Malaysian Journal of Mathematical Sciences 15(2): 253–266 (2021)

Malaysian Journal of Mathematical Sciences

Journal homepage: https://einspem.upm.edu.my/journal

Improved Runge-Kutta Method with Trigonometrically-Fitting Technique for Solving Oscillatory Problem

Senu, N. *[1,2], Lee, K. C. [2], Wan Ismail, W. F. [2], Ahmadian, A. [3], Ibrahim, S. N. I. [1,2], and Laham, M. F. [1]

[1]*Institute for Mathematical Research, Universiti Putra Malaysia, Malaysia*
[2]*Department of Mathematics and Statistics, Faculty of Science, Universiti Putra Malaysia, Malaysia*
[3]*Institute of Industry Revolution 4.0, The National University of Malaysia, 43600 UKM Bangi, Selangor, Malaysia.*

E-mail: norazak@upm.edu.my
**Corresponding author*

Received: 23 April 2020
Accepted: 30 April 2021

Abstract

In this article we propose a new method of Trigonometrically-Fitted Improved Runge-Kutta (TFIRK3(3)) with third-order and three stages for solving oscillatory ordinary differential equations. The proposed algorithm employs a derivation of method by adding trigonometric into the Improved Runge-Kutta (IRK3(3)) method. It is found that the new method is more accurate as compared to IRK3(3) and classical Runge-Kutta methods. To illustrate the efficiency of this method, number of initial value problem for the system of first-order ordinary differential equations (ODEs) are solved. The computational experiments show that the TFIRK3(3) method performs better than RK3(3), RK4(4), IRK3(3) and PHSFRK5(4) methods in most cases.

Keywords: Trigonometrically-fitted; ordinary differential equations; improved Runge-Kutta method; quantum cryptography; differential equations foundations; Schrödinger equations; computational experiments.

1 Introduction

This article focuses on refining classical Runge-Kutta method into Improved Runge-Kutta (IRK3(3)) method for solving initial value problems of first order ordinary differential equations (ODEs) of the form,

$$y'(x) = f(x, y(x)), \quad y(x_0) = y_0 \quad x \in [x_0, X], \tag{1}$$

that contains oscillatory character in the solutions. Over a century ago, C. Runge had been working on his research paper about the extended approximation method by Euler. Specifically, [12] dealt with an initial value problem of the form (1) and found three main strategies. The first strategy is the midpoint role adjusted to ODEs. The second and third ones consisting varieties of trapezoidal rule. Furthermore, [4] stated that the analysis of Runge-Kutta method has been categorized into fourth-order method and then expanded up to fifth order by Kutta. Analysis of fifth-order RK method and the approach of extending RK methods into second-order ordinary differential equations has been done by Nyström (see [4]).

In the context of quantum cryptography, quantum memory plays the role as a security device for eavesdropping within quantum key distribution (QKD). It is a fact that applying Schrödinger equations to quantum memory is essential for further theoretical understanding as well as ensuring quantum security within quantum cryptography. The numerical study of Schrödinger equations based on ordinary differential equations system with oscillatory problem has recently captured the attention of many applied mathematicians around the world. In particular, [2] and [3] developed an efficient numerical method based on fitted techniques for solving the Schrödinger equations which is related to oscillatory problems. Moreover, in quantum memory, Schrödinger equations with spatial variables plays an important role in formulating potential scattering which leads to a significant decoherence as shown schematically in [15] and [16]. These are basic tools in any study of quantum memory within quantum field scattering theory. Schrödinger equations can be effectively solved by reducing these equations into a system that consisting of two first order ordinary differential equations with initial value conditions and then implementing proper numerical methods with oscillatory properties to solve it.

In 2011, [9] raised the Improved Runge-Kutta methods and attained the order up to fifth order. More recently, the stability of the methods was discussed as well. Later in 2012, [10] also raised the Improved Runge-Kutta method for solving first and second ODEs by presenting the new terms k_{-i}, that takes the k_i; $i \geq 2$ from its previous steps. Trigonometrically-fitted method is another alternative to formulate new methods for solving oscillatory problems. These methods are the upgraded version of any original method. For example, trigonometrically-fitted techniques can be merged into Runge-Kutta methods, multistep method, predictor-corrector method, block backward method and etc. Moreover, there are also another fitted techniques such as exponentially-fitted and phase-fitted that can be used to solve oscillatory problems. In 2003, [13] had developed exponentially-fitted and trigonometrically-fitted symmetric linear multistep methods for solving second order differential equations. In the same year, [14] also developed exponentially-fitted method for the numerical integration of the Schrödinger equation. Correspondingly, [17] explored the trigonometrically-fitted Scheifele methods for oscillatory problems. These two fitted techniques are very effective in order to solve the oscillatory problems. In 2013, [18] built a new trigonometrically-fitted fifth-order two-derivative Runge-Kutta method with variable nodes for the numerical solution of the radial Schrodinger equation and related oscillatory problems. In 2019, [7] proposed a numerical integrator with sixth order of convergence and taking dispersion and dissipation errors into account for solving oscillatory problems.

The aim of this paper is to derive Trigonometrically-Fitted Improved Runge-Kutta (TFIRK3(3)) with three stages third-order method to solve system of first order ordinary differential equation with oscillatory solutions such as Schrödinger equations and the numerical experiment shows the accuracy of new proposed method compared to other existing methods. In Section 2, derivation of TFIRK method is proposed. Numerical test and comparison with existing methods discussed in Section 3 to illustrate the capabality of TFIRK method. This paper ends in Section 4 with the discussion and conclusion.

2 The Derivation of TFIRK3(3) Method

In this method, the value of y_{n+1} is calculated by using the values of y_n and y_{n-1}. IRK method also introduces new term which is k_{-i} that takes the value $k_i, (i > 2)$ from its previous steps. In this section, we are using IRK3(3) method with three stages (s=3). First of all, consider the initial value problem for the system of first-order ordinary differential equation of the form (1). The IRK method with s-stages for solving equation (1) is as follows: (See Rabiei et al. [11])

$$y_{n+1} = (1-a)y_n + ay_{n-1} + h\left(b_1k_1 - b_{-1}k_{-1} + \sum_{i=2}^{s} b_i(k_i - k_{-i})\right), \tag{2}$$

$$k_1 = f(x_n, y_n), \tag{3}$$

$$k_{-1} = f(x_{n-1}, y_{n-1}), \tag{4}$$

$$k_i = f(x_n + c_ih, y_n + h\sum_{j=1}^{i-1} a_{ij}k_j), \tag{5}$$

$$k_{-i} = f(x_{n-1} + c_ih, y_{n-1} + h\sum_{j=1}^{i-1} a_{ij}k_{-j}), \tag{6}$$

for $c_2, \ldots, c_s \in [0,1]$, f depends on both x and y, k_i and k_{-i} depend on the values of k_j and k_{-j} for $j = 1, \ldots, i-1$.

Equations (2) to (6) can be expressed in Butcher's Tableau as shown in Table 1.

Table 1: Butcher's IRK Parameters

0					
c_2	$a_{2,1}$				
c_3	$a_{3,1}$	$a_{3,2}$			
$\vdots$	$\vdots$	$\vdots$	$\ddots$		
c_s	$a_{s,1}$	$a_{s,2}$	$\cdots$	$a_{s,s-1}$	
b_{-1}	b_1	b_2	$\cdots$	b_{s-1}	b_s

Figure 1 shows the general construction of IRK3(3) method. (See [11])

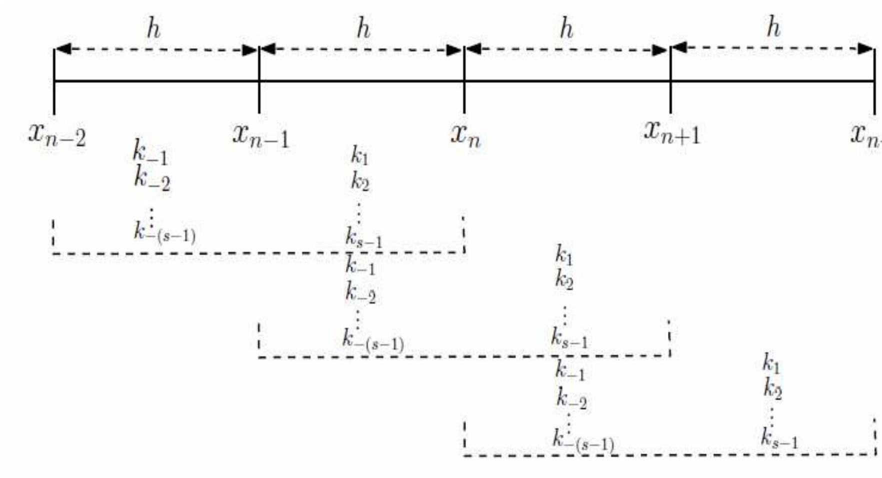

Figure 1: General Construction of Improved Runge-Kutta Method

The general form of IRK3(3) method with three stages is given by,

$$\begin{aligned}
y_{n+1} &= y_n + h(b_1k_1 - b_{-1}k_{-1} + b_2(k_2 - k_{-2}) + b_3(k_3 - k_{-3})), && (7)\\
k_1 &= f(x_n, y_n), && (8)\\
k_{-1} &= f(x_{n-1}, y_{n-1}), && (9)\\
k_2 &= f(x_n + c_2h, y_n + ha_{21}k_1), && (10)\\
k_{-2} &= f(x_{n-1} + c_2h, y_{n-1} + ha_{21}k_{-1}), && (11)\\
k_3 &= f(x_n + c_3h, y_n + h(a_{31}k_1 + a_{32}k_2)), && (12)\\
k_{-3} &= f(x_{n-1} + c_3h, y_{n-1} + h(a_{31}k_{-1} + a_{32}k_{-2})). && (13)
\end{aligned}$$

Next, the exponential function is implemented into IRK3(3) formulas. The trigonometrically fitting techniques are applied by first letting,

$$\begin{aligned}
y_n &= e^{\frac{\mathrm{i}Vx_n}{h}}, && (14)\\
y_{n-1} &= e^{\frac{\mathrm{i}V}{h}(x_n-h)}, && (15)\\
y_{n+1} &= e^{\frac{\mathrm{i}V}{h}(x_n+h)}. && (16)
\end{aligned}$$

Let $e^{\mathrm{i}V} = \cos(V) + \mathrm{i}\sin(V)$ and substituting the equations (14)-(16) into equation (7)-(13), we will have,

$$\begin{aligned}
e^{\mathrm{i}V} &= \cos(V) + \mathrm{i}\sin(V)\\
&= -\mathrm{i}V^3a_{21}a_{32}b_3 + \mathrm{i}e^{-\mathrm{i}V}V^3a_{21}a_{32}b_3 - V^2a_{21}b_2 - V^2a_{31}b_3\\
&\quad -V^2a_{32}b_3 + \mathrm{i}b_1V + \mathrm{i}Vb_2 + \mathrm{i}Vb_3 + e^{-\mathrm{i}V}V^2a_{21}b_2\\
&\quad +e^{-\mathrm{i}V}V^2a_{31}b_3 + e^{-\mathrm{i}V}V^2a_{32}b_3 - \mathrm{i}e^{-\mathrm{i}V}Vb_{-1}\\
&\quad -\mathrm{i}e^{-\mathrm{i}V}Vb_2 - \mathrm{i}e^{-\mathrm{i}V}Vb_3 + 1, && (17)
\end{aligned}$$

where

$$V = \omega h.$$

Then, we separate the real and imaginary part. The equation of $\cos(V)$ is the real part and $\sin(V)$ is the imaginary part.

$$\begin{aligned}\cos(V) = \quad & 1 + \sin(V)V^3 a_{21}a_{32}b_3 - V^2 a_{21}b_2 - V^2 a_{31}b_3 - V^2 a_{32}b_3 \\ & + \cos(V)V^2 a_{21}b_2 + \cos(V)V^2 a_{31}b_3 + cos(V)V^2 a_{32}b_3 - \\ & \sin(V)Vb_{-1} - \sin(V)Vb_2 - \sin(V)Vb_3, \qquad (18)\\ \sin(V) = \quad & -V^3 a_{21}a_{32}b_3 + \cos(V)V^3 a_{21}a_{32}b_3 + b_1 V + Vb_2 + Vb_3 \\ & - \sin(V)V^2 a_{21}b_2 - \sin(V)V^2 a_{31}b_3 - \sin(V)V^2 a_{32}b_3 \\ & - \cos(V)Vb_{-1} - \cos(V)Vb_2 - \cos(V)Vb_3. \qquad (19)\end{aligned}$$

We utilise the method proposed by Rabiei [9] that shown in Table 2, and implement trigonometrically fitted techniques into it.

Table 2: Butcher's IRK Parameters

0			
$\frac{1}{3}$	$\frac{1}{3}$		
$\frac{2}{3}$	$\frac{2}{21}$	$\frac{4}{7}$	
$\frac{1}{8}$	$\frac{9}{8}$	$-\frac{1}{2}$	$\frac{7}{8}$

Before substituting the values from Table 2, there will be nine undetermined coefficients such as $c_2, c_3, a_{21}, a_{31}, a_{32}, b_{-1}, b_1, b_2$ and b_3. Thus, we let c_2, c_3, a_{21}, b_{-1}, b_1, b_2 and b_3 as free parameters where their values are obtained from Table 2. Altogether we have eighteen possible solutions satisfying with the original coefficients. But, the values for a_{31} and a_{32} are chosen. This is because, according to the numerical experiment, this coefficients combination gives the most accurate result for trigonometrically-fitted IRK method. By solving (18) and (19), yields

$$\begin{aligned}a_{31} = \quad & \frac{4}{21}\frac{1}{V^3\left(\sin(V)^2 + \cos(V)^2 - 2\cos(V) + 1\right)}(\sin(V)^2V^3 + \cos(V)^2V^3 \\ & -2\cos(V)V^3 - 15\sin(V)^2V + 6\sin(V)V^2 - 3\cos(V)^2V \\ & +V^3 - 36\cos(V)\sin(V) + 24\cos(V)V + 36\sin(V) - 21V), \qquad (20)\\ a_{32} = \quad & \frac{12}{7}\frac{\sin(V)^2V + \cos(V)^2V + 4\cos(V)\sin(V) - 4\cos(V)V - 4\sin(V) + 3V}{V^3(\sin(V)^2 + \cos(V)^2 - 2\cos(V) + 1)}. \qquad (21)\end{aligned}$$

Taylor series expansions are being used for the coeffiecient a_{31} and a_{32} above because the value of V is always taken as the small value. Thus, we have the Taylor series expansions for these coefficients are given as follows:

$$\begin{aligned}a_{31} = \quad & \frac{2}{21} - \frac{13}{630}V^2 + \frac{23}{26460}V^4 - \frac{1}{50400}V^6 + \frac{43}{209563200}V^8 \\ & -\frac{2791}{1144215072000}V^{10} - \frac{1}{208039104000}V^{12} + O(V^{14}), \qquad (22)\\ a_{32} = \quad & \frac{4}{7} - \frac{1}{35}V^2 + \frac{1}{1470}V^4 - \frac{1}{105840}V^6 + \frac{1}{11642400}V^8 - \frac{1}{1816214400}V^{10} \\ & +\frac{1}{381405024000}V^{12} + O(V^{14}), \qquad (23)\end{aligned}$$

where $V = \omega h$.

3 Problems Tested and Numerical Results

In this section, the new method obtained is implemented into C programming to determine its accuracy in solving oscillatory problems. Five numerical tests have been solved and their maximum global errors have been recorded. From this step, numerical results for intervals $x_n = 100$, $x_n = 1000$ and $x_n = 5000$ are tabulated and compared with the existing methods; RK3(3), RK4(4), IRK3(3) and PHSFRK5(4) methods.

The notations used are as follows:

- h: The step size
- MAXERR: The maximum error obtained from experiments
- RK3(3) : Classical third-order Runge Kutta method by [8]
- RK4(4) : Classical fourth-order Runge-Kutta method by [5]
- IRK3(3) : Improved Runge-Kutta method (proposed by [9])
- PHAFRK5(4) : Phase-fitted Runge-Kutta method (proposed by [6])
- TFIRK3(3) : Trigonometrically-fitted Improved Runge-Kutta method
- $1.755783(-2) : 1.755783 \times 10^{-2}$

The maximum error is defined by

$$MAXERR = max|y(x_n) - y_n|,$$

where $y(x_n)$ is the exact solution and y_n is the approximate solution.

The problems tested are given as follows:

Problem 1 Homogeneous problem

$$\begin{aligned} y_1' &= y_2, \quad y_1(0) = 1, \\ y_2' &= -64y_1, \quad y_2(0) = -2. \end{aligned}$$

Theoretical solution:

$$\begin{aligned} y_1(x) &= -\frac{1}{4}\sin(8x) + \cos(8x), \\ y_2(x) &= -2\cos(8x) - 8\sin(8x). \end{aligned}$$

Source: [6]

Problem 2 Inhomogeneous problem

$$\begin{aligned} q_1' &= q_2, \quad q_1(0) = 1, \\ q_2' &= -100q_1 + 99\sin(x), \quad q_2(0) = 11. \end{aligned}$$

Theoretical solution:

$$\begin{aligned} q_1(x) &= \cos(10x) + \sin(10x) + sin(x), \\ q_2(x) &= -10\sin(10x) + 10\cos(10x) + \cos(x). \end{aligned}$$

Source: [1]

Problem 3 Inhomogeneous problem

$$\begin{aligned} y_1' &= y_2, \quad y_1(0) = 1, \\ y_2' &= -y_1 + x, \quad y_2(0) = 2. \end{aligned}$$

Theoretical solution:

$$\begin{aligned} y_1(x) &= \sin(x) + \cos(x) + x, \\ y_2(x) &= \cos(x) - \sin(x) + 1. \end{aligned}$$

Source: [6]

Problem 4 An "almost" periodic orbit problem

$$\begin{aligned} q_1' &= q_2, \quad q_1(0) = 1, \\ q_2' &= -q_1 + 0.001\cos(x), \quad q_2(0) = 0, \\ q_3' &= q_4, \quad q_3(0) = 0, \\ q_4' &= -q_3 + 0.001\sin(x), \quad q_4(0) = 0.9995. \end{aligned}$$

Theoretical solution:

$$\begin{aligned} q_1(x) &= \cos(x) + 0.0005x\sin(x), \\ q_2(x) &= -\sin(x) + 0.0005x\cos(x) + 0.0005\sin(x), \\ q_3(x) &= \sin(x) - 0.0005x\cos(x), \\ q_4(x) &= \cos(x) + 0.0005x\sin(x) - 0.0005\cos(x). \end{aligned}$$

Source: [6]

Problem 5 Duffing problem

$$\begin{aligned} q_1' &= q_2, \quad q_1(0) = 0.200426728067, \\ q_2' &= -q_1 - q_1^3 + 0.002\cos(1.01x), \quad q_2(0) = 0. \end{aligned}$$

Theoretical solution:

$$\begin{aligned} q_1(x) \quad &= 0.200179477536\cos(1.01x) + 2.46946143 \times 10^{-4}\cos(3.03x) \\ &+3.04014 \times 10^{-7}\cos(5.05x) + 3.74 \times 10^{-10}\cos(7.07x), \\ q_2(x) \quad &= -0.2021812723\sin(1.01x) - 7.482468133 \times 10^{-4}\sin(3.03x) \\ &-1.53527070 \times 10^{-6}\sin(5.05x) - 2.64418 \times 10^{-9}\sin(7.07x). \end{aligned}$$

Source: [1]

Tabulated numerical results obtained from the comparison among five methods; TFIRK3(3), RK3, RK4, IRK3(3) and PHSFRK5(4) are shown in tables below.

Table 3: Comparison between TFIRK3(3) method with existing methods for problem 1

h	**METHODS**	**MAXERR**		
		$x_n = 100$	$x_n = 1000$	$x_n = 5000$
0.01	RK3(3)	1.394186(-1)	1.293230(+0)	4.735657(+0)
	RK4(4)	2.244519(-3)	2.250934(-2)	1.125192(-1)
	IRK3(3)	1.162631(-2)	1.165625(-1)	5.816297(-1)
	PHAFRK5(4)	1.315568(-7)	5.040964(-8)	4.136408(-6)
	TFIRK3(3)	9.299654(-10)	5.041495(-8)	4.136380(-6)
0.005	RK3(3)	1.755783(-2)	1.740365(-1)	8.341912(-1)
	RK4(4)	1.403343(-4)	1.406961(-3)	7.041328(-3)
	IRK3(3)	7.255028(-4)	7.274243(-3)	3.637883(-2)
	PHAFRK5(4)	5.162173(-9)	5.519592(-8)	5.668248(-6)
	TFIRK3(3)	1.200941(-9)	5.519994(-8)	5.668227(-6)
0.0025	RK3(3)	2.196717(-3)	2.195992(-2)	1.092108(-1)
	RK4(4)	8.767422(-6)	8.861223(-5)	4.353489(-4)
	IRK3(3)	4.531957(-5)	4.550967(-4)	2.268064(-3)
	PHAFRK5(4)	3.591340(-9)	6.740343(-7)	5.132166(-6)
	TFIRK3(3)	3.591466(-9)	6.740333(-7)	5.132161(-6)
0.00125	RK3(3)	2.746279(-4)	2.748205(-3)	1.373123(-2)
	RK4(4)	5.433997(-7)	6.553077(-6)	2.057327(-5)
	IRK3(3)	2.827586(-6)	2.945530(-5)	1.351054(-4)
	PHAFRK5(4)	4.786536(-9)	1.056863(-6)	1.905828(-5)
	TFIRK3(3)	4.786705(-9)	1.056860(-6)	1.905829(-5)

Table 4: Comparison between TFIRK3(3) method with existing methods for problem 2

h	**METHODS**	**MAXERR**		
		$x_n = 100$	$x_n = 1000$	$x_n = 5000$
0.01	RK3(3)	5.760447(-1)	4.818550(+0)	1.239631(+1)
	RK4(4)	1.178225(-2)	1.177423(-1)	5.880912(-1)
	IRK3(3)	6.111265(-2)	6.098878(-1)	3.023856(+0)
	PHAFRK5(4)	3.332185(-6)	3.334079(-6)	1.181473(-5)
	TFIRK3(3)	1.185505(-7)	1.316540(-7)	8.965980(-6)
0.005	RK3(3)	7.332561(-2)	7.176175(-1)	3.242427(+0)
	RK4(4)	7.364328(-4)	7.361126(-3)	3.683143(-2)
	IRK3(3)	3.808660(-3)	3.807026(-2)	1.904036(-1)
	PHAFRK5(4)	2.095014(-7)	2.095014(-7)	1.232402(-5)
	TFIRK3(3)	8.740533(-9)	1.191941(-7)	1.215407(-5)
0.0025	RK3(3)	9.187417(-3)	9.176156(-2)	4.529118(-1)
	RK4(4)	4.601817(-5)	4.616719(-4)	2.292256(-3)
	IRK3(3)	2.378494(-4)	2.379896(-3)	1.188573(-2)
	PHAFRK5(4)	1.356308(-8)	1.454460(-6)	1.101093(-5)
	TFIRK3(3)	7.765059(-9)	1.444829(-6)	1.100230(-5)
0.00125	RK3(3)	1.148831(-3)	1.150292(-2)	5.742467(-2)
	RK4(4)	2.866413(-6)	3.102950(-5)	1.290402(-4)
	IRK3(3)	1.485345(-5)	1.508892(-4)	7.284915(-4)
	PHAFRK5(4)	1.035779(-8)	2.265474(-6)	4.085754(-5)
	TFIRK3(3)	1.029836(-8)	2.265489(-6)	4.085750(-5)

Table 5: Comparison between TFIRK3(3) method with existing methods for problem 3

h	METHODS	MAXERR		
		$x_n = 100$	$x_n = 1000$	$x_n = 5000$
0.32	RK3(3)	1.266253(-1)	7.743516(-1)	2.323993(+0)
	RK4(4)	8.628808(-3)	8.873541(-2)	6.845592(-1)
	IRK3(3)	4.608263(-2)	4.592178(-1)	2.922123(+0)
	PHAFRK5(4)	3.474871(-4)	3.474871(-4)	1.198748(-3)
	TFIRK3(3)	3.304926(-5)	3.327421(-4)	1.666644(-3)
0.16	RK3(3)	1.670246(-2)	1.615221(-1)	9.930187(-1)
	RK4(4)	5.393389(-4)	5.610690(-3)	4.367456(-2)
	IRK3(3)	2.808563(-3)	2.924361(-2)	2.287304(-1)
	PHAFRK5(4)	2.182663(-5)	2.182663(-5)	2.182663(-5)
	TFIRK3(3)	2.002977(-6)	2.022489(-5)	1.013254(-4)
0.08	RK3(3)	2.108374(-3)	2.171241(-2)	1.634239(-1)
	RK4(4)	3.377142(-5)	3.511027(-4)	2.728517(-3)
	IRK3(3)	1.746746(-4)	1.817343(-3)	1.413606(-2)
	PHAFRK5(4)	1.364891(-6)	1.364891(-6)	1.364891(-6)
	TFIRK3(3)	1.244537(-7)	1.255653(-6)	6.291697(-6)
0.04	RK3(3)	2.637470(-4)	2.739510(-3)	2.121374(-2)
	RK4(4)	2.110465(-6)	2.194628(-5)	1.706168(-4)
	IRK3(3)	1.089817(-5)	1.133772(-4)	8.818884(-4)
	PHAFRK5(4)	8.232989(-8)	8.537730(-8)	8.537730(-8)
	TFIRK3(3)	7.760707(-9)	7.870927(-8)	3.964782(-7)

Table 6: Comparison between TFIRK3(3) method with existing methods for problem 4

h	METHODS	MAXERR		
		$x_n = 100$	$x_n = 1000$	$x_n = 5000$
0.64	RK3(3)	1.266893(-1)	2.291779(-1)	2.291779(-1)
	RK4(4)	4.079385(-2)	6.735437(-2)	6.735437(-2)
	IRK3(3)	1.915567(-1)	3.468308(-1)	3.468308(-1)
	PHAFRK5(4)	8.632681(-4)	2.062964(-3)	6.606088(-3)
	TFIRK3(3)	7.979304(-4)	1.815122(-3)	3.815122(-3)
0.32	RK3(3)	3.666083(-2)	9.562916(-2)	9.562916(-2)
	RK4(4)	2.164506(-3)	2.893624(-3)	2.893624(-3)
	IRK3(3)	1.145490(-2)	1.545616(-2)	1.545616(-2)
	PHAFRK5(4)	2.661557(-5)	8.432287(-5)	4.352526(-4)
	TFIRK3(3)	3.048019(-5)	5.450344(-5)	5.450344(-5)
0.16	RK3(3)	5.630694(-3)	1.442741(-2)	1.442741(-2)
	RK4(4)	1.153071(-4)	1.381891(-4)	1.381891(-4)
	IRK3(3)	5.894151(-4)	7.082662(-4)	7.082662(-4)
	PHAFRK5(4)	1.071357(-6)	5.094904(-6)	2.651443(-5)
	TFIRK3(3)	1.446763(-6)	2.095395(-6)	2.095459(-6)
0.08	RK3(3)	7.385177(-4)	1.854599(-3)	1.854599(-3)
	RK4(4)	6.592747(-6)	7.471922(-6)	7.471922(-6)
	IRK3(3)	3.337535(-5)	3.786252(-5)	3.786252(-5)
	PHAFRK5(4)	4.781862(-8)	3.192325(-7)	1.627078(-6)
	TFIRK3(3)	7.439918(-8)	8.890895(-8)	8.892069(-8)

Table 7: Comparison between TFIRK3(3) method with existing methods for problem 5

h	**METHODS**	**MAXERR**		
		$x_n = 100$	$x_n = 1000$	$x_n = 5000$
0.16	RK3(3)	2.387444(-2)	2.214491(-1)	8.130300(-1)
	RK4(4)	7.567639(-4)	7.714520(-3)	3.852962(-2)
	IRK3(3)	4.015683(-3)	4.019858(-2)	1.993549(-1)
	PHAFRK5(4)	2.283841(-6)	2.725074(-6)	2.725074(-6)
	TFIRK3(3)	1.975309(-12)	8.903023(-11)	3.113445(-9)
0.08	RK3(3)	3.007309(-3)	2.979526(-2)	1.429583(-1)
	RK4(4)	4.741912(-5)	4.822438(-4)	2.412815(-3)
	IRK3(3)	2.493380(-4)	2.498021(-3)	1.249608(-2)
	PHAFRK5(4)	1.342089(-7)	1.5205127)	1.520743(-7)
	TFIRK3(3)	2.586376(-12)	3.085833(-10)	4.821121(-9)
0.04	RK3(3)	3.762166(-4)	3.759574(-3)	1.872147(-2)
	RK4(4)	3.962544(-6)	3.016369(-5)	1.508013(-4)
	IRK3(3)	1.555660(-5)	1.559452(-4)	7.796756(-4)
	PHAFRK5(4)	8.108070(-9)	8.965586(-9)	8.965586(-9)
	TFIRK3(3)	6.323830(-12)	4.578304(-10)	5.581152(-9)
0.02	RK3(3)	4.702772(-5)	4.705015(-4)	2.353847(-3)
	RK4(4)	1.851573(-7)	1.884332(-6)	9.449318(-6)
	IRK3(3)	9.718886(-7)	9.740780(-6)	4.873297(-5)
	PHAFRK5(4)	5.148999(-10)	5.770075(-10)	5.770139(-10)
	TFIRK3(3)	6.191048(-12)	8.809625(-10)	2.313725(-8)

Figures below show the numerical results of selected methods, comprised of TFIRK3(3), RK3(3), RK4(4), IRK3(3) and PHSFRK5(4) in term of maximum global truncation error versus number of function evaluation.

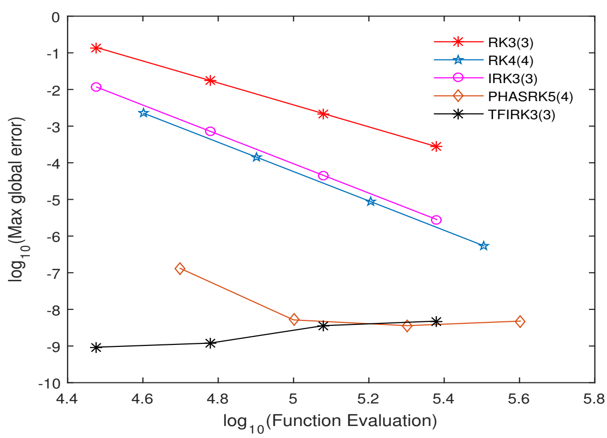

Figure 2: Efficiency curves for problem 1 with $x_n = 100$ and $h = \frac{1}{100(2^i)}, i = 0, \ldots, 3$.

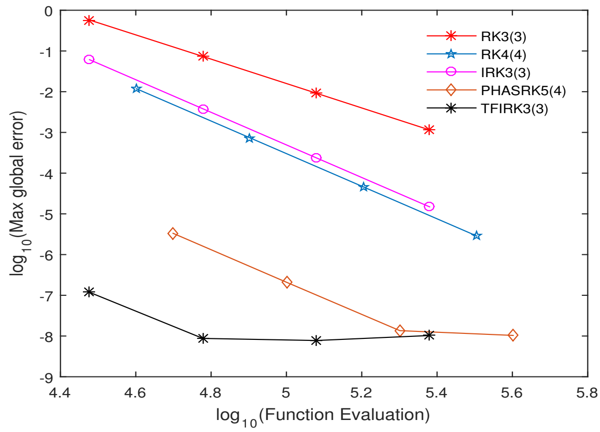

Figure 3: Efficiency curves for problem 2 with $x_n = 100$ and $h = \frac{1}{100(2^i)}$, $i = 0, \ldots, 3$.

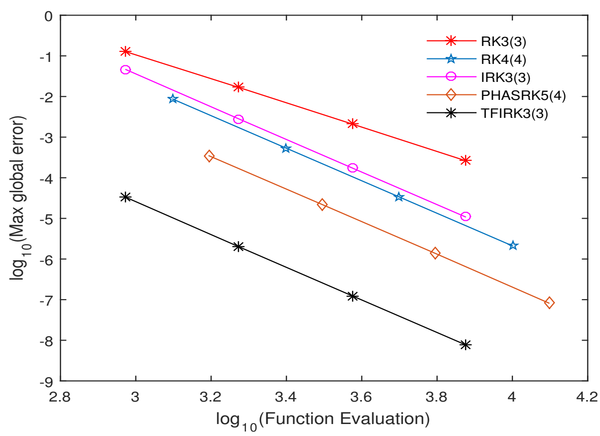

Figure 4: Efficiency curves for problem 3 with $x_n = 100$ and $h = \frac{8}{25(2^i)}$, $i = 0, \ldots, 3$.

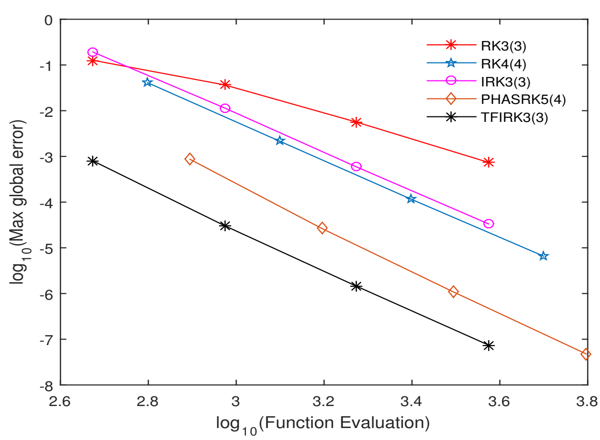

Figure 5: Efficiency curves for problem 4 with $x_n = 100$ and $h = \frac{16}{25(2^i)}$, $i = 0, \ldots, 3$.

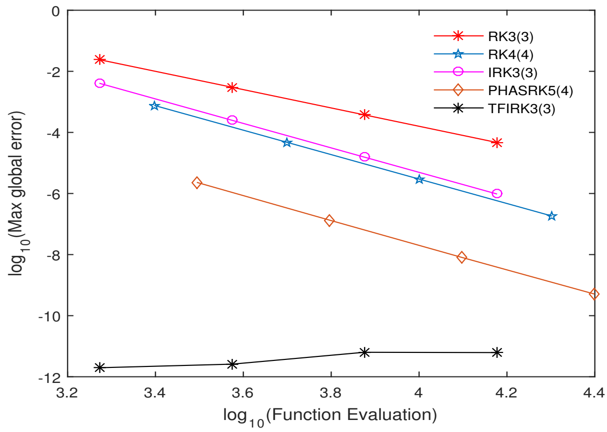

Figure 6: Efficiency curves for problem 5 with $x_n = 100$ and $h = \frac{4}{25(2^i)}, i = 0, \ldots, 3$.

4 Discussion and Conclusion

This study proposed an accurate three-stage third-order method that can be used to solve oscillatory problems. The proposed method gives more accurate numerical results compared to existing methods.

Tabulated numerical results are shown in Tables 3 - 7, TFIRK3(3) method is compared with existing RK3(3), RK4(4), IRK3(3) and PHSFRK5(4) methods to solve system of first order ODEs with different endpoints, $x_n = 100, 1000$ and 5000. We clearly notice that maximum global error decreases as the number of step size, h decreases. The proposed method, TFIRK3(3) has the least maximum global error among all four methods in solving five different numerical tests in all chosen endpoints.

Figures 2 - 6 show the efficiency of the TFIRK3(3) method compared with existing methods, RK3(3), RK4(4), IRK3(3) and PHSFRK5(4) in term of maximum global truncation error and number of function evaluation used. All four methods are also tested by using system of linear equations as well as application problem. In Figure 2 and Figure 6, the maximum global error of TFIRK3(3) method flattens and raises a little with the increment of number of evaluation, but still lower than other existing methods. This is because the values of a_{31} and a_{32} are depend on the step size. As the value of step size decreases, the values of a_{31} and a_{32} will approach to the original coefficients without trigonometric-fiiting properties and lead to higher global error. In other figures, TFIRK3(3) method converges to exact solution as the number of evaluation increases.

TFIRK method is more effective than other methods in solving linear homogeneous system in Figure 2. Similarly, TFIRK method outperforms over RK3(3), RK4(4), IRK3(3) and PHSFRK5(4) for solving linear nonhomogenoues problem as displayed in Figures 3 - 5. In Figure 6, we obtained the TFIRK method is efficient than other method in solving application problem, i.e. Duffing oscillatory problem by getting the least maximum global error and less or equal number of function evaluation.

From the results presented, we clearly see that our proposed method is able to provide more accurate results as it gives least maximum global error compared to RK3(3), RK4(4), IRK3(3) and PHSFRK5(4) with less or equal number of function evaluation.

Acknowledgement This study was supported by the Fundamental Research Grant Scheme (Ref. No. FRGS/2/2014/SG04/UPM/02/2) awarded by the Malaysia Ministry of Education.

Conflict of Interest The authors proclaim no partiallity about the publication of this paper.

References

[1] N. A. Ahmad, N. Senu & F. Ismail (2017). Phase-fitted and amplification-fitted higher order two-derivative Runge-Kutta method for the numerical solution of orbital and related periodical IVPs. *Mathematical Problems in Engineering*, *2017*, Article ID: 1871278, 11 pages. https://doi.org/10.1155/2017/1871278.

[2] I. Alolyan, Z. A. Anastassi & T. E. Simos (2012). A new family of symmetric linear four-step methods for the efficient integration of the Schrödinger equation and related oscillatory problemsesis. *Applied Mathematics and Computation*, *218*(9), 5370–5382.

[3] Z. A. Anastassi & T. E. Simos (2012). A parametric symmetric linear four-step method for the efficient integration of the Schrödinger equation and related oscillatory problems. *Journal of Computational and Applied Mathematics*, *236*(16), 3880–3889.

[4] J. Butcher (1996). A history of Runge-Kutta methods. *Applied Numerical Mathematics*, *20*(3), 247–260.

[5] B. Bylina & J. Potiopa (2017). Explicit fourth-order Runge-Kutta method on intel xeon phi oprocessor. *Applied Numerical Mathematics*, *45*(5), 1073–1090.

[6] F. Fawzi, N. Senu & Z. Abd Majid (2016). New phase-fitted and amplification-fitted modified Runge-Kutta method for solving oscillatory problems. *Global Journal of Pure and Applied Mathematics*, *12*(2), 1229–1242.

[7] Y. Jikantoro, A. Yahaya, A. Ma'ali, A. Abubakar & I. Musa (2019). A numerical integrator for oscillatory problems. *Asian Research Journal of Mathematics*, *14*, 1–10.

[8] S. Narayanamoorthy & T. Yookes (2017). Third-order Runge-Kutta method for solving differential equation in fuzzy environment. *International Journal of Pure and Applied Mathematics*, *101*(5), 795–802.

[9] F. Rabiei (2011). Third-order improved Runge-Kutta method for solving ordinary differential equation. *International Journal of Applied Physics and Mathematics*, *1*(3), 191–194.

[10] F. Rabiei & F. Ismail (2011). Fifth-order improved Runge-Kutta method with reduced number of function evaluations. *Australian Journal of Basic and Applied Sciences*, *6*(3), 97–105.

[11] F. Rabiei & F. Ismail (2013). Improved Runge-Kutta methods for solving ordinary differential equations. *Sains Malaysiana*, *42*(11), 1679–1687.

[12] C. Runge (1895). Ueber die numerische auflosung von differentialgleichungen. *Mathematische Annalen*, *46*(2), 167–178.

[13] T. Simos (2003). Exponentially-fitted and trigonometrically-fitted symmetric linear multistep methods for the numerical integration of orbital problems. *Physics Letters A*, *315*(6), 437–446.

[14] T. Simos & P. S. Williiams (2003). A new family of exponentially-fitted methods. *Mathematical and Computer Modelling*, *38*(5-6), 571–584.

[15] T. T. Wu (2003). Quantum memory: Write, read, reset, and decoherence. In *Quantum Information and Computation*, volume 5105 pp. 204–215. International Society for Optics and Photonics, United States.

[16] T. T. Wu (2004). Quantum cryptography and quantum memory. In *Quantum Information and Computation II*, volume 5436 pp. 81–92. International Society for Optics and Photonics, United States.

[17] X. You, Y. Zhang & J. Zhao (2011). Trigonometrically-fitted scheifele two-step methods for perturbed oscillators. *Mathematical and Computer Modelling*, *182*(7), 1481–1490.

[18] Y. Zhang, H. Che, Y. Fang & X. You (2013). A new trigonometrically fitted two-derivative Runge-Kutta method for the numerical solution of the Schrödinger equation and related problems. *Journal of Applied Mathematics*, *2013*, Article ID: 937858 , 9 pages. https://doi.org/10.1155/2013/937858.