symmetry

MDPI

Article

Diagonally Implicit Multistep Block Method of Order Four for Solving Fuzzy Differential Equations Using Seikkala Derivatives

Syahirbanun Isa [1,2,3], Zanariah Abdul Majid [1,2,*], Fudziah Ismail [1,2] and Faranak Rabiei [1,2]

[1] Institute for Mathematical Research, Universiti Putra Malaysia, Serdang 43400, Selangor, Malaysia; syahirbanun.isa@gmail.com (S.I.); fudziah@upm.edu.my (F.I.); faranak_rabiei@upm.edu.my (F.R.)
[2] Department of Mathematics, Faculty of Science, Universiti Putra Malaysia, Serdang 43400, Selangor, Malaysia
[3] Department of Mathematics and Statistics, Faculty of Applied Science and Technology, Universiti Tun Hussein Onn Malaysia (Pagoh Campus), Muar 84000, Johor, Malaysia
* Correspondence: am_zana@upm.edu.my; Tel.: +60-389-467-959

Received: 31 October 2017; Accepted: 8 January 2018; Published: 8 February 2018

Abstract: In this paper, the solution of fuzzy differential equations is approximated numerically using diagonally implicit multistep block method of order four. The multistep block method is well known as an efficient and accurate method for solving ordinary differential equations, hence in this paper the method will be used to solve the fuzzy initial value problems where the initial value is a symmetric triangular fuzzy interval. The triangular fuzzy number is not necessarily symmetric, however by imposing symmetry the definition of a triangular fuzzy number can be simplified. The symmetric triangular fuzzy interval is a triangular fuzzy interval that has same left and right width of membership function from the center. Due to this, the parametric form of symmetric triangular fuzzy number is simple and the performing arithmetic operations become easier. In order to interpret the fuzzy problems, Seikkala's derivative approach is implemented. Characterization theorem is then used to translate the problems into a system of ordinary differential equations. The convergence of the introduced method is also proved. Numerical examples are given to investigate the performance of the proposed method. It is clearly shown in the results that the proposed method is comparable and reliable in solving fuzzy differential equations.

Keywords: block method; fuzzy differential equations; predictor-corrector; Seikkala's derivatives

---

1. Introduction

Modeling of real-world problems involves ordinary differential equations (ODEs) that are not always perfect. For example, the initial value may not be known exactly and the function may contain uncertain parameters. This inexactness leads to the necessity of fuzzy differential equations (FDEs) to overcome the situation. Examples of FDEs application in real life include models such as in solid waste management systems [1], hydraulic differential servo cylinders [2], population models [3], hyperchaotic systems [4], and service composition [5].

Chang and Zadeh in [6] first introduced the fuzzy derivative concept. The concept was then extended by Dubois and Prade [7] and the authors also addressed the problem of linearity. Goetschel and Voxman [8] and Puri and Ralescu [9] studied and proposed several important definitions and theorems in the fuzzy derivative. A significant contribution for fuzzy initial value problems (FIVPs) is done by Seikkala [10] and Kaleva [11,12].

The exact solutions for certain FDEs are inconvenient and tedious to determine because of the complexity of the FDE problems. Therefore, numerical methods are used for solving the FDEs.

Ma et al. in [13] contribute an important work in solving FDEs by using a numerical method. They adopted classical Euler's method and solve the FIVPs.

Several researchers discussed predictor–corrector methods for solving FDEs [14–17]. Allahviranloo et al. in [15] developed a predictor–corrector method based on the Adams-Bashforth three-step method as a predictor and the Adams-Moulton two-step method as a corrector. The improvise method of [15] is then proposed in [14]. Shang and Guo [17] derived the predictor–corrector method based on Adams-Bashforth four-step method and the Adams-Moulton three-step method as a predictor and corrector respectively. Jayakumar [16] introduced Adams fifth order predictor–corrector method for solving FDEs by considering the Adams-Bashforth five-step method as a predictor and the Adam-Moulton four-step method as a corrector. All studies aimed to determine better accuracy and efficiency in finding solutions to FDEs.

The multistep block method is an efficient method for solving ODEs. In this method, several approximation points will be computed simultaneously on the x-axis in the block. Mehrkanoon et al. in [18] and Isa and Majid in [19] solved FDEs by using the 2-point fully implicit multistep block method. Diagonally implicit multistep block method is used by Zawawi et al. [20] and Ramli and Majid [21,22] for solving FDEs. Zawawi et al. proposed diagonally implicit block backward differentiation formulas, while Ramli and Majid developed an Adams-type diagonally implicit multistep block method, which considered order four for first point and order five for the second point. Fook and Ibrahim in [23] solved FDEs by using a 2-points hybrid block method.

This paper will discuss the implementation of a 2-point diagonally implicit multistep block method to solve fuzzy differential equations. The method will compute the numerical solutions at two points simultaneously using constant step size. The two formulas of the diagonally implicit block method of Adams type will have the same order for the first and second formula.

This paper is organized as follows. Some basic definitions and notations are in Section 2. Definition of FIVP in Section 3 and description of the 2-point diagonally implicit multistep block method of order four (2PDO4) will be discussed in Section 4. In the next section, the implementation of the block method for solving FIVP and its algorithm is proposed in Sections 5 and 6 respectively. The convergence of the method is proved in Section 7. Numerical results are provided in Section 8 and conclusion of this study is at the end of the paper.

2. Preliminaries

Some basic definitions and notations that are significant for this paper will be reviewed in this section. For further details, refer [15,24–26] (to name a few).

Definition 1. *Let $\eta : R \rightarrow [0,1]$ is a mapping of fuzzy number with R is set of all real numbers. The mapping has following properties:*

1. *η upper semicontinuous,*
2. *η is fuzzy convex, that is, $\eta(\lambda x + (1-\lambda)y) \geq \min\{\eta(x), \eta(y)\}$, for all $x, y \in R$, $\lambda \in [0,1]$,*
3. *η is normal, that is exist $x_0 \in R$ such that $\eta(x_0) = 1$,*
4. *The support of η is $supp\ \eta = \{x \in R | \eta(x) > 0\}$ and its closure $cl(supp\ \eta)$ is compact.*

Consider E as set of all fuzzy numbers on R.

Definition 2. *The r-level set of a fuzzy number $\eta \in E, 0 \leq r \leq 1$ and denoted by $[\eta]_r$, is defined as*

$$[\eta]_r = \begin{cases} \{x \in R | \eta(x) > r\}, & if\ 0 \leq r \leq 1, \\ cl(supp\ \eta), & if\ r = 0, \end{cases}$$

and is closed and bounded interval by $\left[\underline{\eta}(r), \overline{\eta}(r)\right]$ where $\underline{\eta}(r)$ and $\overline{\eta}(r)$ denotes the left-hand endpoint and right-hand endpoint of $[\eta]_r$ respectively.

The summation and scalar multiplication in E are defined as:

1. $\eta \oplus \mu = \left(\underline{\eta} + \underline{\mu}, \overline{\eta} + \overline{\mu}\right)$,
2. $\lambda \odot \eta = \begin{cases} \left(\lambda \underline{\eta}(r), \lambda \overline{\eta}(r)\right), & \lambda \geq 0, \\ \left(\lambda \overline{\eta}(r), \lambda \underline{\eta}(r)\right), & \lambda < 0. \end{cases}$

Given the Hausdorff distance $D : E \times E \to R_+ \cup 0$ as the distance between two fuzzy numbers and a metric in E as follows

$$D(\eta, \mu) = \underset{r \in [0,1]}{\sup\max} \left\{ \left| \underline{\eta}(r) - \underline{\mu}(r) \right|, \left| \overline{\eta}(r) - \overline{\mu}(r) \right| \right\}$$

with the following properties

1. $D(\eta \oplus \omega, \mu \oplus \omega) = D(\eta, \mu)$, $\forall \eta, \mu, \omega \in E$,
2. $D(\lambda \odot \eta, \lambda \odot \mu) = |\lambda| D(\eta, \mu)$, $\forall \lambda \in R, \forall \eta, \mu \in E$,
3. $D(\eta \oplus \mu, \omega \oplus \sigma) \leq D(\eta, \omega) + D(\mu, \sigma)$, $\forall \eta, \mu, \omega, \sigma \in E$,
4. $D(\eta + \mu, 0) \leq D(\eta, 0) + D(\mu, 0)$, $\forall \eta, \mu \in E$,
5. (E, D) is a complete metric space.

Let $f : R \to E$ be a fuzzy valued function.

Definition 3. *If for arbitrary fixed* $t_0 \in R$ *and* $\varepsilon > 0$, $\delta > 0$ *such that*

$$|t - t_0| < \delta \Longrightarrow D(f(t), f(t_0)) < \varepsilon$$

is said to be continuous.

Hukuhara differentiability (H-derivative) for fuzzy mappings is introduced by Puri and Ralescu in [9]. The differentiability is based on H-difference sets. Definitions 4 and 5 explained the concept.

Definition 4. *Let* $x, y \in E$. *If there exist* $z \in E$ *such that* $x = y \oplus z$, *then* z *is called the H-difference of* x *and* y, *and it is denoted by* $x \ominus y$. *The sign "*$\ominus$*" stands for H-difference and note that* $x \ominus y \neq x + (-1)y$.

Definition 5. f *is differentiable at* $t_0 \in R$ *if exist* $f'(t_0) \in E$ *such that*

$$\lim_{h \to 0^+} \frac{f(t_0 + h) \ominus f(t_0)}{h} \text{ and } \lim_{h \to 0^+} \frac{f(t_0) \ominus f(t_0 - h)}{h}$$

exist and are equal to $f'(t_0)$.

Here, the limits are taken in the metric space (E, D) since we have defined $h^{-1} \odot (f(t_0) \ominus f(t_0 - h))$ and $h^{-1} \odot (f(t_0 + h) \ominus f(t_0))$.

3. Fuzzy Initial Value Problems

In this paper, we will consider first order FIVP as follows

$$y'(t) = f(t, y(t)),\ y(t_0) = y_0,\ t \in [t_0, T] \tag{1}$$

where y is a fuzzy function of t, $f(t, y)$ is a fuzzy function of crisp variable t and fuzzy variable y, and y' is Hukuhara fuzzy derivative of y with the given initial value $y(t_0) = y_0$. Let $y(t) = \left[\underline{y}(t), \overline{y}(t)\right]$, if $y(t)$ is Hukuhara differentiable then $y'(t) = \left[\underline{y}'(t), \overline{y}'(t)\right]$, and $f(t, y(t)) = \left[\underline{f}\left(t, \underline{y}(t), \overline{y}(t)\right), \overline{f}\left(t, \underline{y}(t), \overline{y}(t)\right)\right]$.

The parametric form of y is given by

$$[y(t)]_r = \left[\underline{y}(t;r), \overline{y}(t;r)\right],\ t \in R,\ r \in [0,1].$$

As referred to [21], the Seikkala derivative of a fuzzy function is defined by

$$[y'(t)]_r = \left[\underline{y}'(t;r), \overline{y}'(t;r)\right],\ t \in R,\ r \in [0,1]$$

provided that the equation defines a fuzzy number.

For solving (1) by using numerical methods for ODEs, it must be translated to system of ODEs. Bede in [27] proposed characterization theorems for the solutions of FDEs by using ODEs.

Theorem 1. *Let $f : (a,b) \to E$ be Hukuhara differentiable and denote $[f(y)]_r = \left[\underline{f}(y(t;r)), \overline{f}(y(t;r))\right]$. Functions $\underline{f}(y(t;r))$ and $\overline{f}(y(t;r))$ are differentiable and based on Seikkala derivative*

$$[f'(y)]_r = \left[\underline{f}'(y(t;r)), \overline{f}'(y(t;r))\right],\ r \in [0,1].$$

Therefore (1) can be translated to the system of ODEs

$$\begin{aligned} \underline{y}'(t;r) &= \underline{f}\left(t, \underline{y}(t;r), \overline{y}(t;r)\right), \\ \overline{y}'(t;r) &= \overline{f}\left(t, \underline{y}(t;r), \overline{y}(t;r)\right), \\ \underline{y}'(t_0;r) &= \underline{y}'_0(r), \\ \overline{y}'(t_0;r) &= \overline{y}'_0(r). \end{aligned} \tag{2}$$

Theorem 2. *Consider FIVP (1) where $f : [t_0, t_0 + a] \times E \to E$ such that*

1. $[f'(t,y)]_r = \left[\underline{f}\left(t, \underline{y}(t;r), \overline{y}(t;r)\right), \overline{f}\left(t, \underline{y}(t;r), \overline{y}(t;r)\right)\right]$,
2. *$\underline{f}$ and $\overline{f}$ are equicontinuous (that is for any $\varepsilon > 0$ and any $(t, \eta(t;r), \mu(t;r)) \in [t_0, t_0 + a] \times R^2$ we have $\left|\underline{f}(t, \eta(t;r), \mu(t;r)) - \underline{f}(t_1, \eta_1(t;r), \mu_1(t;r))\right| < \varepsilon$ and $\left|\overline{f}(t, \eta(t;r), \mu(t;r)) - \overline{f}(t_1, \eta_1(t;r), \mu_1(t;r))\right| < \varepsilon$ for all $r \in [0,1]$ whenever $||(t_1, \eta_1(t;r), \mu_1(t;r)) - (t, \eta(t;r), \mu(t;r))|| < \delta)$ and uniformly bounded on any bounded set,*
3. *there exist an $L > 0$ such that*

$$\left|\underline{f}(t, \eta(t;r), \mu(t;r)) - \underline{f}(t_1, \eta_1(t;r), \mu_1(t;r))\right| \le L max\{|\eta - \eta_1|, |\mu -, \mu_1|\}, \forall r \in [0,1] \text{ and}$$

$$\left|\overline{f}(t, \eta(t;r), \mu(t;r)) - \overline{f}(t_1, \eta_1(t;r), \mu_1(t;r))\right| \le L max\{|\eta - \eta_1|, |\mu -, \mu_1|\}, \forall r \in [0,1].$$

Then FIVP (1) and the system of ODEs (2) are equivalent.

4. Derivation of 2-Point Diagonally Implicit Multistep Block Method

Consider the initial value problems (IVPs) for ODEs of the form

$$y' = f(x,y),\ y(x_0) = y_0,\ x \in [a,b] \tag{3}$$

where a and b are finite. The interval $[a,b]$ will be divided into a series of blocks. In this proposed method, each block contained two points, y_{n+1} and y_{n+2} which will be computed simultaneously in a block with the step size h at the points x_{n+1} and x_{n+2} respectively as in Figure 1.

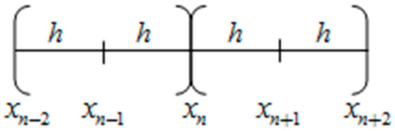


Figure 1. 2-point implicit multistep block method.

The Lagrange interpolation polynomial is applied for the derivation of predictor and corrector formulas. To obtain the predictor formulas, the set of points $\{(x_{n-2}, f_{n-2}), (x_{n-1}, f_{n-1}), (x_n, f_n)\}$ is interpolated and the order of the formulas is one less than the corrector formulas. The corrector formulas will use the set of interpolation points i.e., $\{(x_{n-2}, f_{n-2}), (x_{n-1}, f_{n-1}), (x_n, f_n), (x_{n+1}, f_{n+1})\}$ for derivation of y_{n+1} and $\{(x_{n-1}, f_{n-1}), (x_n, f_n), (x_{n+1}, f_{n+1}), (x_{n+2}, f_{n+2})\}$ for derivation of y_{n+2}. The formulas of y_{n+1} will used three back values $(x_n,\ x_{n-1}, x_{n-2})$, while the y_{n+2} will used two back values i.e., $(x_n,\ x_{n-1})$.

Formulas for y_{n+1} and y_{n+2} are derived by integrating (3) such that

$$\int y' dx = \int f(x, y) dx \tag{4}$$

where the interval of integration for y_{n+1} and y_{n+2} are $[x_n, x_{n+1}]$ and $[x_n, x_{n+2}]$ respectively. These are equivalent to

$$y_{n+1} = y_n + \int_{x_n}^{x_{n+1}} f(x, y) dx \tag{5}$$

and

$$y_{n+2} = y_n + \int_{x_n}^{x_{n+2}} f(x, y) dx. \tag{6}$$

The function $f(x, y)$ in (5) and (6) is replaced by the Lagrange polynomial which interpolates the set of corresponding mentioned points. Evaluating the integral in (5) using MAPLE gives the formula for the first and second point of the block multistep method as follows:

The first point,

$$y_{n+1} = y_n + \frac{h}{24}(9f_{n+1} + 19f_n - 5f_{n-1} + f_{n-2}) \tag{7}$$

and the second point,

$$y_{n+2} = y_n + \frac{h}{3}(f_{n+2} + 4f_{n+1} + f_n). \tag{8}$$

As referred to [28,29],

$$\sum_{k=0}^{j} A_k Y_{n+k} = h \sum_{k=0}^{j} B_k F_{n+k} \tag{9}$$

form the general matrix of linear multistep method where A_k and B_k are r by r matrices with elements a_{lm} and b_{lm} for $l, m = 1, 2, ..., r$. The associated difference operator L is given by

$$L[z(x); h] = \sum_{k=0}^{j} [A_k z(x + h) - hB_k z'(x + kh)] \tag{10}$$

where $z(x)$ is the exact solution and assume to be sufficiently differentiable. If $C_0 = C_1 = ... = C_q = 0$, $C_{p+1} \neq 0$, L is said to be order p and C_{p+1} is called the error constant.

The general form of constant C_q is defined as

$$
\begin{aligned}
C_0 &= \sum_{k=0}^{j} A_k, \\
C_1 &= \sum_{k=0}^{j} [kA_k - B_k], \\
\vdots \quad & \vdots \quad \vdots \\
C_q &= \sum_{k=0}^{j} \left[\frac{1}{q!}k^q A_k - \frac{1}{(q-1)!}k^{q-1}B_k\right],\ q = 2, 3, \ldots, p+1.
\end{aligned}
\tag{11}
$$

To determine the order of the method, (7) and (8) is written in matrix form as follows

$$
\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} y_{n+1} \\ y_{n+2} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} y_{n-1} \\ y_{n} \end{bmatrix} + h\left\{\begin{bmatrix} \frac{9}{24} & 0 \\ \frac{4}{3} & \frac{1}{3} \end{bmatrix}\begin{bmatrix} f_{n+1} \\ f_{n+2} \end{bmatrix} + \begin{bmatrix} -\frac{5}{24} & \frac{19}{24} \\ 0 & \frac{1}{3} \end{bmatrix}\begin{bmatrix} f_{n-1} \\ f_{n} \end{bmatrix} + \begin{bmatrix} 0 & \frac{1}{24} \\ 0 & 0 \end{bmatrix}\begin{bmatrix} f_{n-3} \\ f_{n-2} \end{bmatrix}\right\} \tag{12}
$$

From (12), $A_2 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$, $A_3 = \begin{bmatrix} -1 \\ -1 \end{bmatrix}$, $A_4 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$, $A_5 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$, $B_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$, $B_1 = \begin{bmatrix} \frac{1}{24} \\ 0 \end{bmatrix}$, $B_2 = \begin{bmatrix} -\frac{5}{24} \\ 0 \end{bmatrix}$, $B_3 = \begin{bmatrix} \frac{19}{24} \\ \frac{1}{3} \end{bmatrix}$, $B_4 = \begin{bmatrix} \frac{9}{24} \\ \frac{4}{3} \end{bmatrix}$ and $B_5 = \begin{bmatrix} 0 \\ \frac{1}{3} \end{bmatrix}$. Hence, by applying (11), $C_0 = C_1 = C_2 = C_3 = C_4 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$ and $C_5 = \begin{bmatrix} -\frac{19}{720} \\ -\frac{1}{90} \end{bmatrix}$. Since $C_5 \neq \begin{bmatrix} 0 \\ 0 \end{bmatrix}$ the method is concluded as order four and the error constant is $\begin{bmatrix} -\frac{19}{720} \\ -\frac{1}{90} \end{bmatrix}$.

5. Implementation for Solving FIVPs

Consider the FIVP (1) and let the exact solution is denoted by $[\underline{Y}(t;r), \overline{Y}(t;r)]$ while $\left[\underline{y}(t;r), \overline{y}(t;r)\right]$ is the approximate solution. To integrate the system (2), from t_0 let $T > t_0$, and replace $[t_0, T]$ by

$$
t_0 < t_1 < t_2 < \ldots < t_n = T. \tag{13}
$$

$Y_n(t;r) = [\underline{Y}_n(t;r), \overline{Y}_n(t;r)]$ denoted as the exact solution at t_n, with the approximate solution $y_n(t;r) = \left[\underline{y}_n(t;r), \overline{y}_n(t;r)\right]$. Given $t_n = t_0 + h$ where $0 \le n \le N$ and $h = \frac{T-t_0}{N}$.

The general fuzzy 2PDO4 method by considering (7) and (8) is written as

$$
\begin{aligned}
&\underline{Y}(t_{n+1};r) \approx \underline{Y}(t_n;r) + \frac{h}{24}\Big[9\underline{f}\big(t_{n+1}, \underline{Y}_{n+1}(r), \overline{Y}_{n+1}(r)\big) + 19\underline{f}\big(t_n, \underline{Y}_n(r), \overline{Y}_n(r)\big) \\
&-5\underline{f}\big(t_{n-1}, \underline{Y}_{n-1}(r), \overline{Y}_{n-1}(r)\big) + \underline{f}\big(t_{n-2}, \underline{Y}_{n-2}(r), \overline{Y}_{n-2}(r)\big)\Big], \\
&\overline{Y}(t_{n+1};r) \approx \overline{Y}(t_n;r) + \frac{h}{24}\Big[9\overline{f}\big(t_{n+1}, \underline{Y}_{n+1}(r), \overline{Y}_{n+1}(r)\big) + 19\overline{f}\big(t_n, \underline{Y}_n(r), \overline{Y}_n(r)\big) \\
&-5\overline{f}\big(t_{n-1}, \underline{Y}_{n-1}(r), \overline{Y}_{n-1}(r)\big) + \overline{f}\big(t_{n-2}, \underline{Y}_{n-2}(r), \overline{Y}_{n-2}(r)\big)\Big],
\end{aligned}
\tag{14}
$$

$$
\begin{aligned}
&\underline{Y}(t_{n+2};r) \approx \underline{Y}(t_n;r) + \frac{h}{3}\Big[\underline{f}\big(t_{n+2}, \underline{Y}_{n+2}(r), \overline{Y}_{n+2}(r)\big) + 4\underline{f}\big(t_{n+1}, \underline{Y}_{n+1}(r), \overline{Y}_{n+1}(r)\big) \\
&+\underline{f}\big(t_n, \underline{Y}_n(r), \overline{Y}_n(r)\big)\Big], \\
&\overline{Y}(t_{n+2};r) \approx \overline{Y}(t_n;r) + \frac{h}{3}\Big[\overline{f}\big(t_{n+2}, \underline{Y}_{n+2}(r), \overline{Y}_{n+2}(r)\big) + 4\overline{f}\big(t_{n+1}, \underline{Y}_{n+1}(r), \overline{Y}_{n+1}(r)\big) \\
&+\overline{f}\big(t_n, \underline{Y}_n(r), \overline{Y}_n(r)\big)\Big],
\end{aligned}
\tag{15}
$$

for the exact solutions and

$$\begin{aligned}
&\underline{y}(t_{n+1};r) = \underline{y}(t_n;r) + \tfrac{h}{24}\Big[9\underline{f}\Big(t_{n+1},\underline{y}_{n+1}(r),\overline{y}_{n+1}(r)\Big) + 19\underline{f}\Big(t_n,\underline{y}_n(r),\overline{y}_n(r)\Big)\\
&-5\underline{f}\Big(t_{n-1},\underline{y}_{n-1}(r),\overline{y}_{n-1}(r)\Big) + \underline{f}\Big(t_{n-2},\underline{y}_{n-2}(r),\overline{y}_{n-2}(r)\Big)\Big],\\
&\overline{y}(t_{n+1};r) \approx \overline{y}(t_n;r) + \tfrac{h}{24}\Big[9\overline{f}\Big(t_{n+1},\underline{y}_{n+1}(r),\overline{y}_{n+1}(r)\Big) + 19\overline{f}\Big(t_n,\underline{y}_n(r),\overline{y}_n(r)\Big)\\
&-5\overline{f}\Big(t_{n-1},\underline{y}_{n-1}(r),\overline{y}_{n-1}(r)\Big) + \overline{f}\Big(t_{n-2},\underline{y}_{n-2}(r),\overline{y}_{n-2}(r)\Big)\Big],
\end{aligned} \tag{16}$$

$$\begin{aligned}
&\underline{y}(t_{n+2};r) \approx \underline{y}(t_n;r) + \tfrac{h}{3}\Big[\underline{f}\Big(t_{n+2},\underline{y}_{n+2}(r),\overline{y}_{n+2}(r)\Big) + 4\underline{f}\Big(t_{n+1},\underline{y}_{n+1}(r),\overline{y}_{n+1}(r)\Big)\\
&+\underline{f}\Big(t_n,\underline{y}_n(r),\overline{y}_n(r)\Big)\Big],\\
&\overline{y}(t_{n+2};r) = \overline{y}(t_n;r) + \tfrac{h}{3}\Big[\overline{f}\Big(t_{n+2},\underline{y}_{n+2}(r),\overline{y}_{n+2}(r)\Big) + 4\overline{f}(t_{n+1},\underline{Y}_{n+1}(r),\overline{Y}_{n+1}(r))\\
&+\overline{f}\Big(t_n,\underline{y}_n(r),\overline{y}_n(r)\Big)\Big],
\end{aligned} \tag{17}$$

for the approximate solutions.

6. Algorithm

The following algorithm is based on using 2PDO4 method. To approximate the solution of the following FIVP

$$y'(t) = f(t,y(t)),\ y(t_0) = y_0,\ t_0 \le t \le T.$$

For N as an arbitrary positive integer,

- Step 1. Let $h = \frac{T-t_0}{N}$. The initial value $y(t_0) = y_0$ is obtained from the FIVP and Runge–Kutta order four is used to determine the two starting points. Hence,

$$\begin{aligned}
&\underline{y}(t_0;r) = \underline{y}_0,\ \underline{y}(t_1;r) = \underline{y_1},\ \underline{y}(t_2;r) = \underline{y}_2,\\
&\overline{y}(t_0;r) = \overline{y}_0,\ \overline{y}(t_1;r) = \overline{y}_1,\ \overline{y}(t_2;r) = \overline{y}_2.
\end{aligned}$$

- Step 2. Let $n = 2$.
- Step 3. Let $t_{n+1} = t_1 + nh$ and $t_{n+2} = t_2 + nh$.
- Step 4. Let the predictor formulas

$$\begin{aligned}
&\underline{y}^P(t_{n+1};r) = \underline{y}(t_n;r) + \tfrac{h}{12}\Big[23\underline{f}\Big(t_n,\underline{y}_n(r),\overline{y}_n(r)\Big) - 16\underline{f}\Big(t_{n-1},\underline{y}_{n-1}(r),\overline{y}_{n-1}(r)\Big)\\
&+5\underline{f}\Big(t_{n-2},\underline{y}_{n-2}(r),\overline{y}_{n-2}(r)\Big)\Big],\\
&\overline{y}^P(t_{n+1};r) \approx \overline{y}(t_n;r) + \tfrac{h}{12}\Big[23\overline{f}\Big(t_n,\underline{y}_n(r),\overline{y}_n(r)\Big) - 16\Big(t_{n-1},\underline{y}_{n-1}(r),\overline{y}_{n-1}(r)\Big)\\
&+5\overline{f}\Big(t_{n-2},\underline{y}_{n-2}(r),\overline{y}_{n-2}(r)\Big)\Big],
\end{aligned}$$

$$\begin{aligned}
&\underline{y}^P(t_{n+2};r) = \underline{y}(t_n;r) + \tfrac{h}{3}\Big[19\underline{f}\Big(t_n,\underline{y}_n(r),\overline{y}_n(r)\Big) - 20\Big(t_{n-1},\underline{y}_{n-1}(r),\overline{y}_{n-1}(r)\Big)\\
&+7\underline{f}\Big(t_{n-2},\underline{y}_{n-2}(r),\overline{y}_{n-2}(r)\Big)\Big],\\
&\overline{y}^P(t_{n+2};r) \approx \overline{y}(t_n;r) + \tfrac{h}{3}\Big[19\overline{f}\Big(t_n,\underline{y}_n(r),\overline{y}_n(r)\Big) - 20\Big(t_{n-1},\underline{y}_{n-1}(r),\overline{y}_{n-1}(r)\Big)\\
&+7\overline{f}\Big(t_{n-2},\underline{y}_{n-2}(r),\overline{y}_{n-2}(r)\Big)\Big].
\end{aligned}$$

- Step 5. Let the corrector formulas

$$\begin{aligned}&\underline{y}^C(t_{n+1};r) = \underline{y}(t_n;r) + \tfrac{h}{24}\Big[9\underline{f}^P\Big(t_{n+1},\underline{y}_{n+1}(r),\overline{y}_{n+1}(r)\Big) + 19\underline{f}\Big(t_n,\underline{y}_n(r),\overline{y}_n(r)\Big)\\ &-5\underline{f}\Big(t_{n-1},\underline{y}_{n-1}(r),\overline{y}_{n-1}(r)\Big) + \underline{f}\Big(t_{n-2},\underline{y}_{n-2}(r),\overline{y}_{n-2}(r)\Big)\Big],\\ &\overline{y}^C(t_{n+1};r) \approx \overline{y}(t_n;r) + \tfrac{h}{24}\Big[9\overline{f}^P\Big(t_{n+1},\underline{y}_{n+1}(r),\overline{y}_{n+1}(r)\Big) + 19\overline{f}\Big(t_n,\underline{y}_n(r),\overline{y}_n(r)\Big)\\ &-5\overline{f}\Big(t_{n-1},\underline{y}_{n-1}(r),\overline{y}_{n-1}(r)\Big) + \overline{f}\Big(t_{n-2},\underline{y}_{n-2}(r),\overline{y}_{n-2}(r)\Big)\Big],\end{aligned}$$

$$\begin{aligned}&\underline{y}^C(t_{n+2};r) \approx \underline{y}(t_n;r) + \tfrac{h}{3}\Big[\underline{f}^P\Big(t_{n+2},\underline{y}_{n+2}(r),\overline{y}_{n+2}(r)\Big) + 4\underline{f}^P\Big(t_{n+1},\underline{y}_{n+1}(r),\overline{y}_{n+1}(r)\Big)\\ &+\underline{f}\Big(t_n,\underline{y}_n(r),\overline{y}_n(r)\Big)\Big],\\ &\overline{y}^C(t_{n+2};r) = \overline{y}(t_n;r) + \tfrac{h}{3}\Big[\overline{f}^P\Big(t_{n+2},\underline{y}_{n+2}(r),\overline{y}_{n+2}(r)\Big) + 4\overline{f}^P\left(t_{n+1},\underline{Y}_{n+1}(r),\overline{Y}_{n+1}(r)\right)\\ &+\overline{f}\Big(t_n,\underline{y}_n(r),\overline{y}_n(r)\Big)\Big].\end{aligned}$$

- $n = n + 2$.
- If $n \le N - 2$, go to Step 3.
- Algorithm is completed and $\Big[\underline{y}^C(t_n;r),\overline{y}^C(t_n;r)\Big]$ approximates $\big[\underline{Y}(t_n;r),\overline{Y}(t_n;r)\big]$.

7. Convergence

From (16) and (17)

$$\begin{aligned}\underline{y}(t;h;r) &= \Big\{\Big[t_0,\underline{y}_0(r)\Big],\ \Big[t_1,\underline{y}_1(r)\Big],\ \dots,\ \Big[t_N,\underline{y}_N(r)\Big]\Big\},\\ \overline{y}(t;h;r) &= \big\{[t_0,\overline{y}_0(r)],\ [t_1,\overline{y}_1(r)],\ \dots,\ [t_N,\overline{y}_N(r)]\big\},\end{aligned} \tag{18}$$

are the 2PDO4 method over the interval $t_0 \le t \le t_N$ and approximate to $\underline{Y}(t;r)$ and $\overline{Y}(t;r)$ respectively. Given

$$\lim_{h\to 0}\underline{y}(t;h;r) = \underline{Y}(t;r),\ \lim_{h\to 0}\overline{y}(t;h;r) = \overline{Y}(t;r), \tag{19}$$

is the convergence condition for the approximates. The following lemma is considered to proof the convergence.

Lemma 1. *[9] Let $\{w_n\}_{n=0}^N$ (a sequence of numbers) satisfy:*

$$|w_{n+1}| \le A|w_n| + B|w_{n-1}| + C,\ 0 \le n \le N-1$$

for some given positive constants $\boldsymbol{A}$, $\boldsymbol{B}$ and $\boldsymbol{C}$. Then

$$\begin{aligned}&|w_n| \le \Big(A^{n-1} + \beta_1 A^{n-3}B + \beta_2 A^{n-5}B^2 + \dots + \beta_l B^{[n/2]}\Big)|w_1|\\ &+\Big(A^{n-2}B + \gamma_1 A^{n-4}B^2 + \dots + \gamma_m A B^{[n/2]}\Big)|w_0|\\ &+\left(A^{n-2} + A^{n-3} + \dots + 1\right)C\\ &+\left(\delta_1 A^{n-4} + \delta_2 A^{n-5} + \dots + \delta_p A + 1\right)BC\\ &+\left(\zeta_1 A^{n-6} + \zeta_2 A^{n-7} + \dots + \zeta_q A + 1\right)B^2C\\ &+\left(\lambda_1 A^{n-8} + \lambda_2 A^{n-9} + \dots + \lambda_s A + 1\right)B^3C + \dots,\ n \text{ odd}\end{aligned}$$

and

$$\begin{aligned}|w_n| \leq {} & \left(A^{n-1} + \beta_1 A^{n-3} B + \beta_2 A^{n-5} B^2 + \dots + \beta_l B^{[n/2]-1}\right)|w_1| \\ & + \left(A^{n-2} B + \gamma_1 A^{n-4} B^2 + \dots + \gamma_m A B^{[n/2]}\right)|w_0| \\ & + \left(A^{n-2} + A^{n-3} + \dots + 1\right)C \\ & + \left(\delta_1 A^{n-4} + \delta_2 A^{n-5} + \dots + \delta_p A + 1\right)BC \\ & + \left(\zeta_1 A^{n-6} + \zeta_2 A^{n-7} + \dots + \zeta_q A + 1\right)B^2 C \\ & + \left(\lambda_1 A^{n-8} + \lambda_2 A^{n-9} + \dots + \lambda_s A + 1\right)B^3 C + \dots, \ n \text{ even}\end{aligned}$$

where for all l, m, p, q and s, β_l, γ_m, δ_p, ζ_q and λ_s are constants. The proof by using mathematical induction is straightforward.

Theorem 3. *For arbitrary fixed $r, (r \in [0,1])$, approximate solutions (16) and (17) converge to the exact solutions $\underline{Y}(t;r)$ and $\overline{Y}(t;r)$ for $\underline{Y}, \overline{Y} \in C^4[t_0, T]$.*

Proof. By applying (19), it is sufficient to show

$$\lim_{h\to 0} \underline{y}_N(r) = \underline{Y}(T;r), \ \lim_{h\to 0} \overline{y}_N(r) = \overline{Y}(T;r).$$

Consider the exact solutions (14) and (15), then

$$\begin{aligned}&\underline{Y}(t_{n+1};r) = \underline{Y}(t_n;r) + \tfrac{h}{24}\Big[9\underline{f}\big(t_{n+1}, \underline{Y}_{n+1}(r), \overline{Y}_{n+1}(r)\big) + 19\underline{f}\big(t_n, \underline{Y}_n(r), \overline{Y}_n(r)\big) \\ &-5\underline{f}\big(t_{n-1}, \underline{Y}_{n-1}(r), \overline{Y}_{n-1}(r)\big) + \underline{f}\big(t_{n-2}, \underline{Y}_{n-2}(r), \overline{Y}_{n-2}(r)\big)\Big] \\ &-\tfrac{19}{720} h^5 \underline{Y}^{(5)}\left(\underline{\xi}_n\right), \\ &\overline{Y}(t_{n+1};r) = \overline{Y}(t_n;r) + \tfrac{h}{24}\Big[9\overline{f}\big(t_{n+1}, \underline{Y}_{n+1}(r), \overline{Y}_{n+1}(r)\big) + 19\overline{f}\big(t_n, \underline{Y}_n(r), \overline{Y}_n(r)\big) \\ &-5\overline{f}\big(t_{n-1}, \underline{Y}_{n-1}(r), \overline{Y}_{n-1}(r)\big) + \overline{f}\big(t_{n-2}, \underline{Y}_{n-2}(r), \overline{Y}_{n-2}(r)\big)\Big] \\ &-\tfrac{19}{720} h^5 \overline{Y}^{(5)}\left(\overline{\xi}_n\right),\end{aligned} \tag{20}$$

$$\begin{aligned}&\underline{Y}(t_{n+2};r) = \underline{Y}(t_n;r) + \tfrac{h}{3}\Big[\underline{f}\big(t_{n+2}, \underline{Y}_{n+2}(r), \overline{Y}_{n+2}(r)\big) + 4\underline{f}\big(t_{n+1}, \underline{Y}_{n+1}(r), \overline{Y}_{n+1}(r)\big) \\ &+\underline{f}\big(t_n, \underline{Y}_n(r), \overline{Y}_n(r)\big)\Big] h^5 - \tfrac{1}{90}\underline{Y}^{(5)}\left(\underline{\xi}_n\right), \\ &\overline{Y}(t_{n+2};r) \approx \overline{Y}(t_n;r) + \tfrac{h}{3}\Big[\overline{f}\big(t_{n+2}, \underline{Y}_{n+2}(r), \overline{Y}_{n+2}(r)\big) + 4\overline{f}\big(t_{n+1}, \underline{Y}_{n+1}(r), \overline{Y}_{n+1}(r)\big) \\ &+\overline{f}\big(t_n, \underline{Y}_n(r), \overline{Y}_n(r)\big)\Big] - \tfrac{1}{90} h^5 \overline{Y}^{(5)}\left(\overline{\xi}_n\right)\end{aligned} \tag{21}$$

where $t_n < \underline{\xi}_n, \overline{\xi}_n < t_{n+1}$. Therefore, by subtracting (20) and (16), and (21) and (17) yield

$$\begin{aligned}
&\underline{Y}(t_{n+1};r) - \underline{y}(t_{n+1};r) = \underline{Y}(t_n;r) - \underline{y}(t_n;r) \\
&+\tfrac{9h}{24}\left[\underline{f}\left(t_{n+1},\underline{Y}_{n+1}(r),\overline{Y}_{n+1}(r)\right) - \underline{f}\left(t_{n+1},\underline{y}_{n+1}(r),\overline{y}_{n+1}(r)\right)\right] \\
&+\tfrac{19h}{24}\left[\underline{f}\left(t_n,\underline{Y}_n(r),\overline{Y}_n(r)\right) - \underline{f}\left(t_n,\underline{y}_n(r),\overline{y}_n(r)\right)\right] \\
&-\tfrac{5h}{24}\left[\underline{f}\left(t_{n-1},\underline{Y}_{n-1}(r),\overline{Y}_{n-1}(r)\right) - \underline{f}\left(t_{n-1},\underline{y}_{n-1}(r),\overline{y}_{n-1}(r)\right)\right] \\
&+\tfrac{h}{24}\left[\underline{f}\left(t_{n-2},\underline{Y}_{n-2}(r),\overline{Y}_{n-2}(r)\right) - \underline{f}\left(t_{n-2},\underline{y}_{n-2}(r),\overline{y}_{n-2}(r)\right)\right] \\
&-\tfrac{19}{720}h^5\underline{Y}^{(5)}\left(\underline{\xi}_n\right), \\
&\overline{Y}(t_{n+1};r) - \overline{y}(t_{n+1};r) = \overline{Y}(t_n;r) - \overline{y}(t_n;r) \\
&+\tfrac{9h}{24}\left[\overline{f}\left(t_{n+1},\underline{Y}_{n+1}(r),\overline{Y}_{n+1}(r)\right) - \overline{f}\left(t_{n+1},\underline{y}_{n+1}(r),\overline{y}_{n+1}(r)\right)\right] \\
&+\tfrac{19h}{24}\left[\overline{f}\left(t_n,\underline{Y}_n(r),\overline{Y}_n(r)\right) - \overline{f}\left(t_n,\underline{y}_n(r),\overline{y}_n(r)\right)\right] \\
&-\tfrac{5h}{24}\left[\overline{f}\left(t_{n-1},\underline{Y}_{n-1}(r),\overline{Y}_{n-1}(r)\right) - \overline{f}\left(t_{n-1},\underline{y}_{n-1}(r),\overline{y}_{n-1}(r)\right)\right] \\
&+\tfrac{h}{24}\left[\overline{f}\left(t_{n-2},\underline{Y}_{n-2}(r),\overline{Y}_{n-2}(r)\right) - \overline{f}\left(t_{n-2},\underline{y}_{n-2}(r),\overline{y}_{n-2}(r)\right)\right] \\
&-\tfrac{19}{720}h^5\overline{Y}^{(5)}\left(\overline{\xi}_n\right),
\end{aligned} \tag{22}$$

and

$$\begin{aligned}
&\underline{Y}(t_{n+2};r) - \underline{y}(t_{n+2};r) = \underline{Y}(t_n;r) - \underline{y}(t_n;r) \\
&+\tfrac{h}{3}\left[\underline{f}\left(t_{n+2},\underline{Y}_{n+2}(r),\overline{Y}_{n+2}(r)\right) - \underline{f}\left(t_{n+2},\underline{y}_{n+2}(r),\overline{y}_{n+2}(r)\right)\right] \\
&+\tfrac{4h}{3}\left[\underline{f}\left(t_{n+1},\underline{Y}_{n+1}(r),\overline{Y}_{n+1}(r)\right) - \underline{f}\left(t_{n+1},\underline{y}_{n+1}(r),\overline{y}_{n+1}(r)\right)\right] \\
&+\tfrac{h}{3}\left[\underline{f}\left(t_n,\underline{Y}_n(r),\overline{Y}_n(r)\right) - \underline{f}\left(t_n,\underline{y}_n(r),\overline{y}_n(r)\right)\right] \\
&-\tfrac{1}{90}h^5\underline{Y}^{(5)}\left(\underline{\xi}_n\right), \\
&\overline{Y}(t_{n+2};r) - \overline{y}(t_{n+2};r) = \overline{Y}(t_n;r) - \overline{y}(t_n;r) \\
&+\tfrac{h}{3}\left[\overline{f}\left(t_{n+2},\underline{Y}_{n+2}(r),\overline{Y}_{n+2}(r)\right) - \overline{f}\left(t_{n+2},\underline{y}_{n+2}(r),\overline{y}_{n+2}(r)\right)\right] \\
&+\tfrac{4h}{3}\left[\overline{f}\left(t_{n+1},\underline{Y}_{n+1}(r),\overline{Y}_{n+1}(r)\right) - \overline{f}\left(t_{n+1},\underline{y}_{n+1}(r),\overline{y}_{n+1}(r)\right)\right] \\
&+\tfrac{h}{3}\left[\overline{f}\left(t_n,\underline{Y}_n(r),\overline{Y}_n(r)\right) - \overline{f}\left(t_n,\underline{y}_n(r),\overline{y}_n(r)\right)\right] \\
&-\tfrac{1}{90}h^5\overline{Y}^{(5)}\left(\overline{\xi}_n\right).
\end{aligned} \tag{23}$$

Denote $w_n = \underline{Y}(t_n;r) - \underline{y}(t_n;r)$ and $v_n = \overline{Y}(t_n;r) - \overline{y}(t_n;r)$. Then (22) and (23) becomes

$$\begin{aligned}
|w_{n+1}| &\le \left(1 + \tfrac{19hL_2}{24}\right)|w_n| + \tfrac{9hL_1}{24}|w_{n+1}| + \tfrac{5hL_3}{24}|w_{n-1}| + \tfrac{hL_4}{24}|w_{n-2}| + \tfrac{19}{720}h^5\underline{M}, \\
|v_{n+1}| &\le \left(1 + \tfrac{19hL_6}{24}\right)|v_n| + \tfrac{9hL_5}{24}|v_{n+1}| + \tfrac{5hL_7}{24}|v_{n-1}| + \tfrac{hL_8}{24}|v_{n-2}| + \tfrac{19}{720}h^5\overline{M},
\end{aligned} \tag{24}$$

and

$$\begin{aligned}
|w_{n+2}| &\le \left(1 + \tfrac{hL_{11}}{3}\right)|w_n| + \tfrac{hL_9}{3}|w_{n+2}| + \tfrac{4hL_{10}}{3}|w_{n+1}| + \tfrac{1}{90}h^5\underline{M}, \\
|v_{n+2}| &\le \left(1 + \tfrac{hL_{14}}{3}\right)|v_n| + \tfrac{hL_{12}}{3}|v_{n+2}| + \tfrac{4hL_{13}}{3}|v_{n+1}| + \tfrac{1}{90}h^5\overline{M},
\end{aligned} \tag{25}$$

where $\underline{M} = \max\limits_{t_0 \le t \le T}\left|\underline{Y}^{(5)}(t;r)\right|$ and $\overline{M} = \min\limits_{t_0 \le t \le T}\left|\overline{Y}^{(5)}(t;r)\right|$.

Consider the first point (24), let $L = \max\{L_1, ..., L_8\} < \frac{24}{9h}$, then

$$\begin{aligned} |w_{n+1}| &\leq \left(1 + \tfrac{28hL}{24-9hL}\right)|w_n| + \tfrac{5hL}{24-9hL}|w_{n-1}| + \tfrac{hL}{24-9hL}|w_{n-2}| + \tfrac{19}{30(24-9hL)}h^5\underline{M}, \\ |v_{n+1}| &\leq \left(1 + \tfrac{28hL}{24-9hL}\right)|v_n| + \tfrac{5hL}{24-9hL}|v_{n-1}| + \tfrac{hL}{24-9hL}|v_{n-2}| + \tfrac{19}{30(24-9hL)}h^5\overline{M}, \end{aligned}$$

are resulted. Let $|u_n| = |w_n| + |v_n|$, then by Lemma 1 and $w_0 = v_0 = 0$ (also $w_1 = v_1 = 0$):

$$\begin{aligned} |u_n| \leq\ & \frac{\left(1+\frac{28hL}{24-9hL}\right)^{n-1}-1}{\frac{28hL}{24-9hL}} \times \frac{19}{30(24-9hL)}h^5\left(\underline{M}+\overline{M}\right) \\ & +\left\{\delta_1\left(1+\tfrac{28hL}{24-9hL}\right)^{n-4}+\delta_2\left(1+\tfrac{28hL}{24-9hL}\right)^{n-5}+...+\delta_p\left(1+\tfrac{28hL}{24-9hL}\right)+1\right\} \\ & \times\left(\tfrac{5hL}{24-9hL}\right)\left(\tfrac{19}{30(24-9hL)}\right)h^5\left(\underline{M}+\overline{M}\right) \\ & +\left\{\zeta_1\left(1+\tfrac{28hL}{24-9hL}\right)^{n-6}+\zeta_2\left(1+\tfrac{28hL}{24-9hL}\right)^{n-7}+...+\zeta_q\left(1+\tfrac{28hL}{24-9hL}\right)+1\right\} \\ & \times\left(\tfrac{hL}{24-9hL}\right)\left(\tfrac{19}{30(24-9hL)}\right)h^5\left(\underline{M}+\overline{M}\right) \\ & +\left\{\lambda_1\left(1+\tfrac{28hL}{24-9hL}\right)^{n-8}+\lambda_2\left(1+\tfrac{28hL}{24-9hL}\right)^{n-9}+...+\lambda_s\left(1+\tfrac{28hL}{24-9hL}\right)+1\right\} \\ & \times\left(\tfrac{5hL}{24-9hL}\right)^2\left(\tfrac{19}{30(24-9hL)}\right)h^5\left(\underline{M}+\overline{M}\right) \\ & +... \end{aligned}$$

are obtained. If $h \to 0$ then $w_n \to 0$ and $v_n \to 0$ which concludes the proof.

Consider the second point (25), let $L = \max\{L_9, ..., L_{14}\} < \frac{3}{h}$, then

$$\begin{aligned} |w_{n+2}| &\leq \tfrac{4hL}{3-hL}|w_{n+1}| + \left(1 + \tfrac{2hL}{3-hL}\right)|w_n| + \tfrac{1}{30(3-hL)}h^5\underline{M}, \\ |v_{n+2}| &\leq \tfrac{4hL}{3-hL}|v_{n+1}| + \left(1 + \tfrac{2hL}{3-hL}\right)|v_n| + \tfrac{1}{30(3-hL)}h^5\overline{M}, \end{aligned}$$

are resulted. Let $|u_n| = |w_n| + |v_n|$, then by Lemma 1 and $w_0 = v_0 = 0$ (also $w_1 = v_1 = 0$):

$$\begin{aligned} |u_n| \leq\ & \frac{\left(\frac{4hL}{3-hL}\right)^{n-1}-1}{\frac{1+hL}{3-hL}} \times \frac{1}{30(3-hL)}h^5\left(\underline{M}+\overline{M}\right) \\ & +\left\{\delta_1\left(\tfrac{4hL}{3-hL}\right)^{n-4}+\delta_2\left(\tfrac{4hL}{3-hL}\right)^{n-5}+...+\delta_p\left(\tfrac{4hL}{3-hL}\right)+1\right\} \\ & \times\left(1+\tfrac{2hL}{3-hL}\right)\left(\tfrac{1}{30(3-hL)}\right)h^5\left(\underline{M}+\overline{M}\right) \\ & +\left\{\zeta_1\left(\tfrac{4hL}{3-hL}\right)^{n-6}+\zeta_2\left(\tfrac{4hL}{3-hL}\right)^{n-7}+...+\zeta_q\left(\tfrac{4hL}{3-hL}\right)+1\right\} \\ & \times\left(1+\tfrac{2hL}{3-hL}\right)^2\left(\tfrac{1}{30(3-hL)}\right)h^5\left(\underline{M}+\overline{M}\right) \\ & +\left\{\lambda_1\left(\tfrac{4hL}{3-hL}\right)^{n-8}+\lambda_2\left(\tfrac{4hL}{3-hL}\right)^{n-9}+...+\lambda_s\left(\tfrac{4hL}{3-hL}\right)+1\right\} \\ & \times\left(1+\tfrac{2hL}{3-hL}\right)^3\left(\tfrac{1}{30(3-hL)}\right)h^5\left(\underline{M}+\overline{M}\right) \\ & +... \end{aligned}$$

are obtained. If $h \to 0$ then $w_n \to 0$ and $v_n \to 0$ which concludes the proof. □

Remark 1. *Above theorem results that convergence order is* $\boldsymbol{O}\left(\boldsymbol{h}^3\right)$.

8. Results and Discussion

To implement the 2PDO4 method, $PE(CE)^m$ mode is used where P, C and E stands for predictor, corrector and evaluation of a function respectively. The CE implementation is repeated m times for each step and step size $h = 0.1$ is considered in the numerical results.

The following FIVPs were tested to show the performance of the proposed method:

Problem 1. *(Ma et al. [13]) Consider linear FIVP:*

$$y'(t) = y(t),\ y(0) = [0.75 + 0.25r, 1.125 - 0.125r],\ t, r \in [0, 1].$$

Exact solution: $Y(t,r) = \left[(0.75 + 0.25r)e^t, (1.125 - 0.125r)e^t\right]$.

Problem 2. *(Pallingkinis et al. [30]) Consider linear FIVP:*

$$y'(t) = \bar{c}y(t),\ y(0) = (8/8.5/9),\ t, r \in [0, 1]$$

where triangular fuzzy number $\bar{c} = (1/2/3)$.

[$\bar{c}$ *is a triangular fuzzy number where the basis of the membership function of* $\bar{c}$ *is in the interval* $[c_1, c_3]$ *with the known summit* c_2*. Hence, the triangular fuzzy numbers are denoted by* $\bar{c} = (c_1/c_2/c_3)$*].*

Exact solution: $Y(t,r) = \left[(8 + 0.5r)e^{(1+r)t}, (9 - 0.5r)e^{(3-r)t}\right]$.

Problem 3. *(Ghanbari [31]) Consider nonlinear FIVP:*

$$y'(t) = 2y(t) + t^2 + 1,\ y(0) = [r, 2 - r],\ t, r \in [0, 1].$$

Exact solution:

$$Y(t,r) = \left[\left(r + \frac{3}{4}\right)e^{2t} - \frac{1}{4}\left(2t^2 + 2t + 3\right), \left(\frac{11}{4} - r\right)e^{2t} - \frac{1}{4}\left(2t^2 + 2t + 3\right)\right].$$

Problem 4. *(Dizicheh et al. [32]) Consider nonlinear FIVP:*

$$y'(t) = k_1 y^2(t) + k_2,\ y(0) = [-0.0012 + 0.0012r, 0.0012 - 0.0012r],$$

where $k_i > 0$ *for* $i = 1, 2$ $t, r \in [0, 1]$*. Given*

$$k_1(r) = [0.5 + 0.5r, 1.5 - 0.5r] \text{ and } k_2(r) = [0.75 + 0.25r, 1.25 - 0.25r].$$

Exact solution:

$$Y(t,r) = \left[\frac{A\tan\left(\frac{At}{4} + \tan^{-1}\left(\frac{-3+3r^2}{1250A}\right)\right)}{2 + 2r}, \frac{-\sqrt{2}B\tan\left(\frac{\sqrt{2}}{4}\left[Bt + 2\sqrt{2}\tan^{-1}\left(\frac{3\sqrt{2}(-3+r)(-1+r)}{2500B}\right)\right]\right)}{-6 + r}\right].$$

Problem 5. *(Ma et al. [13]) Consider nonlinear FIVP:*

$$y'(t) = e^{-y^2(t)},\ y(0) = [0.75 + 0.25r, 1.5 - 0.5r],\ t, r \in [0, 1].$$

The following notations are used in the tables:

r: Fuzzy numbers with bounded $\boldsymbol{r}$-level intervals
$\underline{Y}, \overline{Y}$: Lower and upper bounded exact solution
$\underline{y}, \overline{y}$: Lower and upper bounded approximated solution
FCN: Total function calls
TS: Total steps taken
TIME: The execution time taken in seconds
2PDO4: 2-point diagonally implicit multistep method of order four
ERK4: Extended Runge–Kutta-like formulae of order 4 [26]
RK4: Runge–Kutta method of order 4 [33]
ANN: Artificial neural network approach [34]

Absolute error is defined as $|Y_n(t;r) - y_n(t;r)|$, where $Y_n(t;r) = \left[\underline{Y}_n(t;r), \overline{Y}_n(t;r)\right]$, denoted as the exact solutions and $y_n(t;r) = \left[\underline{y}_n(t;r), \overline{y}_n(t;r)\right]$ as approximate solutions at t_n. The code is written in C language by using Microsoft visual C++ platform.

The numerical results of 2PDO4 compared to ERK4, RK4 and ANN are display in Tables 1–5 when solving Problem 1–5. Note that these results are at $t = 1$ with $h = 0.1$.

From Tables 1–3, it is obvious that 2PDO4 gives better results compared to ERK4, RK4 and ANN in terms of accuracy. It is also observed that the execution times of 2PDO4 for solving Problem 3 and 5 are faster than the RK4. Table 4 shows that 2PDO4 used less number of function evaluations to obtain comparable accuracy compared to ERK4 and RK4.

For Problem 5, the exact solutions cannot be calculated analytically. Hence, we compared the approximate solutions of 2PDO4 and RK4. The approximate solutions of Problem 5 at $t = 1$ with $h = 0.1$ are presented in Table 5. It can be observed that 2PDO4 is applicable to solve Problem 5 with an advantage of less total steps and faster execution times compared to RK4.

It is obvious in $1-5$ that 2PDO4 required less number of total steps compared to other methods. This is expected since 2PDO4 approximates the solutions at two points simultaneously.

Table 1. Numerical results of 2PDO4, ERK4, RK4 and ANN for solving Problem 1.

	$\left\lvert \underline{Y}_n(t;r) - \underline{y}(t;r) \right\rvert$			
r	**2PDO4**	**ERK4** [1]	**RK4** [1]	**ANN** [1]
0.0	7.112242(-7)	1.969260(-6)	1.492423(-6)	8.862866(-5)
0.1	7.349317(-7)	2.090533(-6)	1.375277(-6)	2.941706(-5)
0.2	7.586392(-7)	1.734969(-6)	1.734969(-6)	4.646277(-5)
0.3	7.823466(-7)	2.094660(-6)	1.856242(-6)	2.449152(-5)
0.4	8.060541(-7)	1.500678(-6)	1.739096(-6)	4.445810(-6)
0.5	8.297616(-7)	2.337206(-6)	1.621951(-6)	5.440010(-5)
0.6	8.534691(-7)	1.743224(-6)	1.743224(-6)	2.835439(-5)
0.7	8.771765(-7)	2.579752(-6)	1.626078(-6)	4.669132(-5)
0.8	9.008840(-7)	2.462607(-6)	1.985770(-6)	9.726296(-5)
0.9	9.245915(-7)	2.345461(-6)	2.107043(-6)	1.421725(-5)
1.0	9.482990(-7)	1.751478(-6)	1.989897(-6)	5.582846(-5)

Table 1. *Cont.*

$\left\lvert\overline{Y}_n(t;r) - \overline{y}_n(t;r)\right\rvert$				
r	**2PDO4**	**ERK4** [1]	**RK4** [1]	**ANN** [1]
0.0	1.066836(-6)	2.834681(-6)	2.596262(-6)	5.994298(-5)
0.1	1.054983(-6)	2.297207(-6)	2.058788(-6)	1.465839(-6)
0.2	1.043129(-6)	1.998152(-6)	2.236570(-6)	2.598870(-5)
0.3	1.031275(-6)	3.129608(-6)	2.175934(-6)	8.511551(-6)
0.4	1.019421(-6)	2.115298(-6)	2.115298(-6)	3.403441(-5)
0.5	1.007568(-6)	3.008335(-6)	2.293079(-6)	1.444274(-5)
0.6	9.957139(-7)	2.232443(-6)	2.232443(-6)	4.208012(-5)
0.7	9.838602(-7)	1.933388(-6)	2.171806(-6)	5.160297(-5)
0.8	9.720064(-7)	9.190771(-7)	1.872751(-6)	3.012583(-5)
0.9	9.601527(-7)	2.050534(-6)	2.050534(-6)	1.543513(-4)
1.0	9.482990(-7)	1.751478(-6)	1.989897(-6)	2.817154(-5)
FCN	451	451	451	-
TS	66	110	110	-
TIME	0.675	-	-	-

[1] Results of ERK4, RK4 and ANN as referred in [26].

Table 2. Numerical results of 2PDO4, ERK4 and RK4 for solving Problem 2.

$\left\lvert\underline{\boldsymbol{Y}}_n(\boldsymbol{t};\boldsymbol{r}) - \underline{\boldsymbol{y}}(\boldsymbol{t};\boldsymbol{r})\right\rvert$			
r	**2PDO4**	**ERK4** [1]	**RK4** [1]
0.0	7.586392(-6)	1.401183(-5)	1.591918(-5)
0.1	1.366979(-5)	2.666611(-5)	3.238815(-5)
0.2	2.357892(-5)	5.232929(-5)	4.851460(-5)
0.3	3.919819(-5)	8.186697(-5)	8.377432(-5)
0.4	6.313677(-5)	1.334720(-4)	1.296573(-4)
0.5	9.894536(-5)	2.104650(-4)	2.104650(-4)
0.6	1.513830(-4)	3.134706(-4)	3.134706(-4)
0.7	2.267426(-4)	4.676179(-4)	4.638032(-4)
0.8	3.332447(-4)	6.934862(-4)	6.858569(-4)
0.9	4.815107(-4)	9.960255(-4)	9.960255(-4)
1.0	6.851258(-4)	1.419590(-3)	1.423405(-3)
$\left\lvert\overline{Y}_n(t;r) - \overline{y}_n(t;r)\right\rvert$			
r	**2PDO4**	**ERK4** [1]	**RK4** [1]
0.0	1.260390(-2)	2.848404(-2)	2.849930(-2)
0.1	9.798071(-3)	2.185179(-2)	2.185179(-2)
0.2	7.561266(-3)	1.664276(-2)	1.662750(-2)
0.3	5.790072(-3)	1.253006(-2)	1.257584(-2)
0.4	4.397464(-3)	9.443902(-3)	9.421014(-3)
0.5	3.310672(-3)	7.054700(-3)	7.039441(-3)
0.6	2.469226(-3)	5.211172(-3)	5.211172(-3)
0.7	1.823209(-3)	3.812939(-3)	3.805309(-3)
0.8	1.331682(-3)	2.773261(-3)	2.773261(-3)
0.9	9.613063(-4)	1.989020(-3)	1.996649(-3)
1.0	6.851258(-4)	1.419590(-3)	1.423405(-3)
FCN	451	451	451
TS	66	110	110
TIME	0.639	-	-

[1] Results of ERK4 and RK4 as referred in [26].

Table 3. Numerical results of 2PDO4 and RK4 for solving Problem 3.

	$\left|\underline{\boldsymbol{Y}}_n(\boldsymbol{t};\boldsymbol{r})-\underline{\boldsymbol{y}}(\boldsymbol{t};\boldsymbol{r})\right|$	
r	**2PDO4**	**RK4**
0.0	6.963978(-5)	1.010956(-4)
0.1	7.770009(-5)	1.177814(-4)
0.2	8.576039(-5)	1.344671(-4)
0.3	9.382069(-5)	1.511528(-4)
0.4	1.018810(-4)	1.678386(-4)
0.5	1.099413(-4)	1.845243(-4)
0.6	1.180016(-4)	2.012100(-4)
0.7	1.260619(-4)	2.178957(-4)
0.8	1.341222(-4)	2.345815(-4)
0.9	1.421825(-4)	2.512672(-4)
1.0	1.502428(-4)	2.679529(-4)
	$\left|\overline{Y}_n(t;r)-\overline{y}_n(t;r)\right|$	
r	**2PDO4**	**RK4**
0.0	2.308459(-4)	4.348102(-4)
0.1	2.227856(-4)	4.181245(-4)
0.2	2.147253(-4)	4.014387(-4)
0.3	2.066649(-4)	3.847530(-4)
0.4	1.986046(-4)	3.680673(-4)
0.5	1.905443(-4)	3.513816(-4)
0.6	1.824840(-4)	3.346958(-4)
0.7	1.744237(-4)	3.180101(-4)
0.8	1.663634(-4)	3.013244(-4)
0.9	1.583031(-4)	2.846386(-4)
1.0	1.502428(-4)	2.679529(-4)
FCN	451	451
TS	66	110
TIME	0.530	0.718

Note: Simulation results of 2PDO4 and RK4 have been obtained in the same platform.

Table 4. Numerical results of 2PDO4, ERK4 and RK4 for solving Problem 4.

		$\left|\underline{\boldsymbol{Y}}_n(\boldsymbol{t};\boldsymbol{r})-\underline{\boldsymbol{y}}(\boldsymbol{t};\boldsymbol{r})\right|$	
r	**2PDO4**	**ERK4** [1]	**RK4** [1]
0.0	2.459944(-6)	1.9100(-5)	2.6100(-5)
0.1	3.707531(-6)	2.5100(-5)	3.0000(-5)
0.2	5.461503(-6)	2.9100(-5)	3.2200(-5)
0.3	7.872724(-6)	2.9000(-5)	4.2000(-5)
0.4	1.110316(-5)	3.5000(-5)	4.8000(-5)
0.5	1.529207(-5)	3.8000(-5)	5.5000(-5)
0.6	2.047821(-5)	3.9000(-5)	6.8000(-5)
0.7	2.643397(-5)	3.5000(-5)	7.6000(-5)
0.8	3.232537(-5)	3.4000(-5)	8.9000(-5)
0.9	3.602703(-5)	1.6000(-5)	1.0400(-4)
1.0	3.275027(-5)	1.0900(-4)	2.0000(-6)

Table 4. *Cont.*

	$\left\lvert\overline{Y}_n(t;r) - \overline{y}_n(t;r)\right\rvert$		
r	**2PDO4**	**ERK4** [1]	**RK4** [1]
0.0	6.395992(-2)	3.7712(-2)	8.7760(-3)
0.1	2.778184(-2)	1.7758(-2)	3.8480(-3)
0.2	1.248250(-2)	9.0110(-3)	1.8570(-3)
0.3	5.710280(-3)	4.8090(-3)	1.0570(-3)
0.4	2.616533(-3)	2.7230(-3)	6.4900(-4)
0.5	1.175996(-3)	1.6230(-3)	4.2000(-4)
0.6	5.009738(-4)	9.9600(-4)	3.1600(-4)
0.7	1.875370(-4)	6.3700(-4)	2.3300(-4)
0.8	4.657100(-5)	4.3100(-4)	1.8400(-4)
0.9	1.228793(-5)	2.9100(-4)	1.5000(-4)
1.0	3.275027(-5)	1.0900(-5)	2.0000(-6)
FCN	363	451	451
TS	66	110	110
TIME	0.748	-	-

[1] Results of ERK4 and RK4 as referred in [32].

Table 5. Numerical results of 2PDO4 and RK4 for solving Problem 5.

	$\underline{y}(t,r)$	
r	**2PDO4**	**RK4**
0.0	0.795170	0.795172
0.1	0.846301	0.846303
0.2	0.894275	0.894276
0.3	0.940192	0.940193
0.4	0.984968	0.984969
0.5	1.029399	1.029399
0.6	1.074198	1.074198
0.7	1.120029	1.120029
0.8	1.167520	1.167520
0.9	1.217275	1.217275
1.0	1.269868	1.269868
	$\overline{y}_n(t;r)$	
r	**2PDO4**	**RK4**
0.0	2.017674	2.017674
0.1	1.925744	1.925744
0.2	1.836622	1.836622
0.3	1.750863	1.750863
0.4	1.668972	1.668972
0.5	1.591344	1.591344
0.6	1.518228	1.518228
0.7	1.449698	1.449698
0.8	1.385652	1.385652
0.9	1.325836	1.325836
1.0	1.269868	1.269868
FCN	451	451
TS	66	110
TIME	0.390	0.468

Note: Simulation results of 2PDO4 and RK4 has been obtained in the same platform.

Figures 2–5 is a plot of the exact and approximate solutions by using 2PDO4 for Problems 1–4. It is clear in the figures that the solutions by the 2PDO4 method almost coincide with the exact solutions. While the plot of approximate solutions for Problem 5 by using 2PDO4 and RK4 is shown in Figure 6.

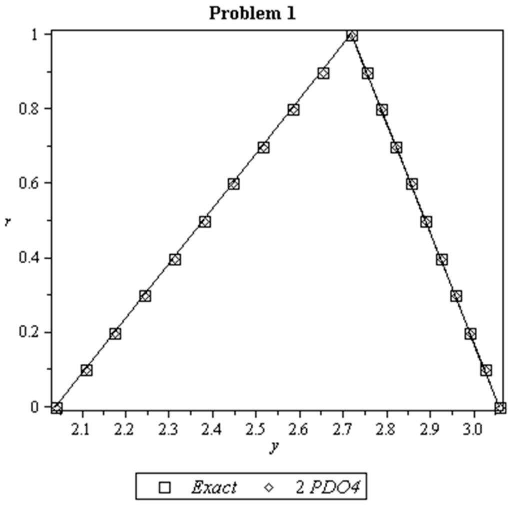


Figure 2. Exact solutions and approximate solutions for Problem 1 at $t = 1$ with $h = 0.1$.

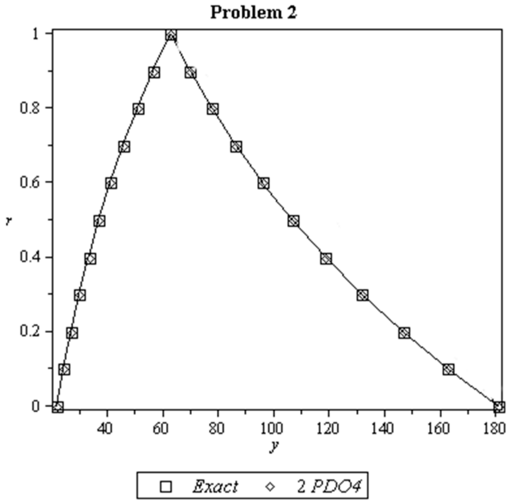


Figure 3. Exact solutions and approximate solutions for Problem 2 at $t = 1$ with $h = 0.1$.

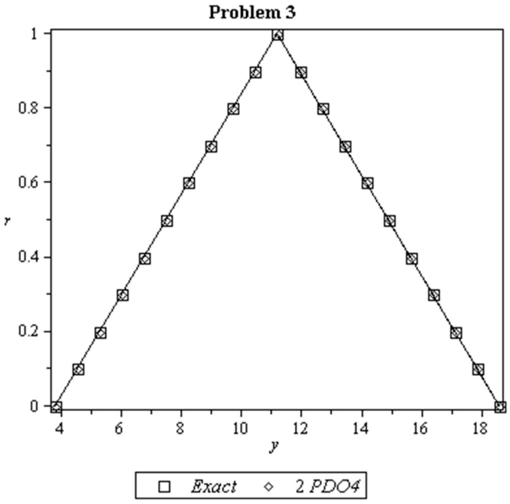


Figure 4. Exact solutions and approximate solutions for Problem 3 at $t = 1$ with $h = 0.1$.

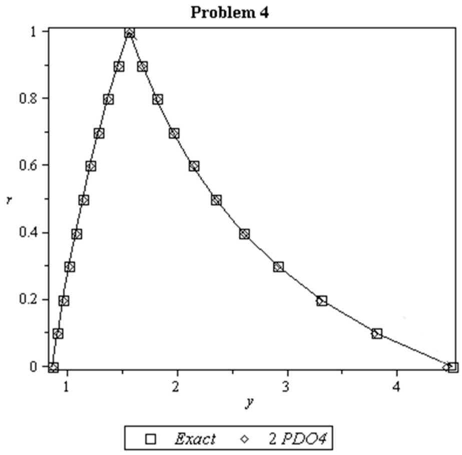


Figure 5. Exact solutions and approximate solutions for Problem 4 at $t = 1$ with $h = 0.1$.

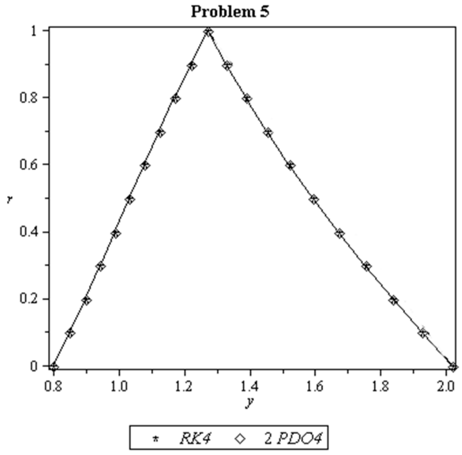
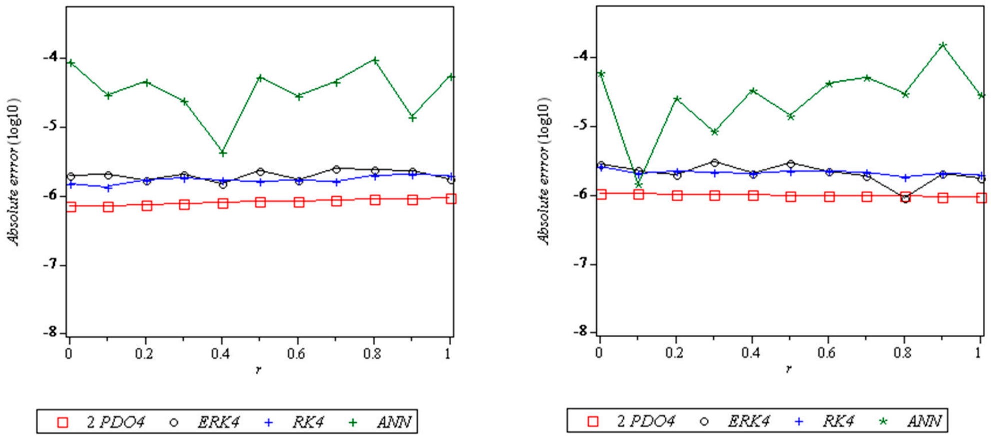


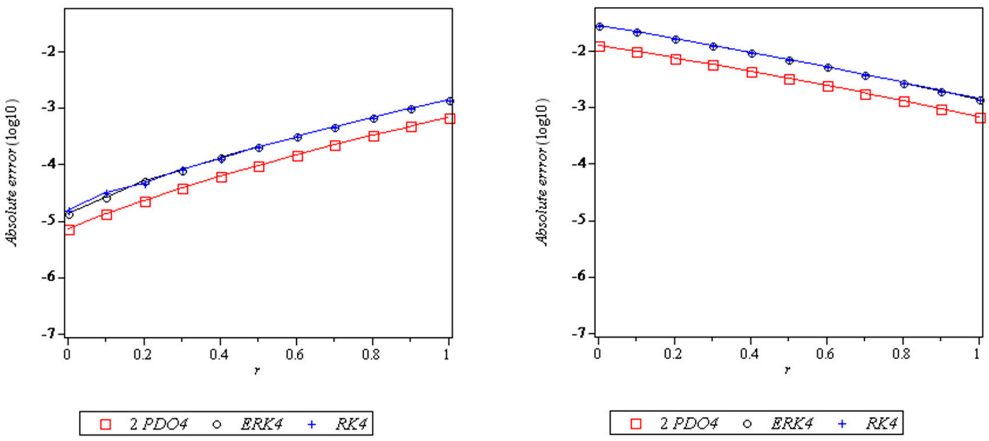
Figure 6. 2PDO4 solutions and RK4 solutions for Problem 5 at $t = 1$ with $h = 0.1$.

Figures 7–10 is a plot of the absolute errors versus r for lower and upper bounds of Problems 1–4. It can be seen in Figures 7–9 that the errors of 2PDO4 are smaller than the comparison methods, but in Figure 10 the errors of 2PDO4 are smaller at certain points.



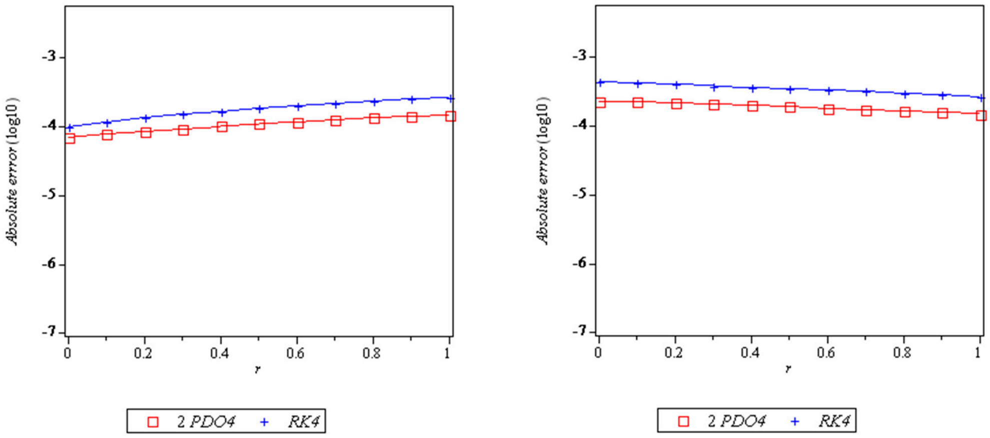
(**a**) (**b**)

Figure 7. Graph of the numerical results when solving Problem 1: (**a**) Graph of absolute errors versus r for lower bounds; (**b**) Graph of absolute errors versus r for upper bounds.



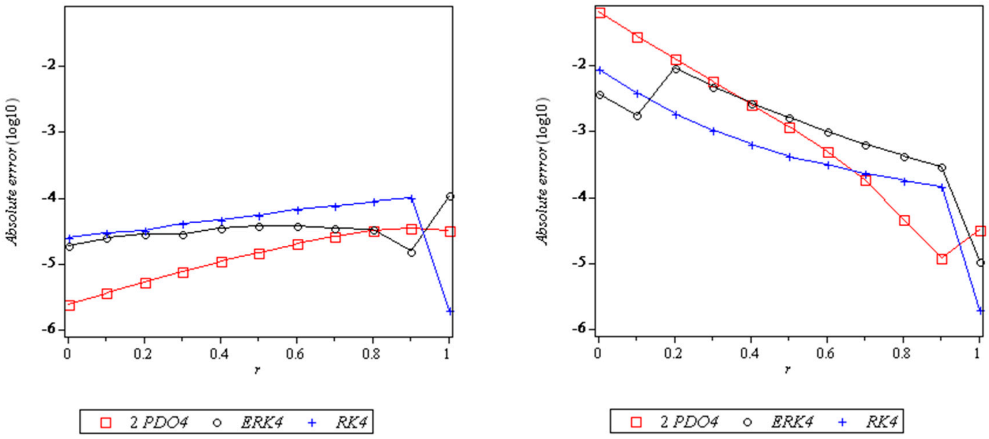
(**a**) (**b**)

Figure 8. Graph of the numerical results when solving Problem 2: (**a**) Graph of absolute errors versus r for lower bounds; (**b**) Graph of absolute errors versus r for upper bounds.



(**a**) (**b**)

Figure 9. Graph of the numerical results when solving Problem 3: (**a**) Graph of absolute errors versus r for lower bounds; (**b**) Graph of absolute errors versus r for upper bounds.



(**a**) (**b**)

Figure 10. Graph of the numerical results when solving Problem 4: (**a**) Graph of absolute errors versus r for lower bounds; (**b**) Graph of absolute errors versus for upper bounds.

9. Conclusions

In this paper, the diagonally implicit block method of order four (2PDO4) has been implemented for solving linear and nonlinear FDEs. Numerical results have shown that the proposed method gave comparable results in terms of accuracy and fewer total steps. The total functions are comparable or less compared to the existing method. The 2PDO4 is able to obtain less execution time than RK4 when solving two tested problems. Hence, the 2PDO4 method can be named in the literature as an additional and reliable method for solving FDEs.

Acknowledgments: Authors gratefully acknowledge the financial support of Research University Grant from Universiti Putra Malaysia and the scholarship of the Academic Training Scheme (SLAI) from Ministry of Higher Education Malaysia.

Author Contributions: Syahirbanun Isa conceived, designed, performed the experiments and wrote the paper. Zanariah Abdul Majid supervised the designed experiments and analyzed the data, Fudziah Ismail and Faranak Rabiei analyzed the data and checking the paper.

Conflicts of Interest: The authors declare that there is no conflict of interests regarding the publication of this paper.

References

1. Chen, H.W.; Chang, N.B. Prediction analysis of solid waste generation based on grey fuzzy dynamic modeling. *Resour. Conserv. Recycl.* **2000**, *29*, 1–18. [CrossRef]
2. Bencsik, A.L.; Bede, B.; Tar, J.K.; Fodor, J. Fuzzy differential equations in modeling of hydraulic differential servo cylinders. In Proceedings of the Third Romanian-Hungarian Joint Symposium on Applied Computational Intelligence (SACI), Timisoara, Romania, 25–26 May 2006.
3. Guo, M.; Xue, X.; Li, R. Impulsive functional differential inclusions and fuzzy population models. *Fuzzy Sets Syst.* **2003**, *138*, 601–615. [CrossRef]
4. Zhang, H.; Liao, X.; Yu, J. Fuzzy modeling and synchronization of hyperchaotic systems. *Chaos Solitons Fractals* **2005**, *26*, 835–843. [CrossRef]
5. Ding, Z.; Shen, H.; Kandel, A. Performance analysis of service composition based on fuzzy differential equations. *IEEE Trans. Fuzzy Syst.* **2011**, *19*, 164–178. [CrossRef]
6. Chang, S.S.L.; Zadeh, L.A. On fuzzy mapping and control. *IEEE Trans. Syst. Man Cybern.* **1972**, 30–34. [CrossRef]
7. Dubois, D.; Prade, H. Towards fuzzy differential calculus part 3: Differentiation. *Fuzzy Sets Syst.* **1982**, *8*, 225–233. [CrossRef]
8. Goetschel, R.; Voxman, W. Elementary fuzzy calculus. *Fuzzy Sets Syst.* **1986**, *18*, 31–43. [CrossRef]
9. Puri, M.L.; Ralescu, D.A. Differential of fuzzy functions. *J. Math. Anal. Appl.* **1983**, *91*, 552–558. [CrossRef]
10. Seikkala, S. On the fuzzy initial value problem. *Fuzzy Sets Syst.* **1987**, *24*, 319–330. [CrossRef]
11. Kaleva, O. Fuzzy differential equations. *Fuzzy Sets Syst.* **1987**, *24*, 301–317. [CrossRef]
12. Kaleva, O. The cauchy problem for fuzzy differential equations. *Fuzzy Sets Syst.* **1990**, *35*, 389–396. [CrossRef]
13. Ma, M.; Friedman, M.; Kandel, A. Numerical solutions of fuzzy differential equations. *Fuzzy Sets Syst.* **1999**, *105*, 133–138. [CrossRef]
14. Allahviranloo, T.; Abbasbandy, S.; Ahmady, N.; Ahmady, E. Improved predictor-corrector method for solving fuzzy initial value problems. *Inf. Sci.* **2009**, *179*, 945–955. [CrossRef]
15. Allahviranloo, T.; Ahmady, N.; Ahmady, E. Numerical solution of fuzzy differential equations by predictor-corrector method. *Inf. Sci.* **2007**, *177*, 1633–1647. [CrossRef]
16. Jayakumar, T.; Raja, C.; Muthukumar, T. Numerical solution of fuzzy differential equations by Adams fifth order predictor-corrector method. *Intern. J. Math. Trends Technol.* **2014**, *8*, 33–50.
17. Shang, D.; Guo, X. Adams predictor-corrector systems for solving fuzzy differential equations. *Math. Probl. Eng.* **2013**, *2013*. [CrossRef]
18. Mehrkanoon, S.; Suleiman, M.; Majid, Z.A. Block method for numerical solution of fuzzy differential equations. *Int. Math. Forum* **2009**, *4*, 2269–2280.

19. Isa, S.; Majid, Z.A. Numerical solution of fuzzy differential equations by multistep block method. In Proceedings of the 7th International Conference on Research and Educations in Mathematics (ICREM7), Kuala Lumpur, Malaysia, 25–27 August 2015; pp. 113–119.
20. Zawawi, I.S.M.; Ibrahim, Z.B.; Suleiman, M. Diagonally implicit block backward differentiation formulas for solving fuzzy differential equations. *AIP Conf. Proc.* **2013**, *1522*, 681–687. [CrossRef]
21. Ramli, A.; Majid, Z.A. An implicit multistep block method for fuzzy differential equations. In Proceedings of the 7th International Conference on Research and Educations in Mathematics (ICREM7), Kuala Lumpur, Malaysia, 25–27 August 2015; pp. 81–87.
22. Ramli, A.; Majid, Z.A. Fourth order diagonally implicit multistep block method for solving fuzzy differential equations. *Int. J. Pure Appl. Math.* **2016**, *107*, 635–660. [CrossRef]
23. Fook, T.K.; Ibrahim, Z.B. Two points hybrid block method for solving first order fuzzy differential equations. *J. Soft Comput. Appl.* **2016**, *2016*, 45–53. [CrossRef]
24. Ahmadian, A.; Suleiman, M.; Salahshour, S. An Operational Matrix Based on Legendre Polynomials for Solving Fuzzy Fractional-Order Differential Equations. *Abstr. Appl. Anal.* **2013**. [CrossRef]
25. Allahviranloo, T.; Salahshour, S. A new approach for solving first order fuzzy differential equation. In Proceedings of the Information Processing and Management of Uncertainty in Knowledge-Based Systems. Applications, Dortmund, Germany, 28 June 28–2 July 2010; pp. 522–531.
26. Ghazanfari, B.; Shakerami, A. Numerical solutions of fuzzy differential equations by extended Runge-Kutta-like formulae of order 4. *Fuzzy Sets Syst.* **2012**, *189*, 74–91. [CrossRef]
27. Bede, B. Note on Numerical solutions of fuzzy differential equations by predictor-corrector method. *Inf. Sci.* **2008**, *178*, 1917–1922. [CrossRef]
28. Fatunla, S.O. Block methods for second order ODEs. *Int. J. Comput. Math.* **1991**, *41*, 55–63. [CrossRef]
29. Nasir, N.; Ibrahim, Z.B.; Suleiman, M.B.; Othman, K.I. Fifth order two-point block backward differentiation formulas for solving ordinary differential equations. *Appl. Math. Sci.* **2011**, *5*, 3505–3519.
30. Pallingkinis, S.C.; Papageorgiou, G.; Famelis, I.T. Runge-Kutta methods for fuzzy differential equations. *Appl. Math. Comput.* **2009**, *209*, 97–105. [CrossRef]
31. Ghanbari, M. Numerical solution of fuzzy initial value problems under generalized differentiability by HPM. *Int. J. Ind. Math.* **2009**, *1*, 19–39.
32. Dizicheh, A.K.; Salahshour, S.; Ismail, F.B. A note on "Numerical solutions of fuzzy differential equations by extended Runge-Kutta-like formulae of order 4". *Fuzzy Sets Syst.* **2013**, *233*, 96–100. [CrossRef]
33. Abbasbandy, S.; Allahviranloo, T. Numerical solution of fuzzy differential equation by Runge-Kutta method. *Nonlinear Stud.* **2004**, *11*, 117–129. [CrossRef]
34. Effati, S.; Pakdaman, M. Artificial neural network approach for solving fuzzy differential equations. *Inf. Sci.* **2010**, *180*, 1434–1457. [CrossRef]

© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).