Direct numerical methods for solving a class of third-order partial differential equations

M. Mechee a,b,⇑, F. Ismail c, Z.M. Hussain a,d, Z. Siri b

Faculty of Mathematics and Computer Sciences, University of Kufa, Najaf, Iraq
Institute of Mathematical Science, University of Malaya, 50603 Kuala Lumpur, Malaysia
Department of Mathematics and Institute for Mathematical Research, Universiti Putra Malaysia, 43400 UPM Serdang, Selangor, Malaysia
School of Engineering, Edith Cowan University, Joondalup, WA 6027, Australia

article info

Keywords:
RKD method
Method of lines
Third-order
PDE
System of ODEs

abstract

In this paper, three types of third-order partial differential equations (PDEs) are classified to be third-order PDE of type I, II and III. These classes of third-order PDEs usually occur in many subfields of physics and engineering, for example, PDE of type I occurs in the impulsive motion of a flat plate. An efficient numerical method is proposed for PDE of type I. The PDE of type I is converted to a system of third-order ordinary differential equations (ODEs) using the method of lines. The system of ODEs is then solved using direct Runge–Kutta which we derived purposely for solving special third-order ODEs of the form $y''' = f(x,y)$. Simulation results showed that the proposed RKD-based method is more accurate than the existing finite difference method.

1. Introduction

Many applications of differential equations (DEs), particularly PDEs of different orders can be found in the mathematical modeling of real life problems [1–9]. Third-order PDEs occur in the impulsive motion of a flat-plate Korteweg-de Vries layer equation with pressure and hydrodynamic boundary layer equations.

In recent years, numerical methods have been applied to wide classes of PDEs problems in many fields of mathematics, physics, and engineering. For example, the method of lines is one such an efficient routine to solve second-order PDEs. This method has been used in [5,10] for solving second-order elliptic PDEs while Shakeri and Dehghan [11] solved wave equation using method of lines.

In this paper, we have classified third-order PDEs to be of type I, II and III. These classes of third-order PDEs usually occur in many fields of physics and engineering, for instance, the impulsive motion of a flat plate. [12], solved a third-order PDE of type I as an application arising from the impulsive motion of a flat plate for various boundary conditions while [13], solved third-order dispersive PDE in one- and higher-dimensional spaces. Moreover, research performed by Furzeland [14], Northrop [5], Schiesser [15] and Subramanian [10] solved PDEs of type II and III.

1.1. Definitions

The general third-order PDE in two variables can be defined as the following:
\[
\frac{\partial}{\partial t} u(x,t) - \frac{\partial}{\partial x} \left( \frac{\partial^2 u(x,t)}{\partial x^2} \right) + \frac{\partial^3 u(x,t)}{\partial t \partial x^2} - \frac{\partial^3 u(x,t)}{\partial x^3} = 0,
\]

and the general linear third-order PDE in \( n \) variables is given by:
\[
\sum_{i=1}^{n} \int (x_1, \ldots, x_n) \frac{\partial u(x_1, \ldots, x_n)}{\partial x_i} + \sum_{i_1 \neq i_2 \neq i_3} \frac{\partial^3 u(x_1, \ldots, x_n)}{\partial x_{i_1} \partial x_{i_2} \partial x_{i_3}} = f(x_1, \ldots, x_n).
\]

The general quasi linear third-order PDE in \( n \) variables is given as:
\[
\sum_{i=1}^{n} \int (x_1, \ldots, x_n, u) \frac{\partial u(x_1, \ldots, x_n, u)}{\partial x_i} + \sum_{i_1 \neq i_2 \neq i_3} \frac{\partial^3 u(x_1, \ldots, x_n, u)}{\partial x_{i_1} \partial x_{i_2} \partial x_{i_3}} = f(x_1, \ldots, x_n, u).
\]

Here, we give some new definitions for a class of quasi linear third-order PDE.

**Definition 1.** Quasi linear PDE of type I

Define the quasi linear third-order PDE of type I which has \( n \) independent variables as follows:
\[
\left( \frac{\partial}{\partial t} - \frac{\partial}{\partial x} \left( \frac{\partial^2 u(x_1, \ldots, x_n, u)}{\partial x^2} \right) + \frac{\partial^3 u(x_1, \ldots, x_n, u)}{\partial t \partial x^2} - \frac{\partial^3 u(x_1, \ldots, x_n, u)}{\partial x^3} \right) = f(x_1, \ldots, x_n, u),
\]

for \( i_1 = 1, 2, \ldots, n \), then the general form of third-order PDE of type I in two variables is given as
\[
u_{xx} = f(x, t, u, u_t, u_{tt}),
\]

or
\[
u_{tt} = f(x, t, u, u_x, u_{xx}, u_{xxx}).
\]

**Definition 2.** Quasi linear PDE of type II

Define the quasi linear third-order PDE of type II which has \( n \) independent variables as:
\[
\left( \frac{\partial}{\partial t} - \frac{\partial}{\partial x} \left( \frac{\partial^2 u(x_1, \ldots, x_n, u)}{\partial x^2} \right) + \frac{\partial^3 u(x_1, \ldots, x_n, u)}{\partial t \partial x^2} - \frac{\partial^3 u(x_1, \ldots, x_n, u)}{\partial x^3} \right) = f(x_1, \ldots, x_n, u),
\]

for \( i_1 = 1, 2, \ldots, n \), then the general form of third-order PDE of type II in two variables is given as
\[
u_{xx} = f(x, t, u, u_t, u_{tt}),
\]

or
\[
u_{tt} = f(x, t, u, u_x, u_{xx}, u_{xxx}).
\]

**Definition 3.** Quasi linear PDE of type III

Define the quasi linear third-order PDE of type III which has \( n \) independent variables as:
\[
\left( \frac{\partial}{\partial t} - \frac{\partial}{\partial x} \left( \frac{\partial^2 u(x_1, \ldots, x_n, u)}{\partial x^2} \right) + \frac{\partial^3 u(x_1, \ldots, x_n, u)}{\partial t \partial x^2} - \frac{\partial^3 u(x_1, \ldots, x_n, u)}{\partial x^3} \right) = f(x_1, \ldots, x_n, u),
\]
for \(i_j = 1, 2, \ldots, n\), then the general form of third-order PDE of type III in two variables is given as

\[
\begin{align*}
u_x &= f(x, t, u, u_t, u_{tt}), \\
u_t &= f(x, t, u, u_x, u_{xx}, u_{xxx}).
\end{align*}
\]

In recent years, numerical methods have been applied to wide classes of PDEs problems in many fields of mathematics, physics and engineering. Such numerical methods provide numerical solutions for some types of PDEs.

The third-order PDE of type I is converted to a system of third-order ODEs by using the method of lines. This system of ODEs is then solved by using direct RKD methods, which we derived purposely for solving special third-order ODEs of the form \(y''' = f(x, y)\) (see [16,17]). In this work, numerical results based on the RKD method are compared with the exact solutions as well as finite difference method. Simulation results show that the method is highly accurate.

2. Analysis of the proposed numerical method

There are special third-order ODEs with no explicit dependence on the first derivative \(y'(x)\) and the second derivative \(y''(x)\). Such ODEs are frequently found in many physical problems, such as electromagnetic waves, thin film flow and gravity. They can be written in the following form

\[
y'''(x) = f(x, y(x)); \quad x \geq x_0,
\]

with initial conditions,

\[
y(x_0) = x^0, \ y'(x_0) = x^1 \text{ and } y''(x_0) = x^2.
\]

where,

\[
f: \mathbb{R} \times \mathbb{R}^N \rightarrow \mathbb{R}^N,
\]

and

\[
\begin{align*}
y(x) &= [y_1(x), y_2(x), \ldots, y_N(x)], \\
f(x, y) &= [f_1(x, y), f_2(x, y), \ldots, f_N(x, y)], \\
x^0 &= [x^0_1, x^0_2, \ldots, x^0_N], \\
x^1 &= [x^1_1, x^1_2, \ldots, x^1_N], \\
x^2 &= [x^2_1, x^2_2, \ldots, x^2_N].
\end{align*}
\]

When the ODE (10) is in \(n\)-dimensional space, then we can simplify to

\[
z''(x) = g(z(x)),
\]

using the following assumption,

\[
z(x) = \begin{pmatrix}
y_1(x) \\
y_2(x) \\
y_3(x) \\
\vdots \\
y_N(x) \\
x
\end{pmatrix}, \quad g(z) = \begin{pmatrix}
f_1(z_1, z_2, \ldots, z_N, z_{N+1}) \\
f_2(z_1, z_2, \ldots, z_N, z_{N+1}) \\
f_3(z_1, z_2, \ldots, z_N, z_{N+1}) \\
\vdots \\
f_N(z_1, z_2, \ldots, z_N, z_{N+1}) \\
0
\end{pmatrix},
\]

with the initial conditions

\[
z(x_0) = \bar{z}^0, \ z'(x_0) = \bar{z}^1, \ z''(x_0) = \bar{z}^2,
\]

where

\[
\begin{align*}
\bar{z}^0 &= [x^0_1, x^0_2, \ldots, x^0_N, x_0], \\
\bar{z}^1 &= [x^1_1, x^1_2, \ldots, x^1_N, 1], \\
\bar{z}^2 &= [x^2_1, x^2_2, \ldots, x^2_N, 0].
\end{align*}
\]

Most researchers, scientists and engineers used to solve ODE (10) or (11) by converting this third-order ODE to a system of first-order ODEs three times the dimensions [18]. Some researchers can solve this equation by using multistep methods. However, it would be more efficient if the ODE (10) or (11) can be solved using direct numerical methods (see [16,17]).
2.1. Direct numerical RKD method

The general form of RKD method with s-stage for solving ODE (10) or (11) can be written as
\[
y_{n+1} = y_n + h y'_n + \frac{h^2}{2} y''_n + h^3 \sum_{i=1}^{s} b_i k_i, \tag{12}
\]
\[
y'_{n+1} = y'_n + h y''_n + h^2 \sum_{i=1}^{s} b_i k_i, \tag{13}
\]
\[
y''_{n+1} = y''_n + h \sum_{i=1}^{s} b_i k_i, \tag{14}
\]
where,
\[
k_i = f(x_n, y_n), \tag{15}
\]
\[
k_i = f \left( x_n + c_i h, y_n + h c_i y'_n + \frac{h^2}{2} c_i^2 y''_n + h^3 \sum_{j=1}^{s} d_{ij} k_j \right), \tag{16}
\]
for \( i = 2, 3, \ldots, s \).

The parameters of the RKD method are \( c_i, a_i, b_i, b_i', b_i'' \) for \( i = 1, 2, \ldots, s \) and \( j = 1, 2, \ldots, s \); and they are real. The RKD method can be expressed in Butcher notation using the table of coefficients as in Table 1 (see [19]).

In [16,17], direct integrators (for solving special third-order ODEs) are derived for Runge-Kutta type with constant step-size of orders three, four, and six. While in [20], variable step-size direct integrators are derived for Runge-Kutta type of orders 6(5), 5(4) and 4(3). Accordingly, we can use RKD methods of orders three, four, five and six as derived in [16,17,21,22] to solve (10) or (11).

The Butcher tableaus of the RKD methods of orders three, four, five and six are shown in Tables 2–5, respectively.

2.2. The proposed numerical method

We present a numerical method for solving third-order PDEs of type I by combining the method of lines with RKD method.

First, consider the third-order PDE of type I as follows,
\[
u_{ttt} = f(x, t, u, u_x, u_{xx}), \quad a \leq x \leq b, \quad 0 < t \leq T, \tag{17}
\]
with initial conditions
\[
u(x, 0) = f_1(x), u_t(x, 0) = f_2(x), u_{xx}(x, 0) = f_3(x), \tag{18}
\]
and the boundary conditions,
\[
u(a, t) = g_1(t), u(b, t) = g_2(t). \tag{19}
\]
Let the interval of the numerical solution in the directions of x and t be \([a, b]\) and \([0, T]\), respectively, with \( h = \frac{b-a}{m} \) and \( k = \frac{T}{m} \). Here, \( n \) is the number of points in the direction of x on the interval \([a, b]\), while \( m \) is the number of points in the direction of t over the interval \([0, T]\). Also, \( x_i = a + ih \), and \( t_j = jk \), for \( i = 1, 2, \ldots, n - 1 \) and \( j = 1, 2, \ldots, m \). We can combine the method of lines (MOL) and RKD method to solve the problem (17) with the initial conditions (18) and the boundary conditions (19) using the following steps:

1. do steps 2–6 while \( 1 \leq j \leq m \)
2. fix \( x = x_i \) at the point \((x, t)\) of the PDE in (17) converting to the following equation
\[
u_{x}^{(m)}(t) = f \left( x, t, u(x, t), \frac{\partial u(x, t)}{\partial x}, \frac{\partial^2 u(x, t)}{\partial x^2}, \frac{\partial^3 u(x, t)}{\partial x^3} \right) \bigg|_{x=x_i}, \tag{20}
\]

\[
| \begin{array}{c|c}
\text{c} \\
\hline
b^T \\
\hline
b'^T \\
\hline
b''^T \\
\end{array}
\]

\[\text{Table 1}\]

The Butcher tableau of the RKD method.
Table 2
The Butcher tableau of the RKD3 method of third-order.

<table>
<thead>
<tr>
<th>0</th>
<th>2/3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1/8</td>
<td>1/24</td>
</tr>
<tr>
<td>1/4</td>
<td>1/4</td>
</tr>
<tr>
<td>1/4</td>
<td>1/4</td>
</tr>
</tbody>
</table>

Table 3
The Butcher tableau of the RKD4 method of fourth-order.

<table>
<thead>
<tr>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1/2</td>
</tr>
<tr>
<td>1/6</td>
</tr>
<tr>
<td>1/3</td>
</tr>
<tr>
<td>1/3</td>
</tr>
<tr>
<td>1/12</td>
</tr>
<tr>
<td>1/12</td>
</tr>
<tr>
<td>0</td>
</tr>
<tr>
<td>1/6</td>
</tr>
<tr>
<td>3/3</td>
</tr>
<tr>
<td>1/6</td>
</tr>
</tbody>
</table>

Table 4
The Butcher tableau of the RKD5 method of fifth-order.

<table>
<thead>
<tr>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>2/5 - \sqrt{5} / 15</td>
<td></td>
</tr>
<tr>
<td>\sqrt{5} / 5 + 1/50</td>
<td></td>
</tr>
<tr>
<td>11/200 + 116/1000</td>
<td></td>
</tr>
<tr>
<td>43/200 + 121/1250</td>
<td></td>
</tr>
<tr>
<td>1/13</td>
<td></td>
</tr>
<tr>
<td>1/18 + \sqrt{5} / 48</td>
<td></td>
</tr>
<tr>
<td>1/18 - \sqrt{5} / 48</td>
<td></td>
</tr>
<tr>
<td>7/36 + \sqrt{5} / 18</td>
<td></td>
</tr>
<tr>
<td>7/36 - \sqrt{5} / 18</td>
<td></td>
</tr>
<tr>
<td>4/9 + \sqrt{5} / 36</td>
<td></td>
</tr>
<tr>
<td>4/9 - \sqrt{5} / 36</td>
<td></td>
</tr>
</tbody>
</table>

Table 5
The Butcher tableau of the RKD6 method of sixth-order.

<table>
<thead>
<tr>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1/2 \sqrt{5} / 18</td>
<td></td>
</tr>
<tr>
<td>7/120 + \sqrt{5} / 200</td>
<td></td>
</tr>
<tr>
<td>1/2</td>
<td></td>
</tr>
<tr>
<td>1/18</td>
<td></td>
</tr>
<tr>
<td>1/18 + \sqrt{5} / 36</td>
<td></td>
</tr>
<tr>
<td>1/18 - \sqrt{5} / 36</td>
<td></td>
</tr>
<tr>
<td>1/2 \sqrt{5} / 18</td>
<td></td>
</tr>
<tr>
<td>1/18 + \sqrt{5} / 36</td>
<td></td>
</tr>
<tr>
<td>1/18 - \sqrt{5} / 36</td>
<td></td>
</tr>
<tr>
<td>2/36 + \sqrt{5} / 36</td>
<td></td>
</tr>
<tr>
<td>2/36 - \sqrt{5} / 36</td>
<td></td>
</tr>
<tr>
<td>5/18</td>
<td></td>
</tr>
<tr>
<td>5/18</td>
<td></td>
</tr>
<tr>
<td>5/18</td>
<td></td>
</tr>
<tr>
<td>5/18</td>
<td></td>
</tr>
<tr>
<td>5/18</td>
<td></td>
</tr>
<tr>
<td>5/18</td>
<td></td>
</tr>
</tbody>
</table>
where,
\[ u'''_i(t) = \frac{d^3 u(x_i, t)}{dt^3}, \]
for \( i = 1, 2, \ldots, n - 1 \).

3. Substitute finite difference formulas of the orders one, two and three into the derivatives of the right side of ODE (20), then we obtain a system of ODEs of third-order

\[ u'''_i(t) = f(x_i, t, u_{i-2}(t), u_{i-1}(t), u_i(t), u_{i+1}(t), u_{i+2}(t)) \]

for \( i = 1, 2, \ldots, n - 1 \).

4. when \( j = 1 \) then the initial conditions are,
\[ u_i(0) = f_1(x_i), \]
\[ u'_i(0) = f_2(x_i), \]
\[ u''_i(0) = f_3(x_i), \] (22)
for \( 2 =< j =< m \), the initial conditions are,
\[ u_i(t_{j-1}) = u(x_i, t_{j-1}), \]
\[ u'_i(t_{j-1}) = \frac{du(x_i, t_{j-1})}{dx}_{|x=x_i}, \]
\[ u''_i(t_{j-1}) = \frac{d^2 u(x_i, t_{j-1})}{dx^2}_{|x=x_i}, \] (23)

5. put the boundary conditions,
\[ u_{a,j} = u(a, t_j) = g_1(t_j), u_{n,j} = u(b, t_j) = g_2(t_j), \] (24)

6. solve the system of third-order ODEs (21) at \( t = t_j \) with initial conditions (22) or (23) and boundary conditions (24) using the RKD method.

This algorithm is for solving third-order PDE (17) of type I with initial conditions (18) and boundary conditions (19) in the region shown in Fig. 1.

2.3. The finite difference method (FDM)

The finite difference method (FDM) is a classical method for solving ordinary and partial differential equations. The forward, backward and central finite difference (FD) formulas were used in solving ODEs and PDEs (see [18,23]). We will use FDM for solving this problem in our study to compare with numerical results of the proposed method. The PDE (17) at the point \((x, t) = (x_i, t_j)\) for \( i = 1, 2, \ldots, n - 1 \) and \( j = 1, 2, \ldots, m \) will be converted into the following equation

\[ \left[ \frac{\partial^3 u(x, t)}{\partial t^3} \right]_{(x,t)=(x_i,t_j)} = f \left( x, t, u(x, t), \frac{\partial u(x, t)}{\partial x}, \frac{\partial^2 u(x, t)}{\partial x^2}, \frac{\partial^3 u(x, t)}{\partial x^3} \right)_{(x,t)=(x_i,t_j)}. \]
By using the following central finite difference formulas of the orders one, two and three as follows

\[
\frac{\partial u(x,t)}{\partial x}(x_i,t_j) = \frac{u_{i+1,j} - u_{i-1,j}}{h},
\]

\[
\frac{\partial^2 u(x,t)}{\partial x^2}(x_i,t_j) = \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2},
\]

\[
\frac{\partial^3 u(x,t)}{\partial x^3}(x_i,t_j) = \frac{u_{i+2,j} - 2u_{i+1,j} + 2u_{i-1,j} - u_{i-2,j}}{2h^3},
\]

\[
\frac{\partial^3 u(x,t)}{\partial x^3}(x_i,t_j) = \frac{u_{i,j+2} - 2u_{i,j+1} + 2u_{i,j} - u_{i,j-2}}{2h^3},
\]

**Table 6**
Comparison between Exact and Numerical Solutions for RKD5 for Problem 1, \(a = 0, b = 1\).

<table>
<thead>
<tr>
<th>Times ((t_j))</th>
<th>(x_i)</th>
<th>Exact solution</th>
<th>Numerical solution</th>
<th>Absolute error</th>
</tr>
</thead>
<tbody>
<tr>
<td>(10^{-6})</td>
<td>0.1</td>
<td>9.950031702750127e-001</td>
<td>9.950021752716852e-001</td>
<td>9.95003275085701e-007</td>
</tr>
<tr>
<td>(50(10^{-6}))</td>
<td>0.3</td>
<td>9.552896787769188e-001</td>
<td>9.55242870730476e-001</td>
<td>4.680803987122673e-005</td>
</tr>
<tr>
<td>(100(10^{-6}))</td>
<td>0.5</td>
<td>8.774948080140238e-001</td>
<td>8.77407062984759e-001</td>
<td>8.774508554787452e-005</td>
</tr>
<tr>
<td>(200(10^{-6}))</td>
<td>0.9</td>
<td>6.18092094990918e-001</td>
<td>6.213626167295144e-001</td>
<td>3.270521738522603e-003</td>
</tr>
</tbody>
</table>

**Fig. 2.** Comparison between numerical and exact solutions for Problems 1a, and 1b (sub-figures a, b, respectively).
### Table 7
Comparison between Exact and Numerical Solutions for RKD6 for Problem 1, $a = -\frac{2}{3}$, $b = \frac{2}{3}$.

<table>
<thead>
<tr>
<th>Times($t_i$)</th>
<th>$x_i$</th>
<th>Exact solution</th>
<th>Numerical solution</th>
<th>Absolute error</th>
</tr>
</thead>
<tbody>
<tr>
<td>$10^{-6}$</td>
<td>$\frac{2}{3}\pi$</td>
<td>3.090166853581076e-001</td>
<td>3.090163763415768e-001</td>
<td>3.09016308083925e-007</td>
</tr>
<tr>
<td>50($10^{-6}$)</td>
<td>$\frac{2}{3}\pi$</td>
<td>8.089773535125675e-001</td>
<td>8.08937145942713e-001</td>
<td>3.96389182961698e-005</td>
</tr>
<tr>
<td>100($10^{-6}$)</td>
<td>$\frac{2}{3}\pi$</td>
<td>8.089369056498938e-001</td>
<td>8.088568248673658e-001</td>
<td>8.0080782508026e-005</td>
</tr>
<tr>
<td>200($10^{-6}$)</td>
<td>$\frac{2}{3}\pi$</td>
<td>3.089542924985687e-001</td>
<td>3.08840300827036e-001</td>
<td>6.02624158650977e-005</td>
</tr>
</tbody>
</table>

### Table 8
Comparison between Exact and Numerical Solutions for RKD4 for Problem 2, $a = 0$, $b = 1$.

<table>
<thead>
<tr>
<th>Times($t_i$)</th>
<th>$x_i$</th>
<th>Exact solution</th>
<th>Numerical solution</th>
<th>Absolute error</th>
</tr>
</thead>
<tbody>
<tr>
<td>$10^{-6}$</td>
<td>0.1</td>
<td>9.048365131989939e-001</td>
<td>9.048365131989939e-001</td>
<td>0</td>
</tr>
<tr>
<td>50($10^{-6}$)</td>
<td>0.3</td>
<td>7.407634021617126e-001</td>
<td>7.407634021616976e-001</td>
<td>1.49880108324396e-014</td>
</tr>
<tr>
<td>100($10^{-6}$)</td>
<td>0.7</td>
<td>4.964684931481651e-001</td>
<td>4.964684931479401e-001</td>
<td>2.24986695402880e-013</td>
</tr>
<tr>
<td>1000($10^{-6}$)</td>
<td>0.9</td>
<td>4.061636994848069e-001</td>
<td>4.061636994614400e-001</td>
<td>2.33686399928567e-011</td>
</tr>
</tbody>
</table>

---

**Fig. 3.** Comparison between numerical and exact solutions for Problems 2 and 3 (sub-figures a, b, respectively).
we obtain, after some manipulations, a system of explicit difference equations as follows

\[ u_{i,j+2} = f(x_i, t_j, u_{i-1,j}, u_{i-1,j}, u_{i+1,j}, u_{i+1,j+1}, u_{i,j-1}, u_{i,j-1}), \]

with the initial conditions (22) and the boundary conditions (24).

3. Implementation

We implemented the third-order PDEs of type I using MATLAB. We tested our proposed method in the following study cases.

Problem 1. (Homogenous)

\[ u_{i \mathrel{|} t} = -4(u - u_{\infty}), \quad a \leq x \leq b, \quad t > 0, \]

Initial conditions: \( u(x, 0) = \cos x \); \( u_{\infty}(x, 0) = -\sin x \); \( u_{\infty}(x, 0) = \cos x \).

Boundary conditions: \( u(a, t) = e^{-2t} \cos a \); \( u(b, t) = e^{-2t} \cos b \).

Exact solution: \( u(x, t) = e^{-2t} \cos x \).

(a) case 1 \( a = 0, b = 1 \); (see Table 6, Fig. 2),

(b) case 2 \( a = -\frac{\pi}{2}, b = \frac{\pi}{2} \); (see Table 7, Fig. 2).
Problem 2. (Homogenous)

\[
\begin{align*}
\text{Initial conditions: } & \quad u(x,0) = e^{-x}; \quad u_t(x,0) = -e^{-x}; \quad u_{xx}(x,0) = e^{-x}, \\
\text{Boundary conditions: } & \quad u(a,t) = e^{-a}e^{-t}; \quad u(b,t) = e^{-b}e^{-t}, \\
\text{Exact solution: } & \quad u(x,t) = e^{-t}e^{-x},
\end{align*}
\]

\[a = 0, \ b = 1; \text{ (see Table 8, Figs. 3, and 4).}\]

Table 9
Comparison between Exact and Numerical Solutions for RKD5 for Problems 3, \(a = -\pi, b = \pi\).

<table>
<thead>
<tr>
<th>Times (t_j)</th>
<th>(x_i)</th>
<th>Exact solution</th>
<th>Numerical solution</th>
<th>Absolute error</th>
</tr>
</thead>
<tbody>
<tr>
<td>(10^{-6})</td>
<td>(\pi/4)</td>
<td>3.090169634732494e-001</td>
<td>3.090171943749472e-001</td>
<td>2.309016978729872e-007</td>
</tr>
<tr>
<td>(50(10^{-6}))</td>
<td>(\pi/4)</td>
<td>-8.09010302013856e-001</td>
<td>-8.090071943749475e-001</td>
<td>5.835826438094216e-006</td>
</tr>
<tr>
<td>(100(10^{-6}))</td>
<td>0</td>
<td>9.99991000489978e-001</td>
<td>1.000019799999999e+000</td>
<td>2.969995100088330e-005</td>
</tr>
<tr>
<td>(150(10^{-6}))</td>
<td>(3\pi/4)</td>
<td>-8.09004901115527e-001</td>
<td>-8.089871943749517e-001</td>
<td>1.77457360096035e-005</td>
</tr>
<tr>
<td>(200(10^{-6}))</td>
<td>(3\pi/4)</td>
<td>3.090108449973623e-001</td>
<td>3.090567943749367e-001</td>
<td>4.59493775746127e-005</td>
</tr>
</tbody>
</table>

Fig. 5. Comparison between RKD and Finite Difference Method (FDM) for Problem 2 with time instants \(t = T/4\) and \(t = T/2\) in direction of \(t\) over the interval \([0, T]\) (sub-figures a, b).
**Problem 3.** (Non homogenous)

$$u_{ttt} = 4u + u_{xx} - 8 \cos 2t - 4 \sin 2t, \quad a \leq x \leq b, \quad t > 0.$$  

Initial conditions: $$u(x, 0) = \cos 2x; \quad u_x(x, 0) = -2 \sin 2x; \quad u_{xx}(x, 0) = -4 \cos 2x,$$

Boundary conditions: $$u(a, t) = \cos 2a + \sin 2t; \quad u(b, t) = \cos 2b + \sin 2t,$$

Exact solution: $$u(x, t) = \cos 2x + \sin 2t,$$

$$a = -\pi, \quad b = \pi; \quad \text{(see Table 9, Fig. 3)}.$$

A comparative simulation has been performed between the proposed method and finite difference method (FDM). Results of **Problem 2** are shown in **Figs. 5 and 6**, where the proposed method outperforms FDM.

### 4. Discussion and Conclusion

In this paper, three types of third-order PDEs are classified as Types I, II, and III. These classes of third-order PDEs usually occur in many subfields of physics and engineering. Computing a solution for PDEs directly by using classical methods can be difficult. We established a new numerical method for solving third-order PDEs of type I. The PDE of type I is converted to a system of third-order ODEs by using the method of lines. The system of ODEs is then solved by using direct integrators of Runge–Kutta type for special third-order ODEs, which we derived purposely to solve special third-order ODEs of the form $y''' = f(x, y)$. The proposed direct method technique requires less computational work; also, it has good accuracy.

We examined the proposed method by using various examples of third-order PDEs to probe the efficiency of the new method. Numerical results indicate that the proposed method is in good agreement with exact solutions. Also, the new method provides encouraging results and is shown to be more efficient and accurate than the existing finite difference method.

### References


