u(\eta,t_n)G_1(x,\eta,\Delta t)d\eta, \end{aligned} \tag{7}$$

where

$$G_1(x,\eta,\beta) = \frac{\mathrm{e}^{-(\eta-x-V\beta)^2/4D\beta}}{2\sqrt{D\beta}}[1-\mathrm{e}^{\eta x/D\beta}].$$

To derive finite difference approximations we substitute a local polynomial approximation to $u(\eta,t_n)$ in (7), and then carry out the integration of a global polynomial. Suppose we have approximations $\mathbf{U}^n := \{U_j^n\}$ to the values $u(x_j,t_n)$ at the mesh points

$$x_j = j\Delta x, \quad j=0,1,2,\ldots.$$

We associate with each point $x_j$ a local interpolating polynomial through $U_j^n$ and the values at a certain number of neighbouring points. Denoting each such polynomial by $p_j(x;\mathbf{U}^n)$, we generate finite difference schemes from

$$U_j^{n+1} = \frac{1}{\sqrt{\pi}}\int_0^{\Delta t} g(t_n+\Delta t-\tau)G^*(x,\tau)d\tau$$

$$+\frac{1}{\sqrt{\pi}}\int_0^{+\infty} p_j(\eta;\mathbf{U}^n)G_1(x_j,\eta;\Delta t)d\eta. \tag{8}$$

The approximation scheme which we obtain comes from approximating $\mathbf{U}^n$ near $x_j$ by a polynomial $p_j(x;\mathbf{U}^n)$, of degree $R$,

$$p_j(x;\mathbf{U}^n)=\sum_{r=0}^{R} b_{jr}(x-x_j)^r.$$

Then

$$\begin{aligned}
U_j^{n+1} &= \frac{1}{\sqrt{\pi}}\int_0^{\Delta t} g(t_n+\Delta t-\tau)G^*(x,\tau)d\tau \\
&+\frac{1}{\sqrt{\pi}}\int_{\frac{\nu-j}{2\sqrt{\mu}}}^{+\infty} p_j(x_j-V\Delta t+2\sqrt{D\Delta t}\xi;\mathbf{U}^n)\mathrm{e}^{-\xi^2}d\xi \\
&-\frac{1}{\sqrt{\pi}}\int_{\frac{\nu+j}{2\sqrt{\mu}}}^{+\infty} p_j(-x_j-V\Delta t+2\sqrt{D\Delta t}\xi;\mathbf{U}^n)\mathrm{e}^{j\nu/\mu}\mathrm{e}^{-\xi^2}d\xi,
\end{aligned} \tag{9}$$

where

$$\nu=\frac{V\Delta t}{\Delta x} \quad \text{and} \quad \mu=\frac{D\Delta t}{\Delta x^2}.$$

First, for clarity, we assume that the left boundary condition is zero, that is, $g(t)=0$. Then the first integral in (9) is zero and we can write after integration of the polynomial form,

$$\begin{aligned}
U_j^{n+1} &= b_{j0}[\frac{1}{2}\mathrm{Erfc}(\frac{\nu-j}{2\sqrt{\mu}})-\frac{1}{2}\mathrm{e}^{\nu j/\mu}\mathrm{Erfc}(\frac{\nu+j}{2\sqrt{\mu}})] \\
&+b_{j1}[-V\Delta t\frac{1}{2}\mathrm{Erfc}(\frac{\nu-j}{2\sqrt{\mu}})+(2x_j+V\Delta t)\frac{1}{2}\mathrm{e}^{\nu j/\mu}\mathrm{Erfc}(\frac{\nu+j}{2\sqrt{\mu}})] \\
&+b_{j2}[(V^2(\Delta t)^2+2D\Delta t)\frac{1}{2}\mathrm{Erfc}(\frac{\nu-j}{2\sqrt{\mu}}) \\
&-((2x_j+V\Delta t)^2+2D\Delta t)\frac{1}{2}\mathrm{e}^{\nu j/\mu}\mathrm{Erfc}(\frac{\nu+j}{2\sqrt{\mu}})+2\frac{\sqrt{D\Delta t}}{\sqrt{\pi}}x_j\mathrm{e}^{-(\nu-j)^2/4\mu}] \\
&+b_{j3}[-(V^3(\Delta t)^3+6VD(\Delta t)^2)\frac{1}{2}\mathrm{Erfc}(\frac{\nu-j}{2\sqrt{\mu}})
\end{aligned}$$

$$+(2x_j + V\Delta t)((2x_j + V\Delta t)^2 + 6D\Delta t)\frac{1}{2}\mathrm{e}^{\nu j/\mu}\mathrm{Erfc}(\frac{\nu + j}{2\sqrt{\mu}})$$
$$-2(2V\Delta t + 3x_j)\frac{\sqrt{D\Delta t}}{\sqrt{\pi}}x_j\mathrm{e}^{-(\nu - j)^2/4\mu}] + \dots,$$

where $\mathrm{Erfc}(x)$ is the complementary error function $\mathrm{Erfc}(x) = \frac{2}{\sqrt{\pi}}\int_x^\infty \mathrm{e}^{-t^2}dt.$

Within this general framework we can now obtain finite difference schemes by interpolation on a uniform mesh. We use the usual central, backward and second difference operators to evaluate the coefficients $b_{jr},\ r = 0, 1, 2, \dots$ in terms of the nodal values $\mathbf{U}^n$,

$$\Delta_0 U_j = \frac{U_{j+1} - U_{j-1}}{2}, \quad \Delta_- U_j = U_j - U_{j-1}, \quad \text{and} \quad \delta^2 U_j = U_{j+1} - 2U_j + U_{j-1}.$$

We present two new numerical schemes using a quadratic interpolant and a cubic interpolant, obtaining in that way the schemes that we call the Modified Lax-Wendroff scheme and the Modified Quickest scheme. Other methods of higher order could be obtained by using higher order interpolants.

Using the quadratic interpolant of $U^n_{j-1}$, $U^n_j$ and $U^n_{j+1}$ we have the approximation formula for $U^{n+1}_j$, $j \geq 1$

$$U^{n+1}_j = a(j)[1 - \nu\Delta_0 + (\frac{\nu^2}{2} + \mu)\delta^2]U^n_j + b(j)\Delta_0 U^n_j + c(j)\delta^2 U^n_j, \tag{10}$$

where

$$\begin{aligned} a(j) &= \frac{1}{2}\mathrm{Erfc}(\frac{\nu - j}{2\sqrt{\mu}}) - \frac{1}{2}\mathrm{e}^{\nu j/\mu}\mathrm{Erfc}(\frac{\nu + j}{2\sqrt{\mu}}) \\ b(j) &= j\mathrm{e}^{\nu j/\mu}\mathrm{Erfc}(\frac{\nu + j}{2\sqrt{\mu}}) \\ c(j) &= -j(j + \nu)\mathrm{e}^{\nu j/\mu}\mathrm{Erfc}(\frac{\nu + j}{2\sqrt{\mu}}) + Z(j) \end{aligned}$$

where $Z(j) = \frac{\sqrt{\mu}}{\sqrt{\pi}}j\mathrm{e}^{-(\nu - j)^2/4\mu}$. We call this scheme the Modified Lax-Wendroff scheme since for $a(j) = 1$, $b(j) = 0$ and $c(j) = 0$, we obtain the Lax-Wendroff scheme [4], [5].

Although we will concentrate our attention on the scheme that follows, which is obtained using a cubic term, we have shown above, for completeness, the derivation of the Modified Lax-Wendroff scheme.

If $p_j(x, \mathbf{U}^n)$ is extended to include a cubic term, using the interpolation points $U_{j-2}^n$, $U_{j-1}^n$, $U_j^n$ and $U_{j+1}^n$ then the approximation formula for $j \geq 2$ becomes

$$\begin{aligned} U_j^{n+1} &= a(j)[1 - \nu\Delta_0 + (\frac{\nu^2}{2} + \mu)\delta^2 + \frac{\nu}{6}(1 - \nu^2 - 6\mu)\delta^2\Delta_-]U_j^n \\ &\quad + b(j)\Delta_0 U_j^n + c(j)\delta^2 U_j^n + d(j)\delta^2\Delta_- U_j^n, \end{aligned} \tag{11}$$

where

$$\begin{aligned} d(j) &= -\frac{1}{6}b(j) + \frac{1}{6}e(j) \\ e(j) &= (4j^3 + 2j^2\nu + j\nu^2 + 6j\mu)\mathrm{e}^{\nu j/\mu}\mathrm{Erfc}(\frac{\nu + j}{2\sqrt{\mu}}) - 2(2\nu + 3j)Z(j). \end{aligned}$$

We call this scheme the Modified Quickest scheme since in (11), for $a(j) = 1$, $b(j) = 0$, $c(j) = 0$ and $d(j) = 0$ we obtain the Quickest scheme [6].

To obtain the scheme (11) we interpolate at two points upwind but we do not have these points for interpolation around the first point of the mesh. Here, therefore, we need to consider a numerical boundary condition at the first mesh point. At this point we perform a cubic interpolation of the points $U_0^n$, $U_1^n$, $U_2^n$, $U_3^n$, namely

$$\begin{aligned} U_1^{n+1} &= a(1)[1 - \nu\Delta_0 + (\frac{\nu^2}{2} + \mu)\delta^2 + \frac{\nu}{6}(1 - \nu^2 - 6\mu)\delta^2\Delta_+]U_1^n \\ &\quad + b(1)\Delta_0 U_1^n + c(1)\delta^2 U_1^n + d(1)\delta^2\Delta_+ U_1^n, \end{aligned} \tag{12}$$

where $\Delta_+$ is the forward difference operator defined by $\Delta_+ U_j^n = U_{j+1}^n - U_j^n$.

When $j \to \infty$ we have

$$\mathrm{Erfc}\left(\frac{\nu - j}{2\sqrt{\mu}}\right) \to 1, \quad \mathrm{Erfc}\left(\frac{\nu + j}{2\sqrt{\mu}}\right) \to 0, \quad Z(j) \to 0,$$

and

$$a(j) \to 1 \quad b(j) \to 0 \quad c(j) \to 0 \quad d(j) \to 0.$$

Therefore, these new schemes are considerably different from the Lax-Wendroff scheme and Quickest scheme at the first points of the mesh, but are only slightly different at the other mesh points. In the next section we discuss the stability and accuracy of the new schemes.

## 3. Stability and accuracy

To analyse the stability of the new schemes we cannot use the von Neumann stability analysis since the coefficients are not linear, although under general conditions (see Richtmyer and Morton [7]) it can be proved that for linear, non-constant coefficient problems a local von Neumann analysis will provide a necessary condition for stability. The more natural option in this case is to use the spectrum and matrix analysis based on the observation of the norm and spectrum behaviour of the iterative matrix. Also the matrix method provides information on the influence of boundary conditions.

Concerning accuracy, to calculate the local truncation error we cannot apply the modified equation as described in Warming and Hyett [8], since we have non-linear terms. On the other hand we can derive formal truncation error estimates in the same way as suggested in Morton and Sobey [1] by applying the Peano kernel theorem (see Powell [9]).

**3.1. Stability analysis of the new schemes.** The explicit methods we discuss can be written in the form of a matrix iteration. Assume that the nodal points are $U_j^n, j = 0, \ldots, N$ and that the outflow boundary is such that

$$U_N^n = 0, \quad \forall n. \tag{13}$$

The choice of this outflow boundary is motivated by the fact that we assume that the exact solution goes to zero when $x$ goes to infinity.

Introducing the vector $U^n = \{U_0^n, U_1^n, \ldots, U_{N-1}^n\}^T$, all the schemes may be written as matrix equations

$$U^{n+1} = AU^n + \mathrm{v}^n, \quad n = 0, 1, 2, \ldots. \tag{14}$$

where $A$ is an $N \times N$ matrix and depends on the scheme used and $\mathrm{v}^n$ appears when the inflow boundary condition is not zero.

Any errors $E^n$ in a calculation based on (14) will grow according to

$$E^{n+1} = AE^n, \quad n = 0, 1, 2, \ldots. \tag{15}$$

where $E^n = u^n - U^n$ with $u^n$, $U^n$ the exact and numerical solutions of (14), respectively, at $t = n\Delta t$.

Given $A \in \mathbb{R}^{N\times N}$ denote the spectral radius of $A$ by $\rho(A)$ and the $L_2$-norm of the matrix $A$ by $||A||$. We recall that

$$||A|| = \rho(A) \quad \text{if} \quad A \in \mathbb{R}^{N\times N} \quad \text{is normal.}$$

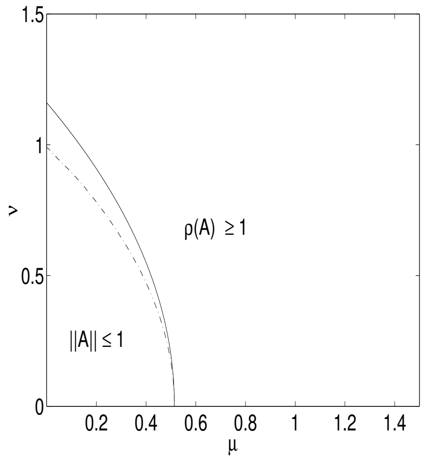



FIGURE 1. Stability region for the Modified Lax-Wendroff scheme: $\rho(A) = 1$ $(-)$ and $||A|| = 1$ $(- \cdot -)$.

It is well known that for any $A \in \mathbb{R}^{N\times N}$

$$A^m \to 0 \quad \text{as} \quad m \to \infty \quad \text{if and only if} \quad \rho(A) < 1,$$

and that

$$\rho(A) \leq ||A||.$$

A simple criterion for regulating the error growth governed by (15) is given by

$$\rho(A) \leq 1. \tag{16}$$

When the matrix $A$ is not normal the spectral radius gives no indication of the magnitude of $E^n$ for finite $n$. In this case a condition of the form $\rho(A) < 1$ guarantees eventual decay of the solution, but does not control the intermediate growth of the solution.

A more severe condition for regulating error growth follows from (15). If the matrix norm, $||A||$, is consistent with the vector norm, $||E||$, then

$$||E^{n+1}|| \leq ||A|| ||E^n||, \quad n = 0, 1, 2, \ldots,$$

and the condition

$$||A|| \leq 1, \tag{17}$$

is sufficient to ensure that the error cannot grow with $n$.

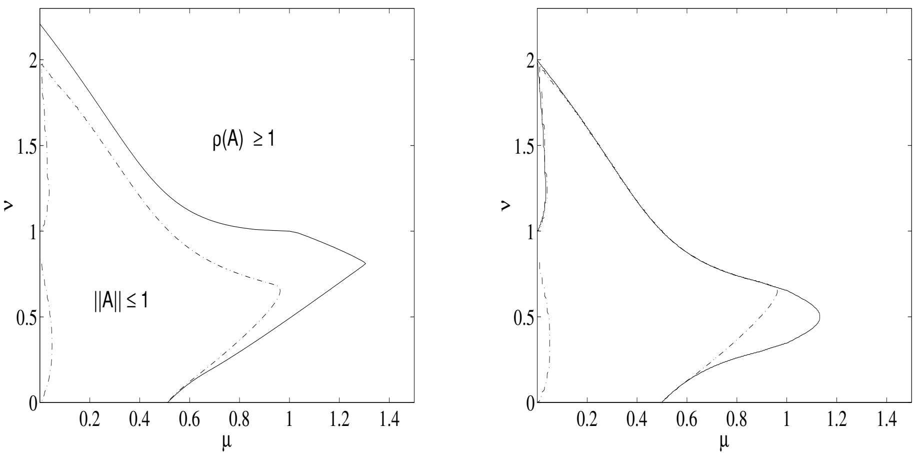
From (15) we have

$$E^n = A^n E^0, \quad n = 1, 2, \ldots. \tag{18}$$

The expression (18) shows that in order for all $E^n$ to remain bounded and the scheme (14) to remain stable the infinite set of operators $A^n$ has to be uniformly bounded for all $n$, $\Delta t$ and $\Delta x$.

Our stability analysis consists essentially in applying the sufficient condition for stability $||A|| \leq 1$ with the necessary stability condition $\rho(A) \leq 1$. We will plot the regions using MATLAB and for the matrix size $N = 30$.

The stability region for the scheme derived using a quadratic polynomial approximation, which we called the Modified Lax-Wendroff scheme, is given by figure 1. Comparing figure 1 with Lax-Wendroff von Neumann stability region, we observe that the stability region is the same, assuming it to be the region where $||A|| \leq 1$. In this case we do not have an advantage in terms of stability by choosing the Modified Lax-Wendroff scheme instead of the Lax-Wendroff scheme.




(a) (b)

FIGURE 2. Stability region for the Modified Quickest scheme with the numerical boundary condition (12): (a)$\rho(A) = 1$ $(-)$ and $||A|| = 1$ $(- \cdot -)$; (b) $||A|| = 1$ $(- \cdot -)$ and practical von Neumann stability for the Quickest scheme $(-)$

Consider the scheme (11), derived using a cubic polynomial approximation and called Modified Quickest scheme associated with the numerical boundary condition (12). The stability region is given by the region plotted in figure 2. Additionally, we plot in figure 2b the practical von Neumann stability region for the Quickest scheme. This allows us to see in what way it relates with the region $||A|| \leq 1$, where $A$ is the iterative matrix of the Modified Quickest scheme with the numerical boundary condition (12). The region for small $\mu$ where $||A|| \geq 1$ and $\rho(A) \leq 1$, plotted in the left corner of figure 2a and figure 2b is a region of practical stability. This was checked by running numerical experiments on this region.

It is important to remember that $||A|| \leq 1$ is a sufficient condition for stability but not a necessary one. In fact, experimentally the new scheme seems to be stable in all the von Neumann region displayed in figure 2b.

To conclude, we observe there are advantages in terms of stability in using the Modified Quickest scheme associated with the suggested numerical boundary condition, since we do not have the usual penalties in stability associated with the presence of a numerical boundary condition (see for instance Sousa and Sobey [10]).

**3.2. Accuracy of the new schemes.** We can derive truncation error estimates in the same way as suggested in Morton and Sobey [1] by applying the Peano kernel theorem (see Powell [9]). We consider the error committed in one time step.

**Theorem**: For the scheme derived using the quadratic interpolant, the Modified Lax-Wendroff, we have

$$\Delta t T_{|_{x_j}} = \frac{1}{6}\Delta x^3 u_{xxx} g_j^2(\nu,\mu) + O(\Delta x^4 u_{x^4}), \tag{19}$$

where

$$g_j^2(\nu,\mu) = \nu(1-\nu^2-6\mu)a(j) - b(j) + e(j).$$

For the Modified Quickest scheme, obtained using a cubic interpolant, the exact error for $x_j, j \geq 2$ is given by the cumbersome expression

$$\Delta t T_{|_{x_j}} = \frac{1}{24}\Delta x^4 u_{xxxx} g_j^3(\nu,\mu) + O(\Delta x^5 u_{x^5}), \tag{20}$$

with

$$\begin{aligned} g_j^3(\nu,\mu) \;=\;& (12\mu^2 - 2\mu - 12\mu\nu(1-\nu) + \nu(1-\nu^2)(2-\nu))a(j) \\ & -2b(j)(12\mu\nu + 2\nu^3 + 2j\nu + 1 + 2j^3) \\ & +2c(j)(1+6j^2) + 2(1-2j)e(j) + 2Z(j)(3\nu^2 + j^2 + 10\mu). \end{aligned}$$

**Proof**: The details of the proof can be seen in Appendix A. □

When $j \to \infty$ then $a(j) = 1, b(j) = c(j) = e(j) = 0$ and the expressions (19) and (20) are the truncation errors obtained for the Lax-Wendroff scheme and Quickest scheme respectively. Consequently, for each scheme, near the boundary we shall have a different truncation error which nevertheless does not have an inferior order.

These results indicate that the new schemes have similarities in terms of accuracy with the Lax-Wendroff and Quickest schemes respectively.

It is well know that numerical boundary conditions do interfere with the stability region of the scheme. This raises the question whether by using the Quickest scheme, instead of the Modified Quickest scheme, with the numerical boundary condition (12), we would still have gains in stability. We check this possibility in the next section.

**3.3. Stability of mixed schemes.** It was seen in Sousa and Sobey [10] that the choice of numerical boundary conditions may strongly affect the stability of a Quickest scheme even if the accuracy is not affected.

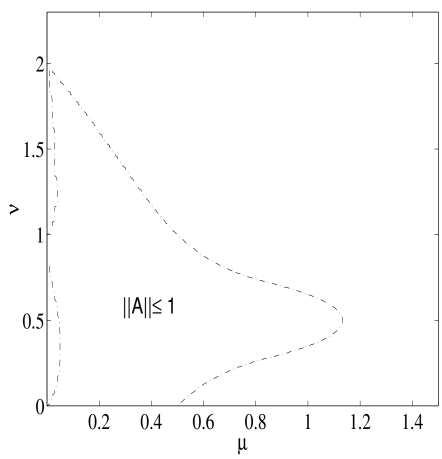



FIGURE 3. Stability region for Quickest scheme with the numerical boundary condition (12);

When $j \to \infty$ the Modified Quickest scheme is identical to the Quickest scheme. It is in the first points of the scheme that a considerable difference may occur and this seems to affect strongly the stability. This fact motivates us to consider the use of the Quickest scheme together with the numerical boundary condition (12) in situations where we are interested in long time behaviour of solutions.

We plot, in figure 3, the region $||A|| \leq 1$, where $A$ is the iteration matrix for the Quickest scheme associated with the numerical boundary conditions (12). We observe that we recover the von Neumann stability region lost when using different numerical boundary conditions with the Quickest scheme studied in Sousa and Sobey [10].

## 4. Test Problems

In this section we consider two test problems to compare the performances between the Modified Quickest scheme and the Quickest scheme. On the first test problem, the solution is dominated by the inflow boundary condition whereas on the second test problem is dominated by the initial condition. For our tests we assume $V = 0.1$ and $D = 0.001$.

**4.1. Test problem: initial condition $f(x) = 0$.** To measure accuracy of the different schemes we have considered a test problem with initial data

$$u(x,0) = 0, \quad x \geq 0, \quad u(0,t) = C_0.$$

The analytical solution is given by

$$u(x,t) \;=\; \frac{C_0}{2}\left[\mathrm{Erfc}\left(\frac{x - Vt}{2\sqrt{Dt}}\right) + \mathrm{e}^{\frac{Vx}{D}}\mathrm{Erfc}\left(\frac{x + Vt}{2\sqrt{Dt}}\right)\right]. \qquad (21)$$

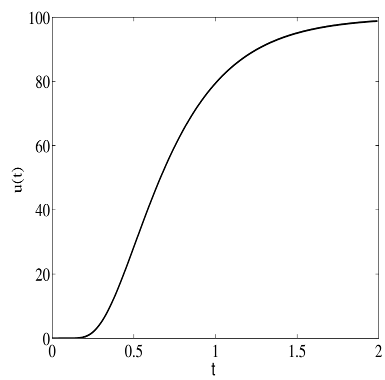



FIGURE 4. Exact solution (21) for $C_0 = 100$, $V = 0.1$ and $D = 0.001$ at $x = 0.075$

In the next examples we consider $C_0 = 100$. The exact solution at $x = 0.075$ is displayed in figure 4.

For this test problem we have $g(t) \neq 0$ and we can not consider the first integral of (9) zero. This integral is now given by

$$\frac{C_0}{\sqrt{\pi}} \int_0^{\Delta t} G^*(x,\tau)d\tau$$

where $G^*$ is defined by (5). We evaluate this integral using Gaussian quadrature formulas [11] and add it to the right side of (11).

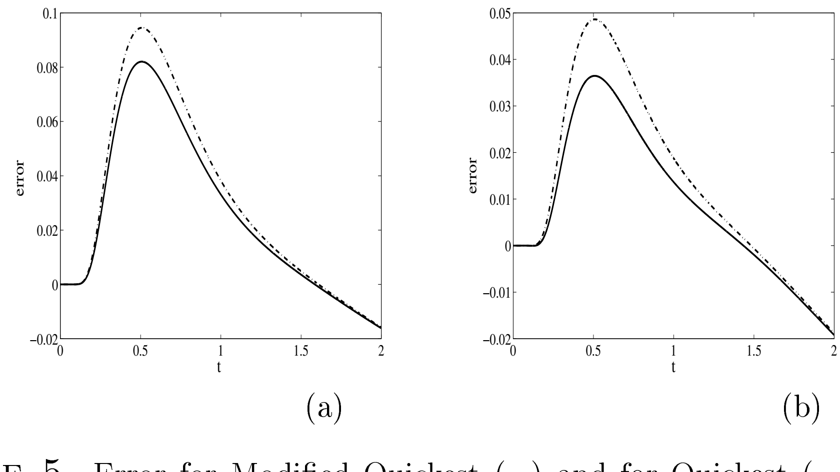
Our main concern, as mentioned before, is with the Modified Quickest scheme obtained using a cubic interpolation.

Let us consider the error function defined as

$$\text{error}(t) = U(t) - u(t),$$

where $U(t)$ is the approximate solution and $u(t)$ is the exact solution.

As we refine the space step we observe , see figure 5, that the error diminishes as expected for both schemes, the Modified Quickest scheme and the Quickest scheme. Although they share the same order of accuracy, the new scheme seems to have the advantage of a smaller error, next to the inflow boundary. In figure 5 we have used $\nu = 0.03$ and $\Delta t = 4.5\times10^{-4}$ $\Delta x = 0.0015$ and $\Delta t = 2.25 \times 10^{-4}$ $\Delta x = 7.5 \times 10^{-4}$ respectively. The part of the error after $t = 2$ tends to keep a constant error as $t$ increases, since both schemes show to be dissipative very near the inflow boundary.




(a) (b)

FIGURE 5. Error for Modified Quickest $(-)$ and for Quickest $(- \cdot -)$ at $x = 0.075$: (a) $\Delta x = 1.5 \times 10^{-3}$ (b) $\Delta x = 7.5 \times 10^{-4}$

As $x$ becomes bigger the two schemes become more similar. This is the reason to suggest the new scheme for problems where we are interested in small values of space, small values of time or as a mixed scheme as pointed out in section 3. In figure 6 we plot the errors when we take $x = 0.15$ and $x = 0.3$ for $\Delta x = 0.0015$. The error functions approach considerably as $x$ increases and additionally they tend to zero as time becomes larger.

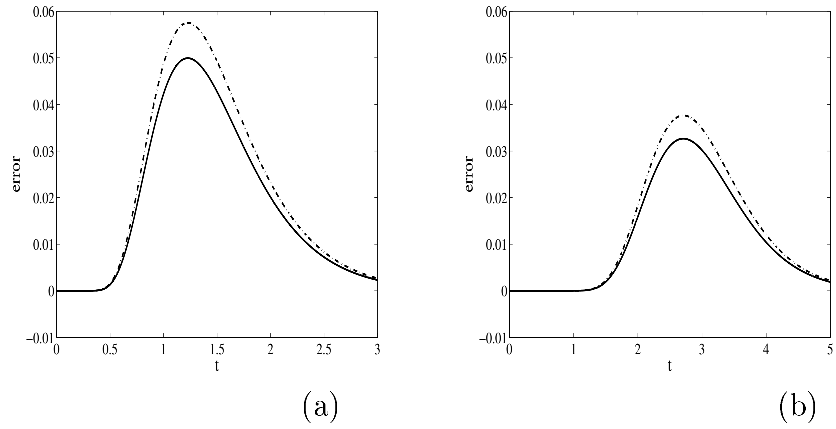



FIGURE 6. Error for Modified Quickest ($-$) and for Quickest ($- \cdot -$) at $\Delta x = 0.0015$: (a) $x = 0.15$; (b) $x = 0.3$

Furthermore, we can also observe that as we refine the space step and not the time step the fact that the stability region of the new scheme is larger allows us to get convergence in less time steps. For instance in figure 7 we show both schemes for $\Delta t = 0.006$ and $\Delta x = 0.003$, where the Quickest scheme with the numerical boundary condition

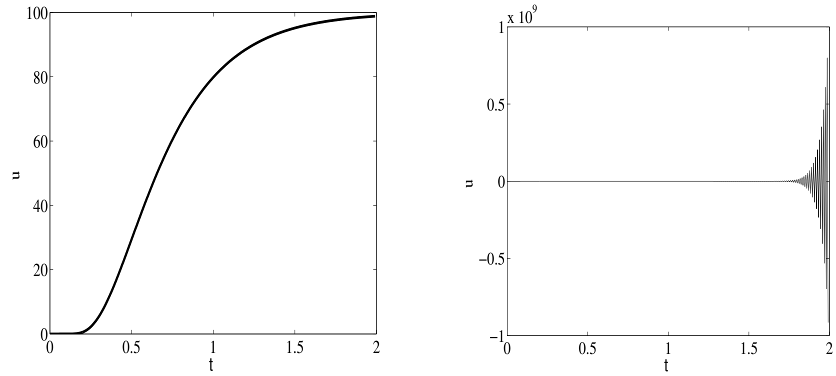
$$U_1^{n+1} = [1 - \nu\Delta_0 + (\frac{\nu^2}{2} + \mu)\delta^2 + \frac{\nu}{6}(1 - \nu^2 - 6\mu)\delta^2\Delta_+]U_1^n$$

diverges but the new scheme with the numerical boundary condition (12) converges.

This fact is quite important concerning that next to the inflow boundary to get a better accuracy, in most cases, it is convenient to refine the space step or the time step. If the stability region is larger when we refine the space step we have no need to refine the time step in order to be inside the stable region as happens in many practical situations.

We also believe that more significant is the inflow boundary more relevant this new scheme can be.

**4.2. Test problem: boundary condition $g(t) = 0$.** We can raise the question: Is it also worthwhile to use the Modified Quickest scheme if the inflow boundary is zero? Concerning stability it is always worthwhile.




(a) (b)

FIGURE 7. $\Delta t = 0.006$ and $\Delta x = 0.003$: (a) Modified Quickest scheme (b) Quickest scheme.

Let us then consider the initial condition and the inflow boundary condition as follows:

$$u(x,0) = \mathrm{e}^{-x^2}, \quad x \geq 0, \quad u(0,t) = 0.$$

Our reason for considering this initial profile is that it is straight forward to calculate an exact solution:

$$\begin{aligned} u(x,t) &= \frac{1}{2\sqrt{4Dt+1}} \left[ \mathrm{e}^{-\dfrac{(x-Vt)^2}{4Dt+1}} \mathrm{Erfc}\left( -\frac{(x-Vt)}{2\sqrt{Dt(4Dt+1)}} \right) \right. \\ & \qquad \left. - \mathrm{e}^{-\dfrac{(x+Vt)^2}{4Dt+1} + \dfrac{Vx}{D}} \mathrm{Erfc}\left( \frac{(x+Vt)}{2\sqrt{Dt(4Dt+1)}} \right) \right]. \end{aligned} \tag{22}$$

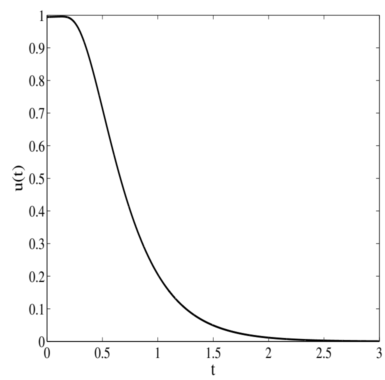
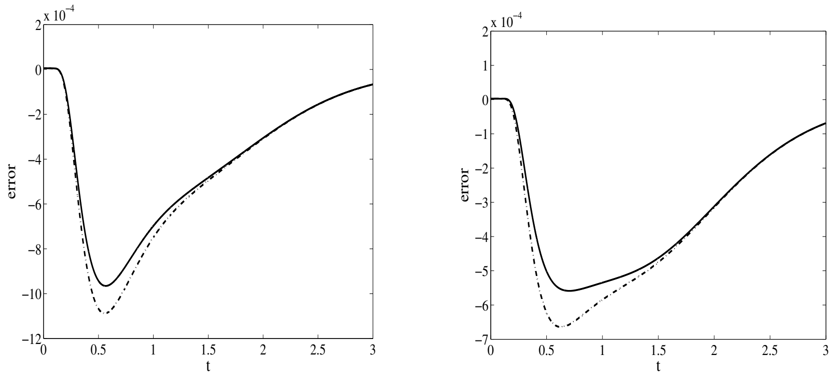
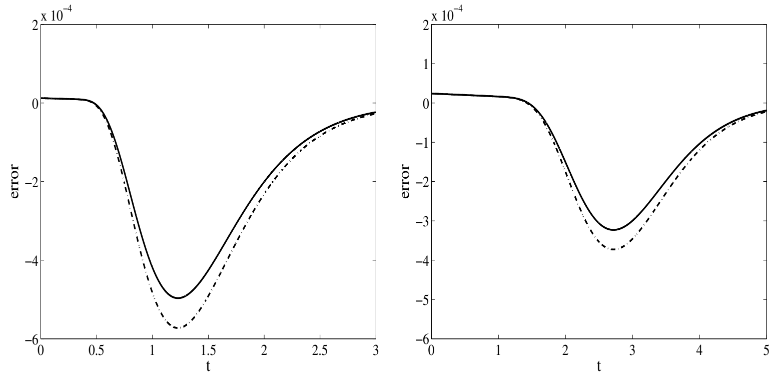



FIGURE 8. Exact solution (22) for $V = 0.1$ and $D = 0.001$ at $x = 0.075$




(a) (b)

FIGURE 9. Error for Modified Quickest ($-$) and for Quickest ($- \cdot -$): (a) $\Delta x = 1.5 \times 10^{-3}$ (b) $\Delta x = 7.5 \times 10^{-4}$




(a) (b)

FIGURE 10. Error for Modified Quickest ($-$) and for Quickest ($- \cdot -$) at $\Delta x = 0.0015$: (a) $x = 0.15$; (b) $x = 0.3$

In figure 8 we display the exact solution (22). We also plot similar results to figures 5-6 in figures 9-10. We have the same symptoms in both test problems. Furthermore the size of the error is influenced by the inflow boundary condition if we are next to it.

## 5. Concluding remarks

The new schemes are theoretically interesting because of the general framework in which they are derived. We have focused our attention on the Modified Quickest scheme which seems to provide a substantial advantage in terms of stability and an advantage in terms of the error in the part of the domain next to the inflow boundary.

The gain in stability seems strongly associated with the choice of the numerical boundary condition at the first point of the mesh. The Modified Quickest scheme for $j \to \infty$ becomes identical to the Quickest scheme and consequently for most practical purposes it is in the first discretisation points that we could expect some significant difference. The extension to the two-dimensional case is straightforward.

## References


[1] Morton, K.W and I.J Sobey (1993). Discretization of a convection-diffusion equation. *IMA Journal of Numerical Analysis* **13**, 141-160.

[2] Smith, G.D. (1985). *Numerical solution of partial differential equations: finite difference methods*, Oxford University Press: Oxford.

[3] Xu, H.Y., M.D. Matovic and A. Pollard (1997). Finite difference schemes for three-dimensional time-dependent convection-diffusion equation using full global discretization. *Journal of Computational Physics* **130**, 109-122.

[4] Lax, P.D. and B. Wendroff (1960). Systems of conservations laws. *Communications on Pure and Applied Mathematics* **13**, 217-237.

[5] Lax, P.D. and B. Wendroff (1964). Difference schemes for hyperbolic equations with high order of accuracy. *Communications on Pure and Applied Mathematics* **17**, 381-398.

[6] Leonard, B.P. (1979). A stable and accurate convective modelling procedure based on quadratic upstream interpolation. *Computer Methods in Applied Mechanics and Engineering* **19**, 59-98.

[7] Richtmyer, R.D. and K.W. Morton (1967). *Difference methods for initial-value problems.* Wiley-Interscience: New York.

[8] Warming, F.F. and B.J. Hyett (1974). The modified equation approach to the stability and accuracy analysis of finite difference methods. *Journal of Computational Physics* **14**, 159-179.

[9] Powell, M.J.D. (1981). *Approximation theory and method*, Cambridge.

[10] Sousa, E. and Sobey, I.J. (2002) On the influence of numerical boundary conditions *Applied Numerical Mathematics*, **41**, 325-344.

[11] Davis, P.J. and P. Rabinowitz (1975). *Methods of Numerical Integration* Academic Press: New York.


# Appendix A: Error analysis for the new schemes

Following Morton and Sobey [1], the truncation error is given by

$$T^n = \frac{1}{\Delta t} RE(\Delta t)(u^n - I_p R u^n),$$

where $E(\Delta t)$ is the evolution operator $u(\cdot, t + \Delta t) = E(\Delta t)u(\cdot, t)$, for our problem in the semi line $x \geq 0$, $R$ is the restriction operator onto the nodes and $I_p$ is the local approximation based on nodal values.

Let us define the interpolation error $Lu^n = u^n - I_p u^n$ and define for $a \geq 0$, the integrals of the form

$$E_m(a;\mu) = \int_0^\infty \xi^m \mathrm{e}^{-(\xi+a)^2/4\mu} \frac{d\xi}{2\sqrt{\pi\mu}}. \tag{23}$$

In what follows we denote $s = (x - x_j)/\Delta x$ and we omit the subscript $n$, referring only to $u(x)$ and its evolution over one time step.

*Quadratic interpolation*

We calculate the error at the point $x_j = j\Delta x$. We have, for $j \geq 1$

$$Lu = u(x) - \frac{1}{2}[-s + s^2]u(x_{j-1}) - [1 - s^2]u(x_j) - \frac{1}{2}[s^2 + s]u(x_{j+1}).$$

Then using the Peano kernel theorem we can write

$$Lu = \int_0^\infty K(x,p)u^{(3)}(p)dp,$$

where $K(x,p) = (1/2)L_x[(x-p)_+^2]$, $L_x$ refers to $L$ acting in $x$, and $(x-p)_+^2 = (x-p)^2$ if $x - p \geq 0$, and zero otherwise. We calculate the Peano kernel function $K(s\Delta x + j\Delta x, \xi\Delta x)$,

$$K = \frac{1}{2}\Delta x^2 \begin{cases} (s+j-\xi)_+^2 - (j-1-\xi)^2(-s/2+s^2/2) & \\ -(j-\xi)^2(1-s^2) - (j+1-\xi)^2(s^2/2+s/2), & 0 < \xi < j-1, \\ (s+j-\xi)_+^2 - (j-\xi)^2(1-s^2) & \\ -(j+1-\xi)^2(s^2/2+s/2), & j-1 < \xi < j, \\ (s+j-\xi)_+^2 - (j+1-\xi)^2(s^2/2+s/2), & j < \xi < j+1, \\ (s+j-\xi)_+^2, & \xi > j+1. \end{cases}$$

The local error is given by

$$\Delta t T = RE_x \int_0^\infty K(x,p)u^{(3)}(p)dp = \int_0^\infty RE_x K(x,p)u^{(3)}(p)dp$$

where we use the notation $E_x$ to describe $E(\Delta t)$ acting on $x$. After some manipulation to calculate $RE_xK(x,p)u^{(3)}(p)$ we have,

$$RE_xK = \frac{1}{2}\Delta x^2 \begin{cases} E_2(\xi+\nu-j;\mu) - \mathrm{e}^{\nu j/\mu}E_2(\xi+\nu+j;\mu) & \\ -(j-1-\xi)^2(-f_1+f_2)/2 & \\ -(j-\xi)^2(f_3-f_2) - (j+1-\xi)^2(f_2+f_1)/2, & 0<\xi<j-1, \\ E_2(\xi+\nu-j;\mu) - \mathrm{e}^{\nu j/\mu}E_2(\xi+\nu+j;\mu) & \\ -(j-\xi)^2(f_3-f_2) - (j+1-\xi)^2(f_2+f_1)/2, & j-1<\xi<j, \\ E_2(\xi+\nu-j;\mu) - \mathrm{e}^{\nu j/\mu}E_2(\xi+\nu+j;\mu) & \\ -(j+1-\xi)^2(f_2+f_1)/2, & j<\xi<j+1, \\ E_2(\xi+\nu-j;\mu) - \mathrm{e}^{\nu j/\mu}E_2(\xi+\nu+j;\mu), & \xi>j+1, \end{cases}$$

where,

$$\begin{aligned} f_1 &= -\nu a(j) + b(j) & \qquad f_2 &= (\nu^2+2\mu)a(j) + 2c(j) \\ f_3 &= a(j) & f_4 &= -(\nu^3+6\mu\nu)a(j) + e(j). \end{aligned}$$

The exact error at $x_j$ is given by

$$\Delta t T_{|_{x_j}} = \Delta x^3 [\int_0^{+\infty} K_1(\xi,\nu,\mu)u^{(3)}(\xi\Delta x)d\xi + \int_0^{j+1} K_2(\xi,\nu,\mu)u^{(3)}(\xi\Delta x)d\xi].$$

where we have introduced two functions

$$K_1(\xi,\nu,\mu) = \frac{1}{2}[E_2(-j+\nu+\xi;\mu) - e^{\nu j/\mu}E_2(j+\nu+\xi;\mu)];$$

$$K_2(\xi,\nu,\mu) = \begin{cases} -(f_2-f_1)(j-1-\xi)^2/4 & \\ +(f_2-f_3)(j-\xi)^2/2 & \\ -(f_2+f_1)(j+1-\xi)^2/4 & 0\le\xi<j-1, \\ (f_2-f_3)(j-\xi)^2/2 & \\ -(f_2+f_1)(j+1-\xi)^2/4 & j-1\le\xi<j, \\ -(f_2+f_1)(j+1-\xi-j)^2/4 & j\le\xi\le j+1. \end{cases}$$

Although the exact error expression appears complicated, it can be used to examine the detailed structure of the error and to obtain overall bounds for the local error.

Considering first the structure of the error, if we assume $u \in C^\infty(\mathbb{R})$ and use a Taylor series expansion for $u^{(3)}$ around $x_j$, then after some algebra,

$$\Delta t T_{|_{x_j}} = \frac{1}{6}\Delta x^3 u_{xxx} g_j^2(\nu,\mu) + O(\Delta x^4 u_{x^4}),$$

where

$$g_j^2(\nu,\mu) = \nu(1-\nu^2-6\mu)a(j) - b(j) + e(j).$$

Secondly, we can obtain an overall bound for the local error since

$$|\Delta t T_{|_{x_j}}| \leq \Delta x^3 |u|_{3,\infty}[\int_0^{+\infty} |K_1(\xi,\nu,\mu)|d\xi + \int_0^{j+1} |K_2(\xi,\nu,\mu)|d\xi].$$

*Cubic interpolation*

We have

$$\begin{aligned} Lu &= u(x) - \frac{1}{6}[s-s^3]u(x_{j-2}) - \frac{1}{2}[-2s+s^2+s^3]u(x_{j-1}) \\ &\quad -\frac{1}{2}[2+s-2s^2-s^3]u(x_j) - \frac{1}{6}[2s+3s^2+s^3]u(x_{j+1}). \end{aligned}$$

The Peano kernel function $K(x,p)$ for $x = s\Delta x + j\Delta x$ and $p = \Delta x \xi$ is given by

$$\begin{aligned} K &= \frac{1}{6}\Delta x^3 \Bigg[ (s+j-\xi)_+^3 - \frac{1}{6}(j-2-\xi)_+^3(s-s^3) \\ &\quad -\frac{1}{2}(j-1-\xi)_+^3(-2s+s^2+s^3) - \frac{1}{2}(j-\xi)_+^3(2+s-2s^2-s^3) \\ &\quad -\frac{1}{6}(j+1-\xi)_+^3(2s+3s^2+s^3) \Bigg]. \end{aligned}$$

Summarising the calculations in this case, the exact error for $x_j, j \geq 2$ is given by

$$\Delta t T_{|_{x_j}} = \frac{\Delta x^4}{6}[\int_0^{+\infty} W_1^2(\xi,\nu,\mu)u^{(4)}(\xi\Delta x)d\xi + \int_0^{j+1} W_2^2(\xi,\nu,\mu)u^{(4)}(\xi\Delta x)d\xi].$$

where the functions

$$W_1^2(\xi,\nu,\mu) = [E_3(-j+\nu+\xi;\mu) - e^{\nu j/\mu}E_3(j+\nu+\xi;\mu)];$$

$$W_2^2(\xi,\nu,\mu) = \begin{cases} q_1(\xi-j+2)^3/6 & \\ +q_2(\xi-j+1)^3/2 & \\ +q_3(\xi-j)^3/2 + q_4(j)(\xi-j-1)^3/6, & 0 \le \xi < j-2, \\ q_2(\xi-j+1)^3/2 + q_3(j)(\xi-j)^3/2 & \\ +q_4(\xi-j-1)^3/6, & j-2 \le \xi < j-1, \\ q_3(\xi-j)^3/2 + q_4(j)(\xi-j-1)^3/6, & j-1 \le \xi \le j, \\ q_4(\xi-j-1)^3/6, & j \le \xi \le j+1. \end{cases}$$

where

$$\begin{aligned} q_1 &= f_1 - f_4, \\ q_2 &= f_4 + f_2 - 2f_1, \\ q_3 &= 2f_3 - f_4 + f_1 - 2f_2, \\ q_4 &= 2f_1 + 3f_2 + f_4. \end{aligned}$$

If we assume $u \in C^\infty(\mathbb{R})$ and use a Taylor expansion for $u^{(4)}$ then the truncation error is given by the cumbersome expression

$$\Delta t T_{|_{x_j}} = \frac{1}{24}\Delta x^4 u_{xxxx} g_j^3(\nu,\mu) + \cdots,$$

with

$$\begin{aligned} g_j^3(\nu,\mu) &= (12\mu^2 - 2\mu - 12\mu\nu(1-\nu) + \nu(1-\nu^2)(2-\nu))a(j) \\ &\quad -2b(j)(12\mu\nu + 2\nu^3 + 2j\nu + 1 + 2j^3) \\ &\quad +2c(j)(1+6j^2) + 2(1-2j)e(j) + 2Z(j)(3\nu^2 + j^2 + 10\mu). \end{aligned}$$

Ercília Sousa
Departamento de Matemática, Universidade de Coimbra, Portugal