J. Appl. Comput. Mech., xx(x) (2021) 1-11
DOI: 10.22055/JACM.2021.38381.3212

ISSN: 2383-4536
jacm.scu.ac.ir

Shahid Chamran University of Ahvaz

Journal of Applied and Computational Mechanics

Research Paper

Numerical Simulation of Fuzzy Volterra Integro-differential Equation using Improved Runge-Kutta Method

Faranak Rabiei[1], Fatin Abd Hamid[2], Mohammad Mehdi Rashidi[3,4], Zeeshan Ali[1], Kamal Shah[5], Kamyar Hosseini[6], Touraj Khodadadi[7]

[1] School of Engineering, Monash University Malaysia, 47500, Selangor, Malaysia,
Emails: faranak.rabiei@monash.edu, faranak.rabiei@gmail.com (F.R), zeeshan.ali@monash.edu (Z.A)

[2] Institute of Mathematical Research, Universiti Putra Malaysia, 43400, Selangor, Malaysia; Email: fatinkd92@yahoo.com

[3] Institute of Fundamental and Frontier Sciences, University of Electronic Science and Technology of China, Chengdu, 610054, Sichuan, PR China; Email: mm_rashidi@yahoo.com

[4] Faculty of Mechanical and Industrial Engineering, Quchan University of Technology, Quchan, Iran, Email: mm_rashidi@tongji.edu.cn

[5] Department of Mathematics, University of Malakand, Dir(L), 18000, Khyber Pakhtunkhwa, Pakistan, Email: kamalshah408@gmail.com

[6] Department of Mathematics, Rasht Branch, Islamic Azad University, Rasht, Iran; Email: kamyar_hosseini@yahoo.com

[7] Department of Information Technology, Malaysia University of Science and Technology, 47810, Selangor, Malaysia; Email: tj.kh.it@gmail.com

Received August 27 2021; Revised November 08 2021; Accepted for publication November 09 2021.
Corresponding author: M.M. Rashidi (mm_rashidi@yahoo.com)

Abstract. In this research, fourth-order Improved Runge-Kutta method with three stages for solving fuzzy Volterra integro-differential (FVID) equations of the second kind under the concept of generalized Hukuhara differentiability is proposed. The advantage of the proposed method in this study compared with the same order classic Runge-Kutta method is, Improved Runge-Kutta (IRK) method uses a fewer number of stages in each step which causes less computational cost in total. Here, the integral part is approximated by applying the combination of Lagrange interpolation polynomials and Simpson's rule. The numerical results are compared with some existing methods such as the fourth-order Runge-Kutta (RK) method, variational iteration method (VIM), and homotopy perturbation method (HPM) to prove the efficiency of IRK method. Based on the obtained results, it is clear that the fourth-order Improved Runge-Kutta method with higher accuracy and less number of stages which leads the less computational cost is more efficient than other existing methods for solving FVID equations.

Keywords: Improved Runge-Kutta method, Fuzzy differential equations, Fuzzy Volterra integro-differential equations, Generalized Hukuhara differentiability.

1. Introduction

The study of fuzzy differential equations (FDEs) and fuzzy integral equations (FIEs) have attracted numerous attention in recent years since they serve an essential part in solving real problems involving uncertainty and randomness. Mathematical models based on FDEs and FIEs can be found abundantly in science, physics, engineering, and others. It was discovered that to deal with FDEs several approaches were proposed, such as the Hukuhara derivative and differential inclusions. However, both of them showed some disadvantages where the Hukuhara derivative portrays a solution that is fuzzier as time increases while the differential inclusions do not include a fuzzy differential operator (see [1, 2]). To overcome this drawback, Bede and Gal in [3, 4] developed the generalized Hukuhara differentiability. This concept of differentiability motivated the emergence of many numerical methods in finding the solutions to FDEs such as the Euler method by Ma et al. in [5], Runge-Kutta method by Abbasbandy and Viranloo in [6], and Ghanaie and Moghadam in [7], extended Runge-Kutta-like method by Ghazanfari and Shakerami in [8], predictor and corrector method by Allahviranloo et al. in [9] and many others. Riemann integral type approach by Goetschel and Voxman is preferred by many researchers besides the Lebesgue-type concept by Kaleva for the integration of fuzzy functions (see [10, 11]) for solving FIEs. Recently, the General Linear Method is proposed by Rabiei et al. in [12] for numerical solutions of FDEs using the B-series and generalized Hukuhara differentiability. In this work, we focus on solving the FVID equations. Hajighasemi in [13] and Zeinali in [14] studied the existence and uniqueness of solutions of the aforesaid equations. Recent works on solving FVID equations are found by using the Expansion method presented by Allahviranloo et al. in [15] and the variational iteration method by Matinfar et al. in [10].

Published online: November 15 2021

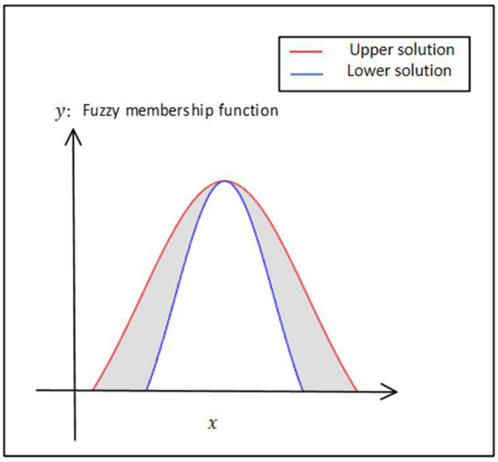


Fig. 1. Schematic configuration of fuzzy membership function.

Here, we consider the Improved Runge-Kutta (IRK) method for finding the solutions of FVID equations that was introduced by Rabiei et al. in [17, 18]. IRK method holds a unique advantage, which is having less computational cost with higher error accuracy. The less computational cost arises from, the lower number of stages per step. As a result, a lower number of function evaluations are needed to compute compared with the classical Runge-Kutta method. Aside from that, IRK method is known as a special two-step method. This project aims to solve the FVID equations using the fourth-order fuzzy IRK method with three stages derived in [17] based on the generalized Hukuhara differentiability. In the integration part of FVID equations, we will consider the Lagrange interpolation polynomials and Simpson's rule. Because the IRK method is not a self-starting method perhaps, one step trapezoidal technique with the classical RK method of order 4 is used to calculate the required starting point.

The rest of this paper is organized as follows: In Section 2 of this paper, some introductions on the fuzzy set theory are presented. A brief review of the FVID equations and the conversion of FVID equations into a system of Volterra integro-differential (VID) equations is given in Section 3. The general form of IRK method is presented in Section 4. We develop the fuzzy IRK for solving FVID equations in Section 5. In section 6, two test problems are performed by derived methods in sections 4 and 5. Lastly, in section 7, a discussion and conclusion are provided. Figure 1 shows the schematic configuration of fuzzy membership function which is used in this research.

2. Background Material

Here, a brief introduction to fuzzy set theory and basic definitions are presented.

Definition 2.1. *[see [4]] A fuzzy subset of the real line* $\mathrm{p}: \mathcal{R} \to [0,1]$ *is considered, where* p *is a fuzzy number satisfying the properties given below:*

(i) p *is normal, (i.e. exist* $\mathcal{V}_0 \in \mathcal{R}$ *with* $\vartheta(\mathcal{V}_0) = 1$);

(ii) p *is fuzzy convex (i.e.* $\mathrm{p}(\mathrm{t}\mathcal{V} - (1-\mathrm{t})\mathcal{U}) \geq \min\{\mathrm{p}(\mathcal{U}), \mathrm{p}(\mathcal{U})\}, \forall \mathrm{t} \in [0,1], \mathcal{V}, \mathcal{U} \in \mathcal{R}$);

(iii) p *is upper semicontinous on* $\mathcal{R}$ *(i.e.* $\forall \epsilon > 0$ *exists* $\delta > 0$ *such that* $|\mathrm{p}(\mathcal{V}) - \mathrm{p}(\mathcal{V}_0)| < \epsilon$ *for* $|\mathcal{V} - \mathcal{V}_0| < \delta$);

(iv) p *is compactly supported (i.e.,* $cl\{\mathcal{V} \in \mathcal{R}; \vartheta(\mathcal{V}) > 0\}$ *is compact, where* $cl(\mathcal{A})$ *is called the closure of the set* $\mathcal{A}$).

$\mathcal{R}_{\mathcal{F}}$ *is expressed as the fuzzy numbers space.*

Definition 2.2. *[see [4]] A fuzzy number* p *is determined by any pair* $\mathrm{p} = (\underline{\mathrm{p}}, \overline{\mathrm{p}})$ *of function* $\underline{\mathrm{p}}, \overline{\mathrm{p}}: [0,1] \to \mathcal{R}$ *defining the end-points of* r*-cuts, given below three conditions:*

(i) $\underline{\mathrm{p}}(\mathrm{r})$ *is a non-decreasing left-continuous function bounded monotonically for* $0 \leq \mathrm{r} \leq 1$ *and for* $\mathrm{r} = 0$*, it is right-continuous;*

(ii) $\overline{\mathrm{p}}(\mathrm{r})$ *is a non-increasing left-continuous function bounded monotonically for* $0 \leq \mathrm{r} \leq 1$ *and for* $\mathrm{r} = 0$*, it is right-continuous;*

(iii) $\underline{\mathrm{p}}(\mathrm{r}) \leq \overline{\mathrm{p}}(\mathrm{r})$ *for* $0 \leq \mathrm{r} \leq 1$.

For arbitrary $\mathrm{p}, \mathrm{q} \in \mathcal{R}_{\mathcal{F}}$ *and* $k \geq 0$*, the sum* $\mathrm{p}+\mathrm{q}$ *and scalar multiplication* $k\mathcal{Y}$ *are defined by* $[\mathrm{p}+\mathrm{q}]_{\mathrm{r}} = [\mathrm{p}]_{\mathrm{r}} + [\mathrm{q}]_{\mathrm{r}}$ *and* $[k\mathcal{Y}]_{\mathrm{r}} = k[\mathcal{Y}]_{\mathrm{r}}$ *while* $[0]_{\mathrm{r}} = \{0\}, \forall \mathrm{r} \in [0,1]$.

Definition 2.3. *[see [4]] The Hausdorff distance between* p *and* q *in* $\mathcal{R}_{\mathcal{F}}$ *is defined as*

$$\mathcal{D}(\mathrm{p},\mathrm{q}) = \sup_{\mathrm{r}\in[0,1]} \max\{|\underline{\mathrm{p}}(\mathrm{r}) - \underline{\mathrm{q}}(\mathrm{r})|, |\overline{\mathrm{p}}(\mathrm{r}) - \overline{\mathrm{q}}(\mathrm{r})|\},$$

where $(\mathcal{R}_{\mathcal{F}}, \mathcal{D})$ *is a complete metric space with the given properties:*

(i) $\mathcal{D}(\mathrm{p}+\ell, \mathrm{q}+\ell) = \mathcal{D}(\mathrm{p},\mathrm{q}), \forall \mathrm{p}, \mathrm{q}, \ell \in \mathcal{R}_{\mathcal{F}}$,

(ii) $\mathcal{D}(k.\mathrm{p}, k.\mathrm{q}) = |k|\mathcal{D}(\mathrm{p},\mathrm{q}), \forall \mathrm{p}, \mathrm{q}, \ell \in \mathcal{R}_{\mathcal{F}}, k \in \mathcal{R}$,

(iii) $\mathcal{D}(\mathrm{p}+\mathrm{q}, \ell + \mathcal{W}) = \mathcal{D}(\mathrm{p}, \ell) + \mathcal{D}(\mathrm{q}, \mathcal{W}), \forall \mathrm{p}, \mathrm{q}, \ell, \mathcal{W} \in \mathcal{R}_{\mathcal{F}}$.

Definition 2.4. *[see [18]] A function* $\chi: \mathcal{R} \to \mathcal{R}_{\mathcal{F}}$ *is called fuzzy continuous if* χ *exists for any fixed* $g_0 \in \mathcal{R}$ *and* $\epsilon > 0, \delta > 0$ *such that* $|g - g_0| < \delta \Longrightarrow \mathcal{D}|\chi(g), \chi(g_0)| < \epsilon$.

Definition 2.5. *[see [18]] For* $s, t \in \mathcal{R}_{\mathcal{F}}$ *if there is* $\mathcal{W} \in \mathcal{R}_{\mathcal{F}}$ *such that* $s = t \oplus \mathcal{W}$*, then* $\mathcal{W}$ *is said Hukuhara difference (H-difference) of* s *and* t *and defined as* $s \ominus t$. $(s \ominus t \neq (s + (-t)))$.

Definition 2.6. *[see [3, 4]] Consider* $\chi: [a,b] \to \mathcal{R}_{\mathcal{F}}$ *for any* $t_0 \in [a,b]$, χ *is strongly generalized Hukuhara differentiable at* t_0 *if there is element* $\chi'(t_0) \in \mathcal{R}_{\mathcal{F}}$*, such that*

(i) *for all* $h > 0$ *sufficiently small,* $\exists \chi(t_0 + h) \ominus_H \chi(t_0)$, $\chi(t_0) \ominus_H \chi(t_0 - h)$ *and the limits (in* $\mathcal{D}$)

$$\lim_{h\searrow 0} \frac{\chi(t_0+h) \ominus_H \chi(t_0)}{h} = \lim_{h\searrow 0} \frac{\chi(t_0) \ominus_H \chi(t_0-h)}{h} = \chi'_G(t_0), \tag{1}$$

is type (i)-differentiability on (a, b).

(ii) *for all* $h > 0$ *sufficiently small,* $\exists \chi(t_0 + h) \ominus_H \chi(t_0)$, $\chi(t_0) \ominus_H \chi(t_0 - h)$ *and the limits*

$$\lim_{h\searrow 0}\frac{\chi(t_0)\ominus_H \chi(t_0+h)}{-h} = \lim_{h\searrow 0}\frac{\chi(t_0-h)\ominus_H \chi(t_0)}{-h} = \chi'_G(t_0), \tag{2}$$

is type (ii)-differentiability on (a, b).

Theorem 2.1 (*see [4]*). *Consider a function* $\chi: [a,b] \to \mathcal{R_F}$ *and* $[\chi(t)]^{\mathrm{r}} = [\underline{\chi}_{\mathrm{r}}(t), \overline{\chi_{\mathrm{r}}}(t)]$, *for every* $0 \le \mathrm{r} \le 1$, *then*

(i) *if* χ *is differentiable in the first form of Definition 2.6 given in equation (1), then the functions* $\underline{\chi}_{\mathrm{r}}$ *and* $\overline{\chi_{\mathrm{r}}}$ *are differential and* $[\chi'(t)]^{\mathrm{r}} = [\underline{\chi'}_{\mathrm{r}}(t), \overline{\chi'}_{\mathrm{r}}(t)]$.

(ii) *if* χ *is differentiable in the second form of Definition 2.6 given in equation (2), then the functions* $\underline{\chi}_{\mathrm{r}}$ *and* $\overline{\chi_{\mathrm{r}}}$ *are differential and* $[\chi'(t)]^{\mathrm{r}} = [\overline{\chi'}_{\mathrm{r}}(t), \underline{\chi'}_{\mathrm{r}}(t)]$.

Definition 2.7. *[see [10]] The Riemann integral concept is used to define the integral of a fuzzy function. Let* $\chi: [\mathcal{U}, \mathcal{Z}] \to \mathcal{R_F}$, *for every partition* $P = \{t_0, t_1, \dots, t_n\}$ *of* $[\mathcal{U}, \mathcal{Z}]$ *and for arbitrary* $\xi_m = [t_{m-1}, t_m], 1 \le m \le n$, *and suppose*

$$\begin{cases} R_p = \sum_{m=1}^{n} (\xi_m)(t_m - t_{m-1}), \\ \Delta := \max\{|t_m - t_{m-1}|, 1 \le m \le n\}. \end{cases}$$

The integration of $\chi(t)$ *over* $[\mathcal{U}, \mathcal{Z}]$ *is*

$$\int_{\mathcal{U}}^{\mathcal{Z}} \chi(t)dt = \lim_{\Delta\to 0} R_p, \tag{3}$$

the limit exists in $\mathcal{D}$. *If the* $\chi(t)$ *is continuous fuzzy function in* $\mathcal{D}$, *its definite integral exists and*

$$\begin{cases} \underline{\int_{\mathcal{U}}^{\mathcal{Z}} \chi(t;\mathrm{r})dt} = \int_{\mathcal{U}}^{\mathcal{Z}} \underline{\chi}(t;\mathrm{r})dt \\ \overline{\int_{\mathcal{U}}^{\mathcal{Z}} \chi(t;\mathrm{r})dt} = \int_{\mathcal{U}}^{\mathcal{Z}} \overline{\chi}(t;\mathrm{r})dt \end{cases} \tag{4}$$

3. FVID Equations

Consider the VID equation given by (see [19])

$$\mathcal{U}'(t) = \mathcal{U}(t) + \lambda \int_0^T \mathcal{K}(t,s)\mathcal{U}(s)ds,\ 0 \le t \le T, \tag{5}$$

where $\lambda > 0$ and the function $\mathcal{K}(t,s)$ is an arbitrary kernel, $\Delta = \{(t,s): 0 \le s \le t \le T\}$ and $\mathcal{U}: [0,T] \times \mathcal{R} \to \mathcal{R}$. The above equation represents function in the crisp case. In short notation, the above function is expressed below:

$$\mathcal{U}'(t) = \chi(\mathcal{U}(t), \lambda \int_0^T \mathcal{K}(t,s)\mathcal{U}(s)ds),\ 0 \le t \le T, \tag{6}$$

In this study, the FVID equation in the form of

$$\mathcal{U}'(t;\mathrm{r}) = \mathcal{U}(t;\mathrm{r}) + \lambda \int_0^T \mathcal{K}(t,s)\mathcal{U}(s;\mathrm{r})ds,\ 0 \le t \le T, \tag{7}$$

is considered and it is shortened as

$$\mathcal{U}'(t;\mathrm{r}) = \chi(\mathcal{U}(t;\mathrm{r}), \lambda \int_0^T \mathcal{K}(t,s)\mathcal{U}(s;\mathrm{r})ds),\ 0 \le t \le T, \tag{8}$$

where $\lambda > 0$ and the function $\mathcal{K}(t,s)$ is an arbitrary kernel, $\Delta = \{(t,s): 0 \le s \le t \le T\}$ and $\mathcal{U}(t;\mathrm{r}): [0,T] \times \mathcal{R_F} \to \mathcal{R_F}$. Since $\mathcal{U}(t;\mathrm{r})$ is a fuzzy function then equation (7) can possess a fuzzy solution. The FVID equation is converted into a system of VID equations and its parametric forms are given as below.

Consider the parametric form $\chi(t)$ be $[\underline{\chi}(t;\mathrm{r}),\overline{\chi}(t;\mathrm{r})]$, $\mathcal{U}(t)$ be $[\underline{\mathcal{U}}(t;\mathrm{r}),\overline{\mathcal{U}}(t;\mathrm{r})]$ and $\mathcal{U}'(t)$ be $[\underline{\mathcal{U}'}(t;\mathrm{r}),\overline{\mathcal{U}'}(t;\mathrm{r})]$ where $0 \le \mathrm{r} \le 1$ and $t \in [a, b]$. From Theorem 2.7 given in Section 2, using type-(i) differentiability the parametric form of the FVID equation is as follows (see [10]):

$$\begin{cases} \underline{\mathcal{U}'}(t;\mathrm{r}) = \underline{\chi}\left(\underline{\mathcal{U}}(t;\mathrm{r}), \lambda \int_0^T \underline{\mathcal{K}(t,s)\mathcal{U}(s;\mathrm{r})}ds\right), \\ \overline{\mathcal{U}'}(t;\mathrm{r}) = \overline{\chi}\left(\overline{\mathcal{U}}(t;\mathrm{r}), \lambda \int_0^T \overline{\mathcal{K}(t,s)\mathcal{U}(s;\mathrm{r})}ds\right), \end{cases} \tag{9}$$

while using type-(ii) differentiability the parametric form of FVIDE is given as

$$\begin{cases} \underline{\mathcal{U}'}(t;\mathrm{r}) = \overline{\chi}\left(\overline{\mathcal{U}}(t;\mathrm{r}), \lambda \int_0^T \overline{\mathcal{K}(t,s)\mathcal{U}(s;\mathrm{r})}ds\right), \\ \overline{\mathcal{U}'}(t;\mathrm{r}) = \underline{\chi}\left(\underline{\mathcal{U}}(t;\mathrm{r}), \lambda \int_0^T \underline{\mathcal{K}(t,s)\mathcal{U}(s;\mathrm{r})}ds\right), \end{cases} \tag{10}$$

where

$$\int_0^T \underline{\mathcal{K}(t,s)\mathcal{U}(s;\mathrm{r})ds} = \begin{cases} \mathcal{K}(t,s)\underline{\mathcal{U}}(s;\mathrm{r}), & \mathcal{K}(t,s) \ge 0, \\ \mathcal{K}(t,s)\overline{\mathcal{U}}(s;\mathrm{r}), & \mathcal{K}(t,s) < 0, \end{cases}$$

and

$$\int_0^T \overline{\mathcal{K}(t,s)\mathcal{U}(s;\mathrm{r})ds} = \begin{cases} \mathcal{K}(t,s)\overline{\mathcal{U}}(s;\mathrm{r}), & \mathcal{K}(t,s) \ge 0, \\ \mathcal{K}(t,s)\underline{\mathcal{U}}(s;\mathrm{r}), & \mathcal{K}(t,s) < 0, \end{cases}$$

for each $0 \le \mathrm{r} \le 1$ and $t \in [a, b]$.

4. Improved Runge-Kutta Method

We consider the first-order initial value problem given by

$$\mathcal{U}'(t) = \chi(t, \mathcal{U}(t)), \qquad \mathcal{U}(t_0) = \mathcal{U}_0 \tag{11}$$

The general form of IRK method to solve (12) is given by Rabiei and Ismail in [16] as follows:

$$\mathcal{U}_{n+1} = \mathcal{U}_n + h\left(b_1 k_1 + b_{-1}k_{-1} + \sum_{m=2}^{s} b_m(k_m - k_{-m})\right), \qquad 1 \le \mathrm{n} \le N-1, \tag{12}$$

where

$$\begin{cases} k_1 = \chi(t_n, \mathcal{U}_n), \\ k_{-1} = \chi(t_{n-1}, \mathcal{U}_{n-1}), \\ k_m = \chi\left(t_n + c_m h, \mathcal{U}_n + \sum_{j=1}^{m-1} a_{mj}k_j\right), & 2 \le \mathrm{m} \le s, \\ k_{-m} = \chi\left(t_{n-1} + c_m h, \mathcal{U}_{n-1} + \sum_{j=1}^{m-1} a_{mj}k_{-j}\right), & 2 \le \mathrm{m} \le s, \end{cases} \tag{13}$$

Table 1. General coefficients of IRK4

0				
c_2	a_{21}			
⋮	⋮	⋱		
c_s	a_{s1}	⋯	a_{ss-1}	
b_{-1}	b_1	⋯	b_{s-1}	b_s

Table 2. Coefficients of IRK4

0			
$\frac{31}{60}$	$\frac{31}{60}$		
$\frac{62}{85}$	$\frac{7502}{24565}$	$\frac{10416}{24565}$	
$\frac{-157}{23064}$	$\frac{23221}{23064}$	$\frac{-1800}{6727}$	$\frac{122825}{161448}$

The general form of coefficients of IRK method is represented in Table 1.

In IRK method, unlike the classical RK method, there are additional internal stages, k_{-m}. The purpose of having extra internal stages is to increase the method's accuracy. In the classical RK4 method, the same strategy is applied to obtain high accuracy by increasing the order of the method and its internal stages. When the order and internal stages of RK method increase the number of functions evaluations increases accordingly, and it causes more computational cost than the lower order method. Although, there are some additional internal stages (k_{-m}) which are approximated from obtained values of k_m in the previous step, IRK method still consists of lower functions evaluations than RK method of the same order.

The derivations of IRK method of order four with three stages are completed by satisfying the required order conditions. These order conditions were achieved through the Taylor series expansion of the general form of IRK method which can be found in Rabiei et al. in [16]. Here, the coefficients used to construct the IRK4 are shown in Table 2.

5. Fuzzy Improved Runge-Kutta Method for FVID Equations

Here, the fuzzy IRK method for solving FVID equations is developed. Lagrange interpolation technique is applied to solve the integral operator in VID equation.

From equation (7) we suppose $\chi: \mathcal{R}_{\mathcal{F}} \times \mathcal{R}_{\mathcal{F}} \to \mathcal{R}_{\mathcal{F}}$ is continuous and $\mathcal{U}_0 \in \mathcal{R}_{\mathcal{F}}$ with r-level sets

$$\mathcal{U}(0;\mathrm{r}) = \left[\underline{\mathcal{U}}(0;\mathrm{r}), \overline{\mathcal{U}}(0;\mathrm{r})\right], \qquad 0 \le \mathrm{r} \le 1 \tag{14}$$

The interval $[0,T]$ is equally divided with grid points $t_0 < t_1 \cdots < t_n = T$. Here, the exact solution is given by $\mathcal{S}(t;\mathrm{r}) = \left[\underline{\mathcal{S}}(t;\mathrm{r}), \overline{\mathcal{S}}(t;\mathrm{r})\right]$, and is approximated by $\mathcal{U}(t;\mathrm{r}) = \left[\underline{\mathcal{U}}(t;\mathrm{r}), \overline{\mathcal{U}}(t;\mathrm{r})\right]$. The solutions are estimated at grid points using $h = \frac{T-t_0}{N}$ and $t_n = t_0 + nh$ where $n < t_0 < N$.

The fuzzy IRK Method with three stages, $s = 3$ using type-(i) differentiability is given by

$$\begin{cases} \underline{\mathcal{U}}(t_{n+1};\mathrm{r}) = \underline{\mathcal{U}}(t_n;\mathrm{r}) + h\Big(\mathrm{b}_1 \underline{k_1}(t_n, \mathcal{U}(t_n;\mathrm{r})) - \mathrm{b}_{-1}\underline{k_{-1}}(t_{n-1}, \mathcal{U}(t_{n-1};\mathrm{r}))\Big) \\ \qquad + \sum_{m=2}^{s} \mathrm{b}_m \left\{\underline{k_m}(t_n, \mathcal{U}(t_n;\mathrm{r})) - \underline{k_{m-1}}(t_n, \mathcal{U}(t_n;\mathrm{r}))\right\}, \\ \overline{\mathcal{U}}(t_{n+1};\mathrm{r}) = \overline{\mathcal{U}}(t_n;\mathrm{r}) + h\Big(\mathrm{b}_1 \overline{k_1}(t_n, \mathcal{U}(t_n;\mathrm{r})) - \mathrm{b}_{-1}\overline{k_{-1}}(t_{n-1}, \mathcal{U}(t_{n-1};\mathrm{r}))\Big) \\ \qquad + \sum_{m=2}^{s} \mathrm{b}_m \left\{\overline{k_m}(t_n, \mathcal{U}(t_n;\mathrm{r})) - \overline{k_{m-1}}(t_n, \mathcal{U}(t_n;\mathrm{r}))\right\}, \end{cases} \tag{15}$$

while for fuzzy IRK Method of order four with three stages, $s = 3$ using type-(ii) differentiability is given by

$$\begin{cases} \underline{\mathcal{U}}(t_{n+1};\mathrm{r}) = \underline{\mathcal{U}}(t_n;\mathrm{r}) + h\Big(\mathrm{b}_1 \overline{k_1}(t_n, \mathcal{U}(t_n;\mathrm{r})) - \mathrm{b}_{-1}\overline{k_{-1}}(t_{n-1}, \mathcal{U}(t_{n-1};\mathrm{r}))\Big) \\ \qquad + \sum_{m=2}^{s} \mathrm{b}_m \left\{\overline{k_m}(t_n, \mathcal{U}(t_n;\mathrm{r})) - \overline{k_{m-1}}(t_n, \mathcal{U}(t_n;\mathrm{r}))\right\}, \\ \overline{\mathcal{U}}(t_{n+1};\mathrm{r}) = \overline{\mathcal{U}}(t_n;\mathrm{r}) + h\Big(\mathrm{b}_1 \underline{k_1}(t_n, \mathcal{U}(t_n;\mathrm{r})) - \mathrm{b}_{-1}\underline{k_{-1}}(t_{n-1}, \mathcal{U}(t_{n-1};\mathrm{r}))\Big) \\ \qquad + \sum_{m=2}^{s} \mathrm{b}_m \left\{\underline{k_m}(t_n, \mathcal{U}(t_n;\mathrm{r})) - \underline{k_{m-1}}(t_n, \mathcal{U}(t_n;\mathrm{r}))\right\}. \end{cases} \tag{16}$$

where

$$\begin{cases} \underline{k_1}(t_n, \mathcal{U}(t_n;\mathrm{r})) = \min\left\{\chi(t_n, \mathcal{Y}) \middle| \mathcal{Y} \in \left[\underline{\mathcal{Z}_1}\left(t_n, \mathcal{U}(t_n;\mathrm{r}), \underline{\mathcal{W}_1}\right), \overline{\mathcal{Z}_1}(t_n, \mathcal{U}(t_n;\mathrm{r}), \overline{\mathcal{W}_1})\right]\right\}, \\ \overline{k_1}(t_n, \mathcal{U}(t_n;\mathrm{r})) = \max\left\{\chi(t_n, \mathcal{Y}) \middle| \mathcal{Y} \in \left[\underline{\mathcal{Z}_1}\left(t_n, \mathcal{U}(t_n;\mathrm{r}), \underline{\mathcal{W}_1}\right), \overline{\mathcal{Z}_1}(t_n, \mathcal{U}(t_n;\mathrm{r}), \overline{\mathcal{W}_1})\right]\right\}, \\ \underline{k_2}(t_n, \mathcal{U}(t_n;\mathrm{r})) = \min\left\{\chi(t_n, \mathcal{Y}) \middle| \mathcal{Y} \in \left[\underline{\mathcal{Z}_2}\left(t_n, \mathcal{U}(t_n;\mathrm{r}), \underline{\mathcal{W}_2}\right), \overline{\mathcal{Z}_2}(t_n, \mathcal{U}(t_n;\mathrm{r}), \overline{\mathcal{W}_2})\right]\right\}, \\ \overline{k_2}(t_n, \mathcal{U}(t_n;\mathrm{r})) = \max\left\{\chi(t_n, \mathcal{Y}) \middle| \mathcal{Y} \in \left[\underline{\mathcal{Z}_2}\left(t_n, \mathcal{U}(t_n;\mathrm{r}), \underline{\mathcal{W}_2}\right), \overline{\mathcal{Z}_2}(t_n, \mathcal{U}(t_n;\mathrm{r}), \overline{\mathcal{W}_2})\right]\right\}, \\ \underline{k_3}(t_n, \mathcal{U}(t_n;\mathrm{r})) = \min\left\{\chi(t_n, \mathcal{Y}) \middle| \mathcal{Y} \in \left[\underline{\mathcal{Z}_3}\left(t_n, \mathcal{U}(t_n;\mathrm{r}), \underline{\mathcal{W}_3}\right), \overline{\mathcal{Z}_3}(t_n, \mathcal{U}(t_n;\mathrm{r}), \overline{\mathcal{W}_3})\right]\right\}, \\ \overline{k_3}(t_n, \mathcal{U}(t_n;\mathrm{r})) = \max\left\{\chi(t_n, \mathcal{Y}) \middle| \mathcal{Y} \in \left[\underline{\mathcal{Z}_3}\left(t_n, \mathcal{U}(t_n;\mathrm{r}), \underline{\mathcal{W}_3}\right), \overline{\mathcal{Z}_3}(t_n, \mathcal{U}(t_n;\mathrm{r}), \overline{\mathcal{W}_3})\right]\right\}, \end{cases} \tag{17}$$

and

$$\begin{cases}\underline{\mathcal{Z}_1}\left(t_n,\mathcal{U}(t_n;\mathrm{r}),\underline{\mathcal{W}_1}\right)=\underline{\mathcal{U}}(t_n;\mathrm{r}),\ \ \underline{\mathcal{W}_1}=\lambda\int_0^{t_n}\underline{\mathcal{K}(t,s)\mathcal{U}(s;\mathrm{r})}ds,\\ \overline{\mathcal{Z}_1}\left(t_n,\mathcal{U}(t_n;\mathrm{r}),\overline{\mathcal{W}_1}\right)=\overline{\mathcal{U}}(t_n;\mathrm{r}),\ \ \overline{\mathcal{W}_1}=\lambda\int_0^{t_n}\overline{\mathcal{K}(t,s)\mathcal{U}(s;\mathrm{r})}ds,\\ \underline{\mathcal{Z}_2}\left(t_n,\mathcal{U}(t_n;\mathrm{r}),\underline{\mathcal{W}_2}\right)=\underline{\mathcal{U}}(t_n;\mathrm{r})+ha_{21}\underline{k_1}(t_n,\mathcal{U}(t_n;\mathrm{r})),\ \ \underline{\mathcal{W}_2}=\lambda\int_0^{t_n+c_2h}\underline{\mathcal{K}(t,s)\mathcal{U}(s;\mathrm{r})}ds,\\ \overline{\mathcal{Z}_2}\left(t_n,\mathcal{U}(t_n;\mathrm{r}),\overline{\mathcal{W}_2}\right)=\overline{\mathcal{U}}(t_n;\mathrm{r})+ha_{21}\overline{k_1}(t_n,\mathcal{U}(t_n;\mathrm{r})),\ \ \overline{\mathcal{W}_2}=\lambda\int_0^{t_n+c_2h}\overline{\mathcal{K}(t,s)\mathcal{U}(s;\mathrm{r})}ds,\\ \underline{\mathcal{Z}_3}\left(t_n,\mathcal{U}(t_n;\mathrm{r}),\underline{\mathcal{W}_3}\right)=\underline{\mathcal{U}}(t_n;\mathrm{r})+h\sum_{j=1}^{2}a_{3j}\underline{k_j}(t_n,\mathcal{U}(t_n;\mathrm{r})),\ \ \underline{\mathcal{W}_3}=\lambda\int_0^{t_n+c_3h}\underline{\mathcal{K}(t,s)\mathcal{U}(s;\mathrm{r})}ds,\\ \overline{\mathcal{Z}_3}\left(t_n,\mathcal{U}(t_n;\mathrm{r}),\overline{\mathcal{W}_3}\right)=\overline{\mathcal{U}}(t_n;\mathrm{r})+h\sum_{j=1}^{2}a_{3j}\overline{k_j}(t_n,\mathcal{U}(t_n;\mathrm{r})),\ \ \overline{\mathcal{W}_3}=\lambda\int_0^{t_n+c_3h}\overline{\mathcal{K}(t,s)\mathcal{U}(s;\mathrm{r})}ds,\end{cases}\tag{18}$$

The fuzzy IRK4 method given in (16)-(18) used the given coefficients in Table 2. The integral operator in (17) and (18), $\mathcal{W}(t)\approx\int_0^t\mathcal{K}(t,s)\mathcal{U}(s)ds,$ is approximated utilizing Lagrange interpolation polynomials and Simpson's II rule (see [19]). We set Simpson's II rule for calculating the integration on interval $[t_0,t_n]$. While on the interval $[t_n,t_{n+ch}]$ we interpolate on $\mathcal{U}_{n-1},\mathcal{U}_n,\mathcal{U}_{ch},$ using Lagrange's formula at points $t=-1,t=0,t=ch$ as shown in (19) and (20).

By applying the integration in internal stages of method on interval $[t_n,t_{n+\frac{31}{60}h}]$, after Lagrange interpolation polynomials are used, we get

$$\int_{t_0}^{t_{n+\frac{31}{60}h}}\mathcal{K}\left(t_{n+\frac{31}{60}h},s\right)\mathcal{U}(s)ds=h\{\frac{-29791}{1965600}\mathcal{K}\left(t_{n+\frac{31}{60}h},t_{-1}\right)\mathcal{U}(t_{-1})+\frac{6541}{21600}\mathcal{K}\left(t_{n+\frac{31}{60}h},t_0\right)\mathcal{U}(t_0)+\frac{3751}{16380}\mathcal{K}\left(t_{n+\frac{31}{60}h},t_{n+\frac{31}{60}h}\right)\mathcal{U}\left(t_{n+\frac{31}{60}h}\right)\},\tag{19}$$

and on the interval $[t_n,t_{n+\frac{62}{85}h}]$, we get

$$\int_{t_0}^{t_{n+\frac{62}{85}h}}\mathcal{K}\left(t_{n+\frac{62}{85}h},s\right)\mathcal{U}(s)ds=h\{\frac{119164}{3186225}\mathcal{K}\left(t_{n+\frac{62}{85}h},t_{-1}\right)\mathcal{U}(t_{-1})+\frac{9827}{21675}\mathcal{K}\left(t_{n+\frac{62}{85}h},t_0\right)\mathcal{U}(t_0)+\frac{11749}{37485}\mathcal{K}\left(t_{n+\frac{62}{85}h},t_{n+\frac{62}{85}h}\right)\mathcal{U}\left(t_{n+\frac{62}{85}h}\right)\}.\tag{20}$$

Note that Lagrange's integration requires value at a point t_{-1} for the first step and it was obtained by using the one-step fourth-order classical RK method.

6. Numerical Results

IRK method using type-(i) differentiability is applied on some test problems to prove the efficiency of the proposed method and obtained results are compared with some existing numerical techniques such as the fourth-order RK method [20], VIM [10], and HPM [10]. The efficiency of the method is shown in terms of maximum global error which is calculated $\overline{e}(t;\mathrm{r})=|\overline{\mathcal{U}}(t;\mathrm{r})-\overline{\mathcal{S}}(t;\mathrm{r})|$ and $\underline{e}(t;\mathrm{r})=|\underline{\mathcal{U}}(t;\mathrm{r})-\underline{\mathcal{S}}(t;\mathrm{r})|$. Here, the list of abbreviations we utilized in the presented results is given as follows:

- r : r-level set of fuzzy number,
- $\overline{\mathcal{S}}$: Upper bound of exact solution,
- $\underline{\mathcal{S}}$: Lower bound of exact solution,
- $\overline{\mathcal{U}}$: Upper bound of approximated solution,
- $\underline{\mathcal{U}}$: Lower bound of approximated solution,
- e^+ : Upper bound of estimated error $(|\mathcal{U}^+-\overline{\mathcal{S}^+}|)$,
- e^- : Upper bound of estimated error $(|\mathcal{U}^--\overline{\mathcal{S}^-}|)$
- TIME : CPU time in seconds,
- s : Number of stages.

Problem 1

Consider the following FVID equation [10]:

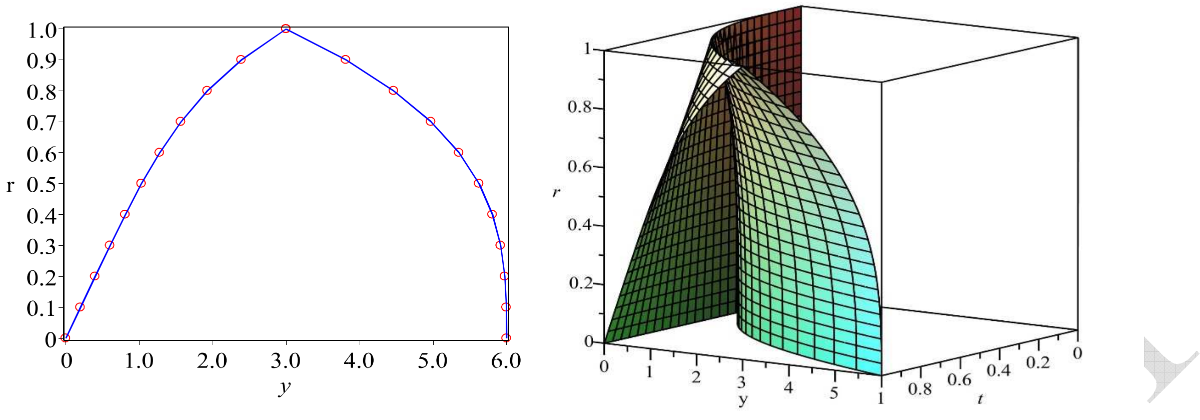
$$\begin{cases}\underline{\mathcal{U}'}(t;\mathrm{r})=\frac{1}{12}\mathrm{r}t^2(\mathrm{r}^4+2)(36-5t^4)+\int_0^t\underline{\mathcal{K}(t,s)\mathcal{U}(s)}ds,\\ \overline{\mathcal{U}'}(t;\mathrm{r})=\frac{1}{4}\mathrm{r}t^2(\mathrm{r}^3-2)(5t^4-36)+\int_0^t\overline{\mathcal{K}(t,s)\mathcal{U}(s)}ds,\end{cases}\tag{21}$$

where the kernel is

$$\mathcal{K}(t,s)=s^2+t^2,\qquad 0\le t\le 1,\ \ 0\le s\le 1.\tag{22}$$

The initial condition of this problem is $\underline{\mathcal{U}}(0;\mathrm{r})=\overline{\mathcal{U}}(0;\mathrm{r})=0$ and the exact solution is given as

$$\underline{\mathcal{U}}(t;\mathrm{r})=(\mathrm{r}^5+2\mathrm{r})t^3,\qquad \overline{\mathcal{U}}(t;\mathrm{r})=(6+3\mathrm{r}^3)t^3.$$



(a) (b)

Fig. 2. (a) Approximate solution of IRK4 (circle) and exact solution (line) at t = 1.0; (b) 3D-plot of IRK4 for Problem 1.

Table 3. Numerical results of problem 1 at t = 0.5

r	IRK4, s = 3		RK4, s = 4		VIM		HPM	
	$\underline{e}(0.5;r)$	TIME	$\underline{e}(0.5;r)$	TIME	$\underline{e}(0.5;r)$	TIME	$\underline{e}(0.5;r)$	TIME
0.0	0		0		0		0	
0.1	2.5060e-11		4.4787e-11		7.6675e-11		2.5125e-6	
0.2	5.0157e-11		8.9641e-11		1.5347e-10		5.0288e-6	
0.3	7.5480e-11		1.3489e-10		2.3095e-10		7.5676e-6	
0.4	1.0151e-10		1.8143e-10		3.1061e-10		1.0178e-5	
0.5	1.2920e-10	3.531	2.3092e-10	4.344	3.9534e-10	-	1.2954e-5	-
0.6	1.6009e-10		2.8612e-10		4.8984e-10		1.6051e-5	
0.7	1.9646e-10		3.5112e-10		6.0113e-10		1.9698e-5	
0.8	2.4152e-10		4.3165e-10		7.3899e-10		2.4215e-5	
0.9	2.9951e-10		5.3529e-10		9.1641e-10		3.0029e-5	
1.0	3.7588e-10		6.7177e-10		9.1641e-10		3.7686e-5	
	$\overline{e}(0.5;r)$	TIME	$\overline{e}(0.5;r)$	TIME	$\overline{e}(0.5;r)$	TIME	$\overline{e}(0.5;r)$	TIME
0.0	7.5175e-10		1.3435e-9		2.3001e-9		7.5371e-5	
0.1	7.5138e-10		1.3428e-9		2.2990e-9		7.5333e-5	
0.2	7.4875e-10		1.3381e-9		2.2909e-9		7.5070e-5	
0.3	7.4161e-10		1.3254e-9		2.2691e-9		7.4354e-5	
0.4	7.2770e-10		1.3005e-9		2.2265e-9		7.2959e-5	
0.5	7.0477e-10	3.531	1.2595e-10	4.344	2.1564e-9	-	7.0660e-5	-
0.6	6.7057e-10		1.1984e-10		2.0517e-9		6.7231e-5	
0.7	6.2283e-10		1.1131e-10		1.9057e-9		6.2444e-5	
0.8	5.5930e-10		9.9960e-10		1.7113e-9		5.6076e-5	
0.9	4.7774e-10		8.5382e-10		1.4617e-9		4.7899e-5	
1.0	3.7588e-10		6.7177e-10		1.1501e-9		3.7686e-5	

Table 4. Numerical results of problem 1 at t = 1.0

r	IRK4	RK4	IRK4, s = 3		RK4, s = 4	
	$\underline{u}$	$\underline{u}$	$\underline{e}(1.0;r)$	TIME	$\underline{e}(1.0;r)$	TIME
0.0	0	0	0		0	
0.1	0.200010000	0.200010000	1.8650e-10		3.7144e-10	
0.2	0.400320000	0.400320001	3.7328e-10		7.4343e-10	
0.3	0.602430001	0.602430001	5.6174e-10		1.1188e-09	
0.4	0.810240001	0.810240001	7.5550e-10		1.5047e-09	
0.5	1.031250001	1.031250002	9.6158e-10	5.422	1.9151e-09	6.797
0.6	1.277760001	1.277760002	1.1914e-09		2.3729e-09	
0.7	1.568070001	1.568070003	1.4621e-09		2.9120e-09	
0.8	1.927680002	1.927680004	1.7975e-09		3.5798e-09	
0.9	2.390490002	2.390490004	2.2290e-09		4.4394e-09	
1.0	3.000000003	3.000000006	2.7974e-09		5.5712e-09	
	$\overline{u}$	$\overline{u}$	$\overline{e}(1.0;r)$	TIME	$\overline{e}(1.0;r)$	TIME
0.0	6.000000006	6.000000011	5.5947e-09		1.1142e-08	
0.1	5.997000006	5.997000011	5.5919e-09		1.1137e-08	
0.2	5.976000006	5.976000011	5.5723e-09		1.1098e-08	
0.3	5.919000005	5.919000011	5.5191e-09		1.0992e-08	
0.4	5.808000005	5.808000011	5.4156e-09		1.0786e-08	
0.5	5.625000005	5.625000010	5.2450e-09	5.422	1.0446e-08	6.797
0.6	5.352000005	5.352000010	4.9905e-09		9.9392e-09	
0.7	4.971000005	4.971000009	4.6352e-09		9.2315e-09	
0.8	4.464000004	4.464000008	4.1625e-09		8.2900e-09	
0.9	3.813000003	3.813000007	3.5554e-09		7.0811e-09	
1.0	3.000000003	3.000000006	2.7974e-09		5.5712e-09	

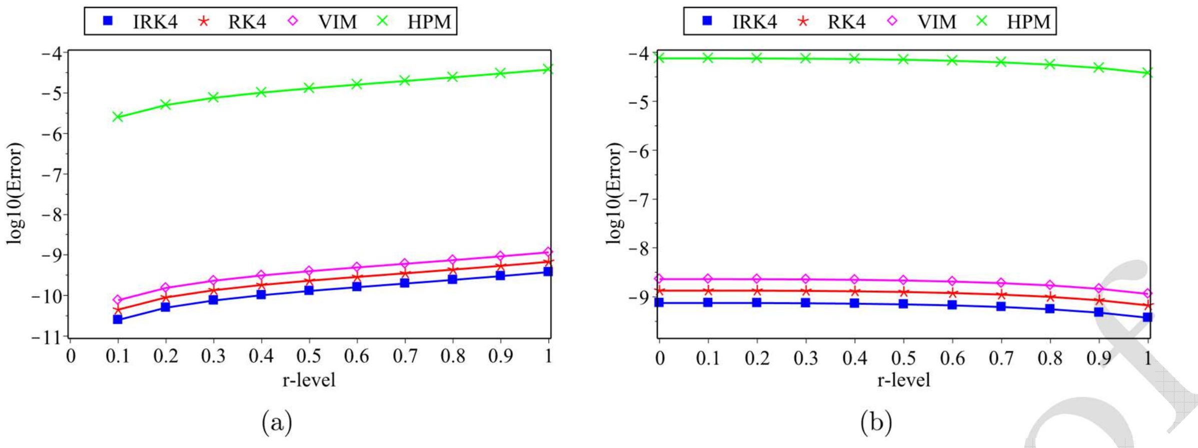


Fig. 3. (a) Error versus r at t = 0.5 for lower bound of solutions; (b) Error versus r at t = 0.5 for upper bound of solutions for Problem 1.

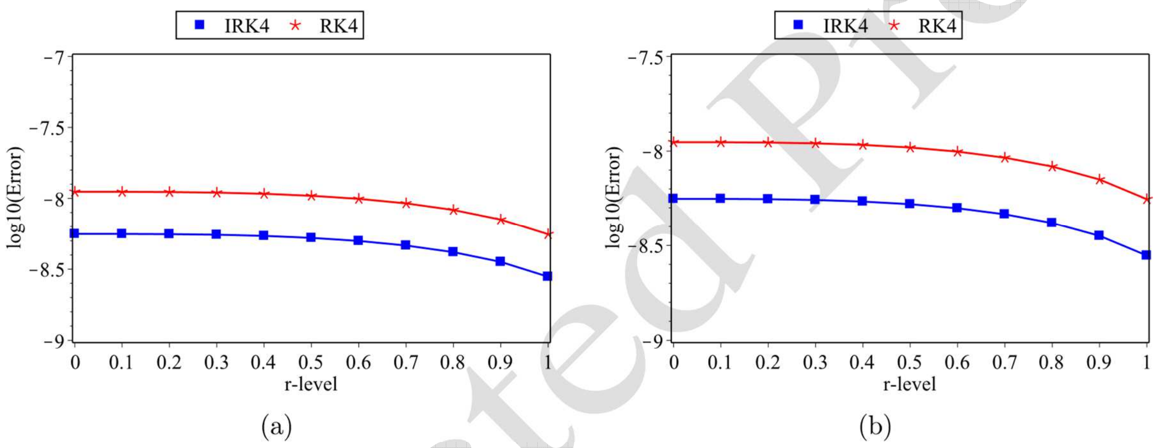


Fig. 4. (a) Error versus r at t = 1.0 for lower bound of solutions; (b) Error versus r at t = 1.0 for upper bound of solutions for Problem 1.

The obtained solutions using the fourth-order IRK method are compared with the exact solution. Also, Figure 2, shows the 3D plot for Problem 1. The result of IRK4 method compared with RK4, VIM and HPM are presented in Tables 3-4 and Figures 3-4. It should be metioned that the comparision of the current method with other analytical methods like HAM, OHAM and DTM is going on [22-25].

Problem 2

Consider the following FVID equation [21]:

$$\begin{cases} \underline{\mathcal{U}}'(t;\mathrm{r}) = \mathrm{r} - 1 + \int_0^t \underline{\mathcal{K}(t,s)\mathcal{U}(s)}ds, \\ \overline{\mathcal{U}}'(t;\mathrm{r}) = \mathrm{r} - 1 + \int_0^t \overline{\mathcal{K}(t,s)\mathcal{U}(s)}ds, \end{cases} \tag{23}$$

where the kernel is

$$\mathcal{K}(t,s) = 1, \qquad 0 \le t \le 1, 0 \le s \le 1. \tag{24}$$

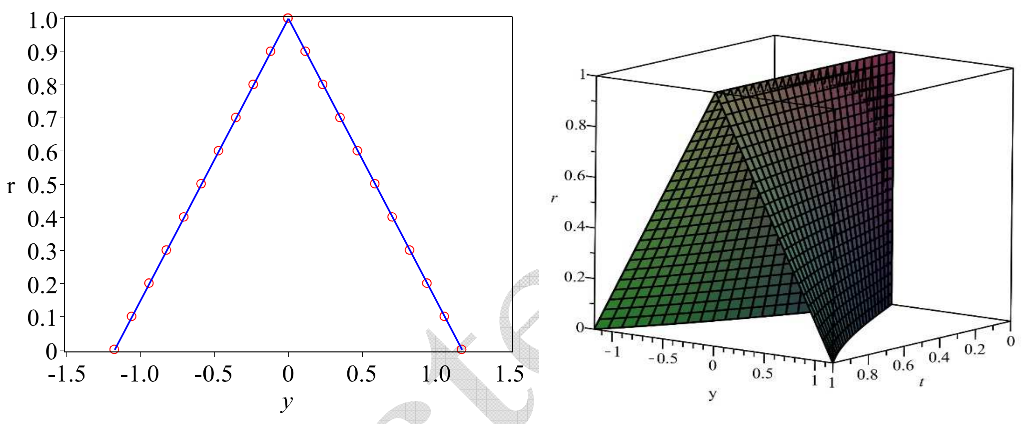
The initial condition of this problem is $\underline{\mathcal{U}}(0;\mathrm{r}) = \overline{\mathcal{U}}(0;\mathrm{r}) = 0$ and the exact solution is given as

$$\underline{\mathcal{U}}(t;\mathrm{r}) = (\mathrm{r} - 1)\sinh(t), \qquad \overline{\mathcal{U}}(t;\mathrm{r}) = (1 - \mathrm{r})\sinh(t).$$

Figure 5 shows the graphical presentation between IRK4 and exact solutions. In Table 5 computed results of IRK4 is compared with RK4.

Table 5. Numerical results of for problem 2 at t = 1.0

r	IRK4	RK4	IRK4, s = 3		RK4, s = 4	
	$\underline{u}$	$\underline{u}$	$\underline{e}(1.0;r)$	TIME	$\underline{e}(1.0;r)$	TIME
0.0	-1.17520119362	-1.17520119369	2.2640e-11		4.8530e-11	
0.1	-1.05768107426	-1.05768107432	2.0370e-11		4.3690e-11	
0.2	-0.94016095490	-0.94016095495	1.8106e-11		3.8822e-11	
0.3	-0.82264083553	-0.82264083558	1.5843e-11		3.3977e-11	
0.4	-0.70512071617	-0.70512071621	1.3584e-11		2.9122e-11	
0.5	-0.58760059681	-0.58760059685	1.1318e-11	4.437	2.4264e-11	6.797
0.6	-0.47008047745	-0.47008047748	9.0540e-12		1.9409e-11	
0.7	-0.35256035809	-0.35256035811	6.7870e-12		1.4562e-11	
0.8	-0.23504023872	-0.23504023874	4.5290e-12		9.7080e-12	
0.9	-0.11752011936	-0.11752011937	2.2640e-12		4.8530e-12	
1.0	0	0	0		0	
	$\overline{u}$	$\overline{u}$	$\overline{e}(1.0;r)$	TIME	$\overline{e}(1.0;r)$	TIME
0.0	1.17520119362	1.17520119369	2.2640e-11		4.8530e-11	
0.1	1.05768107426	1.05768107432	2.0370e-11		4.3690e-11	
0.2	0.94016095490	0.94016095495	1.8106e-11		3.8822e-11	
0.3	0.82264083553	0.82264083558	1.5843e-11		3.3977e-11	
0.4	0.70512071617	0.70512071621	1.3584e-11		2.9122e-11	
0.5	0.58760059681	0.58760059685	1.1318e-11	4.437	2.4264e-11	6.797
0.6	0.47008047745	0.47008047748	9.0540e-12		1.9409e-11	
0.7	0.35256035809	0.35256035811	6.7870e-12		1.4562e-11	
0.8	0.23504023872	0.23504023874	4.5290e-12		9.7080e-12	
0.9	0.11752011936	0.11752011937	2.2640e-12		4.8530e-12	
1.0	0	0	0		0	



(a) (b)

Fig. 5. (a) Approximate solution of IRK4 (circle) and exact solution (line) at t = 1.0; (b) 3D-plot of IRK4 for Problem 2.

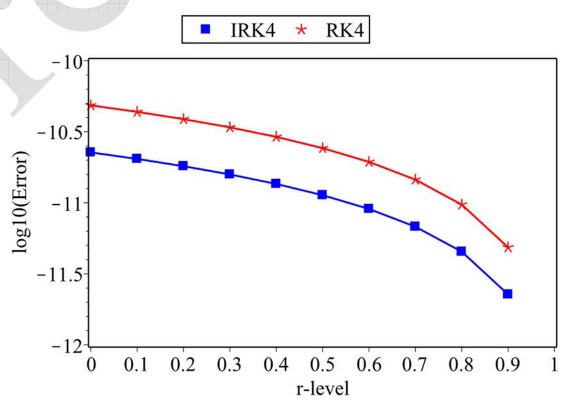


Fig. 6. Error versus r at t = 1.0 for lower and upper bound of solutions for Problem 2.

7. Discussion and Conclusion

In this paper, FVID equations given in problems 1 and 2 are solved using the fuzzy IRK4 and fuzzy RK4. The numerical results are presented and compared with RK4 method, VIM, and HPM for Problem 1 at t = 0.5 in Table 3. The obtained results of IRK4 for problems 1 and 2 are compared with classical RK4 at t = 1 in Tables 4 and 5

In Table 3, the lower bound errors obtained by fuzzy IRK4 are slightly smaller than fuzzy RK4 and VIM. The upper bound solutions given by fuzzy IRK4 yielded better accuracy than fuzzy RK4 and VIM. Also, in terms of CPU time, fuzzy IRK4 performed faster results than fuzzy RK4. This is due to the structure of fuzzy IRK4 having lower stages than fuzzy RK4. We should note that the presented numerical results of VIM and HPM for problem 1 were taken from [10] where the CPU time for those methods was not indicated.

From Table 3, also, it can be seen that fuzzy IRK4 outperformed HPM at all r-levels. In the case of t = 1.0, from Table 4 the accuracy of fuzzy IRK4 is slightly better than fuzzy RK4. Moreover, the upper bound errors obtained by fuzzy IRK4 are one decimal place smaller than fuzzy RK4. Besides, the fuzzy IRK4 has a faster CPU time than the fuzzy RK4. By following the same pattern for problem 2, from Table 5, it is observed that the fuzzy IRK4 performed higher accuracy than fuzzy RK4 by using less CPU time.

Figures 2 and 5 presented the comparison between approximated solutions of fuzzy IRK4 and exact solutions. It can be observed that the approximate solutions of IRK4 have good agreement with the exact solutions and proved the accuracy of IRK4. In addition, 3D plots of fuzzy IRK4 are provided. In Figure 3, the errors obtained by fuzzy IRK4 are competitive with fuzzy RK4 and VIM nevertheless better than HPM. Meanwhile, in Figures 4 and 6, fuzzy IRK4 is more accurate than the fuzzy RK4.

In conclusion, in this paper, the fuzzy IRK for solving FVID equations under the concept of generalized Hukuhara differentiability is proposed. The fuzzy IRK method of order four with three stages by having the fewer computational cost and high accuracy is a more efficient method than the other existing methods such as the fourth-order Runge-Kutta (RK4) method, variational iteration method (VIM), and homotopy perturbation method (HPM) for solving FVID equations.

Author Contributions

F. Rabiei developed the mathematical modeling and examined the theory validation, and; F. Abdul Hamid and Z. Ali conducted the numerical techniques; M.M. Rashidi analyzed the simulated results; K. Shah validated the obtained simulation results. K. Hosseini and T. Khodadadi wrote the original draft. The manuscript was written through the contribution of all authors. All authors discussed the results, reviewed, and approved the final version of the manuscript.

Acknowledgment

This research is studied at the School of Engineering, Monash University Malaysia.

Conflict of Interest

The authors declared no potential conflicts of interest concerning the research, authorship, and publication of this article.

Funding

The authors received no financial support for the research, authorship, and publication of this article.

Data Availability Statements

The datasets generated and/or analyzed during the current study are available from the corresponding author on reasonable request.

References

[1] Akin, O. Khaniyev, T., Oruc, O., Turksen, I.B., Some possible fuzzy solutions for second order fuzzy initial value problems involving forcing terms, *Appl. Comput. Math.*, 13(2), 2014, 239-249.

[2] Abu Arqub, O., Adaptation of reproducing kernel algorithm for solving fuzzy Fredholm-Volterra integrodifferential equations, *Neural. Comput. Applic.*, 28, 2017, 1591–1610.

[3] Bede, B., Gal, S.G., Generalizations of the differentiability of fuzzy number value functions with applications to fuzzy differential equations, *Fuzzy Sets and Systems*, 151(3), 2005, 581-599.

[4] Bede, B., *Mathematics of fuzzy sets and fuzzy logic*, Springer, Berlin, 2013.

[5] Ma, M., Friedman, M., Kandel, A., Numerical solutions of fuzzy differential equations, *Fuzzy Sets and Systems*, 105(1), 1999, 133-138.

[6] Abbasbandy, S., Viranloo, T.A., Numerical solution of fuzzy differential equation by Runge-Kutta method, *J. Sci. Teacher Training University*, 1(3-4), 2001-2002, 31-43.

[7] Ghanaie, Z.A., Moghadam, M.M., Solving fuzzy differential equations by Runge-Kutta method, *Journal of Mathematics and Computer Science*, 2(2), 2011, 208-211.

[8] Ghazanfari, B., Shakerami, A., Numerical solutions of fuzzy differential equations by extended RungeKutta-like formulae of order 4, *Fuzzy Sets and Systems*, 189(1), 2012, 74-91.

[9] Allahviranloo, T., Ahmady, N., Ahmady, E., Numerical solution of fuzzy differential equations by predictor-corrector method, *Information Sciences*, 177(7), 2007, 1633-1647.

[10] Matinfar, M., Ghanbari, M., Nuraei, R., Numerical solution of linear fuzzy Volterra integro-differential equations by variational iteration method, *Intelligence and Fuzzy System*, 24(3), 2013, 575-586.

[11] Allahviranloo, T., Ghanbari, M., Nuraei, R., An application of a semi-analytical method on linear fuzzy Volterra integral equations, *Fuzzy Set Valued Analysis*, 2014, 2014, 1-15.

[12] Rabiei, F., Abd Hamid, F., Rashidi, M. M., Ismail, F., Numerical simulation of fuzzy differential equations using general linear method and B-series, *Advances in Mechanical Engineering*, 9(9), 2017, 1-16.

[13] Hajighasemi, S., Allahviranloo, T., Khazerloo, M., Khorasany, M., Salahshour, S., Existence and uniqueness of solutions of fuzzy Volterra integro-differential equations, *Information Processing and Management of Uncertainty in Knowledge-Based Systems*, 2010, 491-500.

[14] Zeinali, M., Shahmorad, S., Mirnia, K., Fuzzy integro-diffrential equations: discrete solution and error estimation, *Iranian Journal of Fuzzy Systems*, 10(1), 2013, 107-122.

[15] Allahviranloo, T., Abbasbandy, S., Hashemzehi, S., Approximating the solution of the linear and nonlinear fuzzy Volterra integro-differential equations using expansion method, *Abstract and Applied Analysis*, 2014, Article ID 713892.

[16] Rabiei, F., Ismail, F., Suleiman, M., Improved Runge-Kutta methods for solving ordinary differential equations, *Sains Malaysiana*, 42(11), 2013, 107-122.

[17] Rabiei, F., Ismail, F., Ahmadian, A., Salahshour, S., Numerical Solution of Second-Order Fuzzy Differential Equation Using Improved Runge-Kutta Nystrom Method, *Mathematical Problems in Engineering*, 2013, Article ID 803462.

[18] Rabiei, F., Ismail, F., Numerical solution of fuzzy differential equation using improved Runge-Kutta method, *GSTF Journal of Mathematics, Statistics and Operations Research*, 2(2), 2014, 78-83.

[19] Linz, P. *Analytical and Numerical Methods for Volterra Equations*, SIAM, Philadelphia, 1985.

[20] Butcher, J.C., *Numerical Methods or Ordinary Differential Equations*, Second Edition, John Wiley and Sons, 2008.

[21] Mikaelvand, N., Khakrangin, S., Allahviranloo, T., Solving fuzzy Volterra integro-differential equation by fuzzy differential transform method, *Proceedings of the 7th conference of the European Society for Fuzzy Logic and Technology*, August 2011.

[22] Rashidi, M.M., Mohimanian Pour, S.A., Analytic Approximate Solutions for Unsteady Boundary-Layer Flow and Heat Transfer due to a Stretching

Sheet by Homotopy Analysis Method, *Nonlinear Analysis: Modelling and Control*, 15(1), 2010, 83–95.
[23] Sarwar, S., Rashidi, M.M., Approximate Solution of Two-Term Fractional-Order Diffusion, Wave-Diffusion, and Telegraph Models Arising in Mathematical Physics Using Optimal Homotopy Asymptotic Method, *Waves in Random and Complex Media*, 26(3), 2016, 365–382.
[24] Rashidi, M.M., Shooshtari, A., Anwar Bég, O., Homotopy Perturbation Study of Nonlinear Vibration of Von Karman Rectangular Plates, *Computers and Structures*, 106–107, 2012, 46–55.
[25] Rashidi, M.M., Ferdows, M., Uddin, J., Beg, O., Rahimzadeh, N., Group Theory and Differential Transform Analysis of Mixed Convective Heat and Mass Transfer from a Horizontal Surface with Chemical Reaction Effects, *Chemical Engineering Communications*, 199(8), 2012, 1012–1043.

ORCID IDs

F. Rabiei https://orcid.org/ 0000-0002-0648-0477
M.M. Rashidi https://orcid.org/0000-0002-6309-8688
Z. Ali https://orcid.org/0000-0001-6846-9516
K. Shah https://orcid.org/0000-0002-8851-4844
K. Hosseini https://orcid.org/0000-0001-7137-1456
T. Khodadadi https://orcid.org/ 0000-0002-4914-0544

How to cite this article: Rabiei F., Abd Hamid F., Rashidi M.M., Ali Z., Shah, K., Hosseini K., Khodadai T. Numerical Simulation of Fuzzy Volterra Integro-differential Equation using Improved Runge-Kutta Method, *J. Appl. Comput. Mech.*, xx(x), 2021, 1–11. https://doi.org/10.22055/JACM.2021.38381.3212