Malaysian Journal of Mathematical Sciences 16(2): 215–235 (2022)
https://doi.org/10.47836/mjms.16.2.04

Malaysian Journal of Mathematical Sciences

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

Stability Analysis of Diagonally Implicit Two Derivative Runge-Kutta methods for Solving Delay Differential Equations

Ahmad, N. A. *[1], Senu, N. [1,2], Ibrahim, Z. B. [2], and Othman, M. [3]

[1]*Institute for Mathematical Research, Universiti Putra Malaysia, Malaysia*
[2]*Department of Mathematics and Statistics, Faculty of Science, Universiti Putra Malaysia, Malaysia*
[3]*Department of Communication Technology and Network, Universiti Putra Malaysia, Malaysia*

E-mail: nuramirah_ahmad@yahoo.com
**Corresponding author*

Received: 19 June 2021
Accepted: 29 December 2021

Abstract

The stability properties of fourth and fifth-order Diagonally Implicit Two Derivative Runge-Kutta method (DITDRK) combined with Lagrange interpolation when applied to the linear Delay Differential Equations (DDEs) are investigated. This type of stability is known as P-stability and Q-stability. Their stability regions for $(\lambda, \mu \in \Re)$ and $(\mu \in \mathbb{C}, \lambda = 0)$ are determined. The superiority of the DITDRK methods over other same order existing Diagonally Implicit Runge-Kutta (DIRK) methods when solving DDEs problems are clearly demonstrated by plotting the efficiency curves of the log of both maximum errors versus function evaluations and the CPU time taken to do the integration.

Keywords: diagonally implicit two derivative Runge-Kutta method; delay differential equations; initial value problems; P-stability; Q-stability.

1 Introduction

Numerous physical systems possess the feature of having a delayed response to input conditions. Therefore, the level at which the events occur depends not only on the present state but also on the previous state of the system. Mathematical models of such occurrences generally results in differential equations with a time delay. These type of equations are known as delay differential equations (DDEs). The DDE of first-order can be expressed as

$$\left.\begin{aligned} y^{'}(t) &= f(t, y(t), y(t-\tau)), \quad & t > t_0, \\ y(t) &= \varphi(t), & t \leq t_0. \end{aligned}\right\} \tag{1}$$

where the initial function is given as $\varphi(t)$, $t - \tau(t, y(t))$ is termed as the delay argument, $\tau(t, y(t))$ is termed as the delay, the delay term solution is given as the value of $y(t-\tau(t, y(t)))$ or frequently known to only as the delay term. These days, DDEs are becoming a necessary criteria in exploring the nature of real-world problems applicable to neuronal networks, contagious diseases, population structure and the biotic population.

DDEs are widely solvable using methods such as Runge-Kutta (RK), Runge-Kutta-Nyström (RKN), multistep and hybrid methods. Usually, the finest and most possibly the only practical representation of real world phenomena are provided by DDEs. These studies have been published by Orbele and Pesch[24] where a class of numerical methods for the treatment of DDEs is developed based on the well-known Runge-Kutta-Fehlberg methods. The retarded argument is approximated by an appropriate multipoint Hermite Interpolation.

Authors such as Zennaro[31] and Al-Mutib[3] have been investigating on stability properties of DDEs. Zennaro[31] proved that any A-stable one-step collocation method for ODEs inherits the same property when it is applied to DDEs with a constrained mesh (i.e. it is P-stable). Meanwhile, Al-Mutib[3] considered the stability properties of numerical methods for DDEs where some suitable definitions for the stability of the numerical methods are included and RK type methods satisfying these properties are tested on a numerical example.

Ismail and Suleiman[12] investigated the P-stability and Q-stability properties of Singly Diagonally Implicit Runge-Kutta (SDIRK) method with the combination of fourth-order embedded in fifth-order DDE using Lagrange Interpolation. Other than that, Ismail et al.[11] solve first-order DDE using different order embedded DIRK method using Hermite Interpolation. The decomposition method as an integrator for DDE have been done by Taiwo and Odetunde[29] in their paper.

Mechee et al.[22] solve special second-order DDE using RKN method by reducing them to first order DDE and solved them using the existing RK method as written in their paper. The stability of the RK method for linear DDE have been discussed by Maset[21] in which they are implemented to complex linear scalar DDE and is called τ-stability. It has been proven that the implicit Euler method is τ-stable. Bartoszewski and Jackiewicz[4] investigate the stability properties of the two-step RK method which any A-stable two-step RK method, the corresponding method is P-stable. Some other authors proposed block linear multistep method (LMM) in solving DDEs and these study can be found in [10]–[30].

Other recent works related to DDEs can be seen in Kumar and Pushpam[18] where they developed RK method for solving DDEs in which they include new terms of higher derivative of f in the RK k_i terms $(i > 1)$ to obtain better accuracy without increasing number of evaluations of f, but with the addition of approximations of $f^{'}$. Besides, their work in [19] proposed a two-stage multiderivative clarifying RK method of order four whereby Lagrange interpolation is applied for estimating the delay term. The stability polynomial of the method is used in obtaining the corresponding stability region.

Furthermore, Shaalini and Pushpam[28] presented the generalized rational multi-step method for solving DDEs where they developed the r-step p-th order generalized multi-step method which is based on rational function approximation technique. Meanwhile, in Ismail et al.[13], Neutral DDE of pantograph type is solved using fifth order explicit multistep block method where a two-point explicit multistep block method has been modelled by applying Taylor Series interpolation polynomial.

Fang and Zhan[6] analyze the order conditions of high order explicit exponential RK methods for stiff semilinear delay differential equations. The stiff order conditions up to order five are derived by taking into account the framework of analytic semigroup and the natural assumptions on the delay differential equations. In the same year, Jaaffar et al.[14] introduces a direct multistep method to solve third order DDEs of constant and pantograph types based on the boundary conditions given. Direct integration approach is used to reduce the total function calls involved and the method is derived implicitly to attain accuracy.

In this recent year, there are no research findings associated to DITDRK methods for solving DDEs as well as the analysis on their stability properties. The benefits or drawbacks of of the P-stability and Q-stability of DITDRK methods have not yet discussed thoroughly by researchers especially mathematicians. Hence, in this research context, the stability properties of both DITDRK methods of fourth and fifth-order are investigated. The region of P-stability and Q-stability is determined for each method and we will solve some related DDEs problems using these methods. The approximation of the delay term is by using Lagrange interpolation and our focus here is to solve retarded first-order DDEs with constant delay. The efficiency and accuracy of the method derived will be compared with other existing same order DIRK methods. Using DITDRK methods, we can actually achieve a higher order method with a lower stage number. Less number of functions to be evaluated at every step thus, leading to a reduction of computational cost.

2 Two Derivative Runge-Kutta Methods

The Ordinary Differential Equation (ODEs) for solving numerical methods are usually expressed as

$$y^{'}(t) = f(t, y(t)), \quad y(t_0) = c, \tag{2}$$

where they are possible to be adjusted in solving DDEs.

Consider the scalar ODEs (2) with $g : \mathcal{R}^N \rightarrow \mathcal{R}^N$. It is assumed, in this case, the second derivative is known where

$$y^{''} = g(y) := f'(y)f(y), \qquad g : \mathcal{R}^N \rightarrow \mathcal{R}^N. \tag{3}$$

The numerical integration of ODEs (1) for a TDRK method is given by

$$Y_i = g\left(t_n + c_i h, y_n + h\sum_{j=1}^{s} a_{ij} f(y_n) + h^2 \sum_{j=1}^{s} \hat{a}_{ij} Y_j\right), \tag{4}$$

$$y_{n+1} = y_n + h\sum_{i=1}^{s} b_i f(y_n) + h^2 \sum_{i=1}^{s} \hat{b}_i Y_i, \tag{5}$$

where $i = 1, \ldots, s$.

A lowest number of function evaluations for diagonally implicit methods can be established by considering the methods in the following manner

$$Y_i = g\left(t_n + c_i h, y_n + hc_i f(t_n, y_n) + h^2 \sum_{j=1}^{s} \hat{a}_{ij} Y_j\right), \tag{6}$$

$$y_{n+1} = y_n + hf(t_n, y_n) + h^2 \sum_{i=1}^{s} \hat{b}_i Y_i, \tag{7}$$

where $i = 1, \ldots, s$. The method described above is identified as a unique DITDRK method. This method's remarkable aspect is it requires just an evaluation of function f and a number of evaluations of function g at each step compared to a large number of evaluations of function f at each step in the traditional RK methods. Given below illustrates the significant difference between the DITDRK method and the unique DITDRK method in term of Butcher tableau.

$$\begin{array}{c|c||c} \mathrm{c} & \mathrm{A} & \hat{A} \\ \hline & b^T & \hat{b}^T \end{array} \quad \rightarrow \quad \begin{array}{c||c} \mathrm{c} & \hat{A} \\ \hline & \hat{b}^T \end{array}$$

Adapt the DITDRK method given by (6)-(7) to DDE (1) will give us

$$Y_i = g\left(t_n + c_i h, y_n + hc_i f(t_n, y_n) + h^2 \sum_{j=1}^{s} \hat{a}_{ij} Y_j, y(t_n + c_i h - \tau)\right), \tag{8}$$

$$y_{n+1} = y_n + hf(t_n, y_n) + h^2 \sum_{i=1}^{s} \hat{b}_i Y_i, \tag{9}$$

where $i = 1, \ldots, s$.

Referring to the above equation, the delay term is identified as $y(t_n + c_i h - \tau)$ and the approximation of the values of the delay term will be carried out using interpolation. Neves[23], Int Hout[8] and Karoui[17] investigated and studied a wide variety of getting approximation procedures or techniques. Therefore here, we employed Lagrange interpolation for the approximation of the delay term. Hence, the order of interpolation need to be adjusted as well as the support points number.

The order of the DITDRK method is denoted as p, hence q which is the interpolation order must be equal or greater to p. i_p is the number of support points for newton divided difference interpolation, then $2i_p \geq p$.

In this research paper, fourth and fifth-order DITDRK method will be used to solve DDEs. For this method, we used five and six support points respectively so that $2i_p \geq p$ holds.

The most crucial basic requirements in solving DDEs is the preservation of adequate back information. This will allow the delay term to be evaluated whenever $t \leq tn$ is needed at a certain point. The method that we used for approximating the delay term determines the amount of data to be preserved at each time step but at that period of time, the time interval for which the information and the quantity to be preserved must be adjustable enough to each selected problems, based on the required delay term nature and accuracy. The initial function should be used whenever the delay term occurs at a particular point $t \leq t_0$.

The current step comprises of the delay argument due to its size which is smaller than the step size or may very well vanish. When the delay vanishes, we denote this kind of delay as a small delay or vanishing delay. Such delays are treated by the use of extrapolation or by limiting the size of the stepsize to be far less than the approximated delays.

3 Stability of DITDRK method

The linear DDE is is given by the following equation

$$\left.\begin{aligned} y^{'}(t) = &\quad \lambda y(t) - \mu y(t-\tau) \quad (\tau > 0) \\ y(t) = &\quad \varphi(t) \qquad\qquad\qquad (t \in [-\tau, 0]) \end{aligned}\right\} \tag{10}$$

where $\lambda, \mu \in \mathbb{C}, \tau > 0$ and $\varphi(t)$ is a specified initial condition. In addition, assume that the second derivative of (10) is given below

$$\left.\begin{aligned} y^{''}(t) = &\quad \lambda^2 y(t) + 2\lambda\mu y(t-\tau) + \mu^2 y(t-2\tau) \quad (\tau > 0) \\ y^{'}(t) = &\quad \varphi^{'}(t) \qquad\qquad\qquad (t \in [-\tau, 0]) \end{aligned}\right\}. \tag{11}$$

It is known that from [5] and [3], if $\varphi(t)$ is continuous and if

$$|\mu| < -\text{Re}(\lambda), \tag{12}$$

the solution of (10) tends to zero as $t \to \infty$. Now, some definitions given in [31, 5] and [3] will be stated below.

Definition 3.1. *Given a numerical method for DDEs, the P-stability region of the method is the set of S_p of the pairs of complex $(\alpha, \beta), \alpha := h\lambda$ and $\beta := h\mu$, such that the numerical solution of (10) asymptotically vanishes for step lengths h satisfying*

$$h = \frac{\tau}{m}, \qquad m \quad \text{positive integer}.$$

Definition 3.2. *If $\lambda = 0$ and μ is complex in (10), then the Q-stability region of the method is the set of S_q of β, such that the numerical solution vanishes for $h = (\tau/m)$.*

4 P-Stability Analysis

For ODEs (2), we can write the method of order $p \geq 1$ as

$$Y_{n+1}^{(i)} = g\left(t_n + c_i h, y_n + hc_i f(t_n, y_n) + h^2 \sum_{j=1}^{i} \hat{a}_{ij} Y_{n+1}^{(j)}\right), \tag{13}$$

$$y_{n+1} = y_n + hf(t_n, y_n) + h^2 \sum_{i=1}^{q} \hat{b}_i Y_{n+1}^{(i)}, \tag{14}$$

whereby the internal stage of the method is denoted as i. DDE (10) is considered with $\tau = 1$. The calculation assumption of the numerical solution is up till point t_n along with a uniform step-size h which satisfy $h = (\tau/m)$ and m is a non-negative integer. Using previously calculated values of y where Lagrange interpolation is used to approximate the delay term will give us

$$\begin{aligned} y(t_n + c_i h - 1) &= y(t_{n-m} + c_i h) \\ &= \sum_{l=-r1}^{s1} \mathcal{L}_l(c_i) y_{n-m+l}, \end{aligned} \tag{15}$$

with

$$\mathcal{L}_l(c_i) = \prod_{j_1=-r1}^{s1} \frac{(c_i - j_1)}{(l - j_1)}, \quad j_1 \neq l,$$

and y_{n-m+l} is the calculated value of $y(t_{n-m+l})$. The following equations are obtained when the DITDRK method is applied to DDE (10) with constant delay $\tau = 1$,

$$\begin{aligned} Y_{n+1}^{(i)} = g\Bigg(t_n + c_i h, y_n + hc_i \left(\lambda y_n + \mu \sum_{l=r1}^{s1} \mathcal{L}_l(c_i) y_{n-m+l}\right) + h^2 \sum_{j=1}^{i} \hat{a}_{ij} Y_{n+1}^{(j)}, \\ \sum_{l=r1}^{s1} \mathcal{L}_l(c_i) y_{n-m+l}\Bigg), \end{aligned} \tag{16a}$$

$$y_{n+1} = y_n + h\left(\lambda y_n + \mu \sum_{l=r1}^{s1} \mathcal{L}_l(c_i) y_{n-m+l}\right) + h^2 \sum_{i=1}^{q} \hat{b}_i Y_{n+1}^{(i)}. \tag{16b}$$

$u = (1, \ldots, 1)^T$ is defined for $n \geq 1$. Hence,

$$\begin{aligned} \underline{Y_n} &= (Y_n^{(1)}, Y_n^{(2)}, \ldots, Y_n^{(s)})^T, \\ b &= (b_1, \ldots, b_s)^T, \end{aligned}$$

and

$$\mathcal{L}_l(c) = (\mathcal{L}_l(c_1), \ldots, \mathcal{L}_l(c_q))^T.$$

For $n \geq m$, (16) takes the form

$$\begin{aligned}\underline{Y_{n+1}} =&\lambda^2(y_n u + c_i h\lambda y_n + c_i h\mu \sum \mathcal{L}_l(c) y_{n-m+l} + h^2 \hat{A} Y_{n+1}) + \\ &2\mu\lambda \sum \mathcal{L}_l(c) y_{n-m+l} + \mu^2 \sum \mathcal{L}_l(c) y_{n-2m+l}, \qquad (17a)\\ y_{n+1} =&y_n + h\lambda y_n + h\mu \sum \mathcal{L}_l(c) y_{n-m+l} + h^2 \hat{b}^T \underline{Y_{n+1}}. \qquad (17b)\end{aligned}$$

From (17a), we have

$$\begin{aligned}(I - h^2\lambda^2\hat{A})\underline{Y_{n+1}} =&\lambda^2 y_n u + h\lambda^3 c_i y_n + c_i h\lambda^2\mu \sum \mathcal{L}_l(c) y_{n-m+l} + \\ &2\mu\lambda \sum \mathcal{L}_l(c) y_{n-m+l} + \mu^2 \sum \mathcal{L}_l(c) y_{n-2m+l},\end{aligned}$$

or

$$\begin{aligned}h^2\underline{Y_{n+1}} =&\alpha^2(I - \alpha^2\hat{A})^{-1} y_n u + \alpha^3(I - \alpha^2\hat{A})^{-1} c_i y_n + \alpha^2\beta c_i (I - \alpha^2\hat{A})^{-1} \sum \mathcal{L}_l(c_i) y_{n-m+l} \\ &2\alpha\beta(I - \alpha^2\hat{A})^{-1} \sum \mathcal{L}_l(c) y_{n-m+l} + \beta^2(I - \alpha^2\hat{A})^{-1} \sum \mathcal{L}_l(c) y_{n-2m+l}. \qquad (18)\end{aligned}$$

Here, $\alpha = \lambda h, \beta = \mu h$ and I is the identity matrix. Substitute (18) into (17b) will give

$$\begin{aligned}y_{n+1} =&y_n + \alpha y_n + \beta \sum \mathcal{L}_l(c) y_{n-m+l} + b^T \Big[\alpha^2(I - \alpha^2\hat{A})^{-1} y_n u + \alpha^3(I - \alpha^2\hat{A})^{-1} c_i y_n + \\ &\alpha^2\beta(I - \alpha^2\hat{A})^{-1} c_i \sum \mathcal{L}_l(c) y_{n-m+l} + 2\alpha\beta(I - \alpha^2\hat{A})^{-1} \sum \mathcal{L}_l(c) y_{n-m+l} + \\ &\beta^2(I - \alpha^2\hat{A})^{-1} \sum \mathcal{L}_l(c) y_{n-2m+l}\Big], \\ =&\Big(1 + \alpha + \alpha^2 b^T(I - \alpha^2\hat{A})^{-1} u + \alpha^3 b^T(I - \alpha^2\hat{A})^{-1} c_i\Big) y_n + \Big(\beta + \alpha^2\beta b^T(I - \alpha^2\hat{A})^{-1} c_i \\ &+2\alpha\beta b^T(I - \alpha^2\hat{A})^{-1}\Big) \sum \mathcal{L}_l(c) y_{n-m+l} + \beta^2 b^T(I - \alpha^2\hat{A})^{-1} \sum \mathcal{L}_l(c) y_{n-2m+l}. \qquad (19)\end{aligned}$$

Taking $\mathbf{y}_n = (y_n, h^2\underline{Y_n})^T$, the compact form of (19) and (18) can be written as the following,

$$\mathbf{y}_{n+1} = \mathbf{X}\mathbf{y}_n + \mathbf{W}\mathbf{y}_{n-m+l} + \mathbf{Z}\mathbf{y}_{n-2m+l} \qquad (20)$$

where

$$\mathbf{X} = \left[\begin{array}{c|c} 1 + \alpha + \alpha^2 b^T\eta u + \alpha^3 b^T\eta c_i & 0, \dots, 0 \\ \hline & 0 \\ \alpha^2\eta u + \alpha^3\eta c_i & \vdots \\ & 0 \end{array}\right], \quad \mathbf{Z} = \left[\begin{array}{c|c} \underline{\beta^2 b^T\eta \sum \mathcal{L}_l(c)} & 0 \\ & \vdots \\ \beta^2\eta \sum \mathcal{L}_l(c) & \vdots \\ & 0 \end{array}\right],$$

$$\mathbf{W} = \left[\begin{array}{c|c} (\beta u^T + \alpha^2\beta b^T\eta c_i u^T + 2\alpha\beta b^T\eta) \sum \mathcal{L}_l(c) & 0, \dots, 0 \\ \hline & 0 \\ (\alpha^2\beta\eta c_i u^T + 2\alpha\beta\eta) \sum \mathcal{L}_l(c) & \vdots \\ & 0 \end{array}\right].$$

and $\eta = (I - \alpha^2\hat{A})^{-1}$. The standard form of the stability polinomial will appear when we put $n - m = 2$ and $n - 2m = 2$. Meanwhile, the recurrence is said to be stable if the zeroes of ζ_i of the stability polynomial (as shown below) satisfy the root conditions,

$$\mathbf{S}(\alpha,\beta,\zeta) = \det[\zeta^{m+3}I - \zeta^{m+2}\mathbf{X} - \zeta^{2+l}\mathbf{W} - \zeta^{2+l}\mathbf{Z}], \tag{21}$$
$$\mathbf{S}(\alpha,\beta,\zeta) = \det[\zeta^{m+4}I - \zeta^{m+3}\mathbf{X} - \zeta^{3+l}\mathbf{W} - \zeta^{3+l}\mathbf{Z}]. \tag{22}$$

(21) and (22) represent the stability polynomial for fourth and fifth-order respectively. Equation (21) can be expressed as the following equation

$$\det\left[\zeta^{m+3}\begin{bmatrix}1 & 0,\dots,0\\ \underline{0} & I\end{bmatrix} - \zeta^{m+2}\begin{bmatrix}1+\alpha+\alpha^2 b^T\eta u+\alpha^3 b^T\eta c_i & 0,\dots,0\\ \alpha^2\eta u+\alpha^3\eta c_i & \underline{0}\end{bmatrix} - \right.$$
$$\zeta^{2+l}\begin{bmatrix}(\beta u^T+\alpha^2\beta b^T\eta c_i u^T+2\alpha\beta b^T\eta)\sum\mathcal{L}_l(c) & 0,\dots,0\\ (\alpha^2\beta\eta c_i u^T+2\alpha\beta\eta)\sum\mathcal{L}_l(c) & \underline{0}\end{bmatrix} -$$
$$\left.\zeta^{2+l}\begin{bmatrix}\beta^2 b^T\eta\sum\mathcal{L}_l(c) & 0,\dots,0\\ \beta^2\eta\sum\mathcal{L}_l(c) & \underline{0}\end{bmatrix}\right] = 0.$$

To illustrate the stability analysis, the folowing butcher tableau show the coefficients of the fourth and fifth-order method in [2], DITDRK(2,4) and DITDRK(3,5) respectively.

Table 1: Butcher tableau for DITDRK(2,4) method.

$\frac{1}{5}$	$\frac{1}{50}$	
$\frac{3}{4}$	$\frac{209}{800}$	$\frac{1}{50}$
	$\frac{25}{66}$	$\frac{4}{33}$

Table 2: Butcher tableau for DITDRK(3,5) method.

$\frac{1}{3}$	$\frac{1}{18}$		
$\frac{2}{5}-\frac{\sqrt{6}}{10}$	$\frac{49}{900}-\frac{\sqrt{6}}{25}$	$\frac{1}{18}$	
$\frac{1+\sqrt{6}}{-2+3\sqrt{6}}$	$\frac{1}{18}\frac{-118+27\sqrt{6}}{(-2+3\sqrt{6})^3}$	$\frac{-4(-9+\sqrt{6})}{(-2+3\sqrt{6})^3}$	$\frac{1}{18}$
	0	$\frac{1}{4}+\frac{\sqrt{6}}{36}$	$-\frac{1}{24}\frac{(-2+3\sqrt{6})^2}{(-9+\sqrt{6})}$

The P-stability region of these methods will be obtained by using five and six points interpolation to evaluate $y(t_n + c_i h - 1)$. We take $l = -2,\dots,2$ and $l = -3,\dots,2$ for fourth and fifth-order respectively. Hence letting $n - m - 2 = 0$ and $n - m - 3 = 0$, the stability polynomial (21) and (22) can be written as

$$\begin{aligned}\mathbf{S}(\alpha,\beta,\zeta) =& \zeta^{m+3} - \left(1+\alpha+\alpha^2 b^T\eta u+\alpha^3 b^T\eta c_i\right)\zeta^{m+2} - \left(\beta u^T+\alpha^2\beta b^T\eta c_i u^T+2\alpha\beta b^T\eta\right)\\ &\left(\mathcal{L}_{-2}(c)+\mathcal{L}_{-1}(c)\zeta^1+\mathcal{L}_0(c)\zeta^2+\mathcal{L}_1(c)\zeta^3+\mathcal{L}_2(c)\zeta^4\right) - \beta^2 b^T\eta\,(\mathcal{L}_{-2}(c)+\\ &\mathcal{L}_{-1}(c)\zeta^1+\mathcal{L}_0(c)\zeta^2+\mathcal{L}_1(c)\zeta^3+\mathcal{L}_2(c)\zeta^4)\,,\end{aligned} \tag{23}$$

$$\begin{aligned}\mathbf{S}(\alpha,\beta,\zeta) =& \zeta^{m+4} - \left(1+\alpha+\alpha^2 b^T\eta u+\alpha^3 b^T\eta c_i\right)\zeta^{m+3} - \left(\beta u^T+\alpha^2\beta b^T\eta c_i u^T+2\alpha\beta b^T\eta\right)\\ &\left(\mathcal{L}_{-3}(c)+\mathcal{L}_{-2}(c)\zeta^1+\mathcal{L}_{-1}(c)\zeta^2+\mathcal{L}_0(c)\zeta^3+\mathcal{L}_1(c)\zeta^4+\mathcal{L}_2(c)\zeta^5\right) - \beta^2 b^T\eta\\ &\left(\mathcal{L}_{-3}(c)+\mathcal{L}_{-2}(c)\zeta^1+\mathcal{L}_{-1}(c)\zeta^2+\mathcal{L}_0(c)\zeta^3+\mathcal{L}_1(c)\zeta^4+\mathcal{L}_2(c)\zeta^5\right).\end{aligned} \tag{24}$$

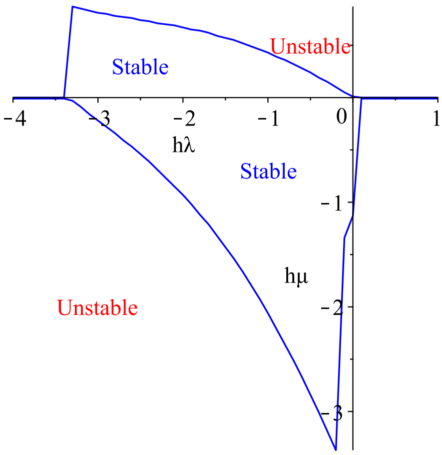


Figure 1: P-Stability region of DITDRK(2,4) method.

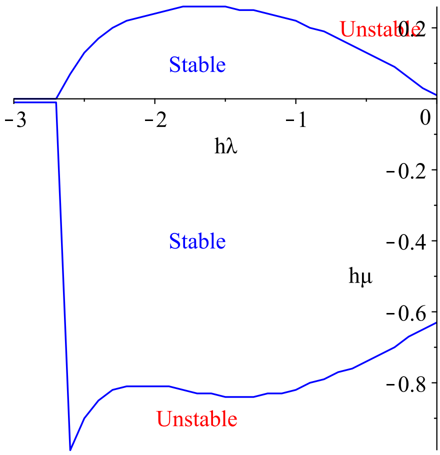


Figure 2: P-Stability region of DITDRK(3,5) method.

4.1 Boundary of P-Stability Region

ζ_i are the stability polynomial roots for $i = 1, \ldots, 4$ and let $h = m = 1$. max r=max$_i|\zeta_i|$ is the roots maximum value, and λ and $\mu \in \Re$. The set of values (λ, μ) forms the stability region of the methods for which the roots ζ_i given by (23) satisfy the root condition $\zeta_i \leq 1$.

Find the values of μ_k and μ_{k+1} for any value of λ_j, such that max r at $(\lambda_j, \mu_k) \leq 1$ and max r at $(\lambda_j, \mu_{k+1}) > 1$. Refine the point by repeating the process for $\mu \in (\mu_k, \mu_{k+1})$ and stop the process when a new μ_k is found. The boundary point is taken by the point (λ_j, μ_k) . The process is repeated for $\lambda \in [-5, 1]$. For simplicity, the starting value of μ_k is set to be 0 and the value is increased and decreased by 0.01 to obtain the value above and below the axis respectively. The P-stability region of DITDRK(2,4) and DITDRK(3,5) method lies in the closed region of the following Figures 1–2 respectively.

We can actually find the value of h that the method can take to remain stable from the stability interval. The value of λ and μ come from the test problem. The demonstration of the following stability test below will show us how the stability regions are used for practical purposes. We have

$$\begin{aligned} y'(t) &= \lambda y(t) - \mu y(x - \tau), && 0 \leq t \leq 10, \\ y(t) &= e^{-2t} \sin\left(-\frac{\pi}{2}t\right), && t \leq 0, \end{aligned}$$

where $\tau(t) = 1, \lambda = -2$ and $\mu = \frac{\pi}{2}e^{-2}$.

If the maximum global error is small and converging to its exact solution, we can say that the method is stable. Otherwise, a bigger maximum global error indicates that the method is unstable which means it is actually diverges from its exact solution. The stability test is carried out to show the relationship between $h, \lambda, \mu, \lambda h$ and μh for both DITDRK(2,4) and DITDRK(3,5) methods. The maximum global errors are collected in Tables 3 and 4 for a variety of h values.

Table 3: Stability test for DITDRK(2,4) method with $\lambda = -2$ and $\mu = \frac{\pi}{2}e^{-2}$ for variable h.

h	λh	μh	MAXERR
9	-18	1.913257492	2.468522×10^{15}
7	-14	1.488089161	2.198640×10^{12}
5	-10	1.062920829	2.782082×10^{11}
3	-6	0.6377524974	3.317338×10^{7}
1	-2	0.2125841658	4.124333×10^{0}
0.5	-1.0	0.1062920829	6.689750×10^{-1}
0.1	-0.2	0.02125841658	3.819775×10^{-2}
0.01	-0.02	0.002125841658	2.752345×10^{-3}

Table 4: Stability test for DITDRK(3,5) method with $\lambda = -2$ and $\mu = \frac{\pi}{2}e^{-2}$ for variable h.

h	λh	μh	MAXERR
9	-18	1.913257492	1.599902×10^{28}
7	-14	1.488089161	7.210864×10^{22}
5	-10	1.062920829	2.011514×10^{25}
3	-6	0.6377524974	2.061618×10^{14}
1	-2	0.2125841658	4.136471×10^{0}
0.5	-1.0	0.1062920829	6.693612×10^{-1}
0.1	-0.2	0.02125841658	3.819747×10^{-2}
0.01	-0.02	0.002125841658	2.752345×10^{-3}

5 Q-Stability Analysis

The DITDRK method is applied to the test equation given below,

$$\left.\begin{aligned} y'(t) = &\quad \mu y(t-\tau) \quad (\tau > 0) \\ y(t) = &\quad \varphi(t) \qquad\quad (t \in [-\tau, 0]) \end{aligned}\right\}. \tag{25}$$

Assume that the second derivative of (25) is given below,

$$\left.\begin{aligned} y''(t) = &\quad \mu^2 y(t-2\tau) \quad (\tau > 0) \\ y'(t) = &\quad \varphi'(t) \qquad\qquad (t \in [-\tau, 0]) \end{aligned}\right\}. \tag{26}$$

Approximate the delay term by using the same interpolation will give us

$$Y_{n+1}^{(i)} = \mu^2 \sum \mathcal{L}_l(c_i) y_{n-2m+l}, \tag{27a}$$

$$y_{n+1} = y_n + h\mu y(t-\tau) + h^2 b^T \underline{Y_{n+1}}, \tag{27b}$$

or

$$h^2 \underline{Y_{n+1}} = h^2 \mu^2 \sum \mathcal{L}_l(c) y_{n-2m+l}, \tag{28a}$$

$$y_{n+1} = y_n + h\mu \sum \mathcal{L}_l(c) y_{n-m+l} + h^2 \mu^2 b^T \sum \mathcal{L}_l(c) y_{n-2m+l}. \tag{28b}$$

Let $\mathbf{K}_n = (y_n, h^2 \underline{Y_n})^T$. The Q-stability polynomial of the method is

$$\det\Big[I\mathbf{K}_{n+1} - \mathbf{1}\mathbf{K}_n - \mathbf{U}\mathbf{K}_{n-m+l} - \mathbf{V}\mathbf{K}_{n-2m+l}\Big], \tag{29}$$

where

$$\mathbf{1} = \begin{bmatrix} 1 & 0, & \dots, & 0 \\ 0 & 0, & \dots, & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0, & \dots, & 0 \end{bmatrix}, \quad \mathbf{U} = \left[\begin{array}{c|c} \frac{h^2\mu^2 \sum \mathcal{L}_l(c)}{\sum \mathcal{L}_l(c)} & \begin{matrix} 0 \\ \vdots \\ \vdots \\ 0 \end{matrix} \end{array}\right], \quad \mathbf{V} = \left[\begin{array}{c|c} h^2\mu^2 b^T \sum \mathcal{L}_l(c) & 0 \\ \hline h^2\mu^2 \sum \mathcal{L}_l(c) & \begin{matrix} 0 \\ \vdots \\ 0 \end{matrix} \end{array}\right].$$

Similarly, take $l = -2, \dots, 2$ and $l = -3, \dots, 2$ for fourth and fifth-order respectively, the Q-stability polynomial of these method are given as

$$\begin{aligned} \mathbf{S}(\alpha, \beta, \zeta) =& \zeta^{m+3} - \zeta^{m+2} - \beta\left(\mathcal{L}_{-2}(c) + \mathcal{L}_{-1}(c)\zeta^1 + \mathcal{L}_0(c)\zeta^2 + \mathcal{L}_1(c)\zeta^3 + \mathcal{L}_2(c)\zeta^4\right) \\ & - \beta^2 b^T \left(\mathcal{L}_{-2}(c) + \mathcal{L}_{-1}(c)\zeta^1 + \mathcal{L}_0(c)\zeta^2 + \mathcal{L}_1(c)\zeta^3 + \mathcal{L}_2(c)\zeta^4\right), \end{aligned} \tag{30}$$

$$\begin{aligned} \mathbf{S}(\alpha, \beta, \zeta) =& \zeta^{m+4} - \zeta^{m+3} - \beta\left(\mathcal{L}_{-3}(c) + \mathcal{L}_{-2}(c)\zeta^1 + \mathcal{L}_{-1}(c)\zeta^2 + \mathcal{L}_0(c)\zeta^3 + \mathcal{L}_1(c)\zeta^4 \right. \\ & \left. + \mathcal{L}_2(c)\zeta^5\right) - \beta^2 b^T \left(\mathcal{L}_{-3}(c) + \mathcal{L}_{-2}(c)\zeta^1 + \mathcal{L}_{-1}(c)\zeta^2 + \mathcal{L}_0(c)\zeta^3 + \mathcal{L}_1(c)\zeta^4 \right. \\ & \left. + \mathcal{L}_2(c)\zeta^5\right). \end{aligned} \tag{31}$$

5.1 Boundary of Q-Stability Region

Take $m = h = 1$ and substitute $\zeta = \cos\theta + i\sin\theta, \theta \in [0, 2\pi]$ into (30) and next, solve for $\mu = a + ib$. The boundary of the Q-stability region of the method are formed using the values of a and b obtained. The region for both DITDRK(2,4) and DITDRK(3,5) method are given as the following Figures 3–4 respectively,

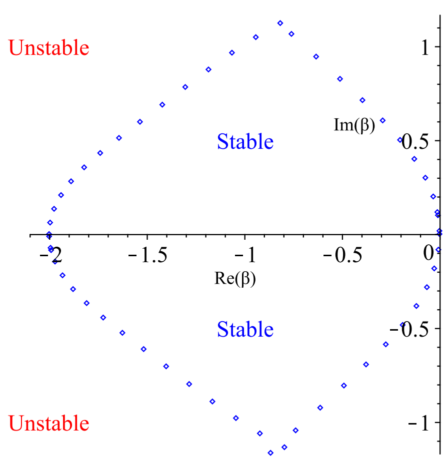


Figure 3: Q-Stability region of DITDRK(2,4) method.

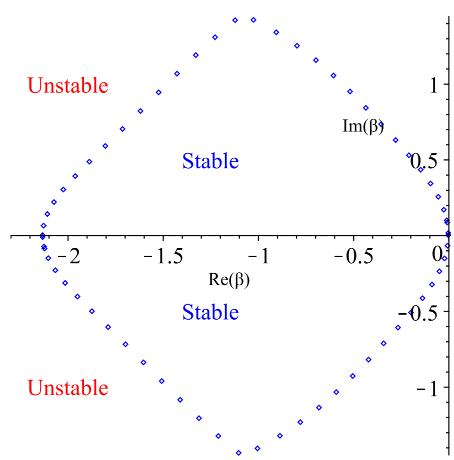


Figure 4: Q-Stability region of DITDRK(3,5) method.

The demonstration of the following stability test below will show us how the stability regions are used for practical purposes.

$$y'(t) = -\mu y(t-\tau), \qquad 0 \le t \le 10,$$
$$y(t) = \sin(t), \qquad t \le 0,$$

where $\tau(t) = \frac{\pi}{2}$ and $\mu = 1$.

The stability test is carried out to show the relationship between h and μ for both DITDRK(2,4) and DITDRK(3,5) methods. The maximum global errors are collected in Tables 5 and 6 for a variety of h values.

Table 5: Stability test for DITDRK(2,4) method with $\mu = 1$ for variable h.

h	Re(β)	MAXERR
10.0	-10.0	1.869274×10^{3}
8.0	-8.0	3.000825×10^{2}
5.0	-5.0	6.522834×10^{2}
3.0	-3.0	1.481671×10^{1}
1.0	-1.0	1.251984×10^{-2}
0.5	-0.5	1.779757×10^{-4}
0.1	-0.1	8.376573×10^{-8}
0.01	-0.01	4.168238×10^{-10}

Table 6: Stability test for DITDRK(3,5) method with $\mu = 1$ for variable h.

h	Re(β)	MAXERR
10.0	-10.0	2.637875×10^{3}
8.0	-8.0	1.367025×10^{2}
5.0	-5.0	6.802028×10^{1}
3.0	-3.0	5.353147×10^{0}
1.0	-1.0	3.145508×10^{-2}
0.5	-0.5	8.457736×10^{-5}
0.1	-0.1	1.976034×10^{-9}
0.01	-0.01	4.166649×10^{-10}

6 Problems Tested and Numerical Results

The set of problems below are solve using DITDRK(2,4) and DITDRK(3,5) methods and the delay term are evaluated using Newton Divided Difference Interpolation (NDDI) of five and six points method respectively. Both methods are compared in term of their numerical performances with some famous existing DIRK methods in the scientific literature. Listed below are the test problems.

Problem 1(Schmitt[27])

$$y_1{}'(t) = y_2(t), \qquad y_2{}'(t) = -\frac{1}{2}y_1(t) + \frac{1}{2}y_1(t-\pi), \qquad 0 \le t \le 10,$$
$$y_1(t) = \sin(t), \qquad y_2(t) = \cos(t), \qquad t \le 0.$$

Exact solution is

$$y_1(t) = \sin(t), \qquad y_2(t) = \cos(t).$$

Problem 2(Radzi et al.[25])

$$y'(t) = -y(t - \frac{\pi}{2}), \qquad 0 \le t \le 10, \qquad y(t) = \sin(t), \qquad t \le 0.$$

Exact solution is $y(t) = \sin(t)$.

Problem 3(Ishak et al.[9])

$$y_1{}'(t) = -y_1(t - \frac{\pi}{2}), \qquad y_2{}'(t) = -y_2(t - \frac{\pi}{2}), \qquad 0 \le t \le 10,$$
$$y_1(t) = \sin(t), \qquad y_2(t) = \cos(t), \qquad t \le 0.$$

Exact solution is

$$y_1(t) = \sin(t), \qquad y_2(t) = \cos(t).$$

Problem 4(Ishak et al.[9])

$$y_1{}'(t) = y_2(t), \qquad y_2{}'(t) = -\frac{1}{2}y_1(t) - \frac{1}{2} + y_1\left(\frac{1}{2}t - \frac{\pi}{4}\right)^2, \qquad 2 \le t \le 10,$$
$$y_1(t) = \sin(t), \qquad y_2(t) = \cos(t), \qquad t \le 2.$$

Exact solution is

$$y_1(t) = \sin(t), \qquad y_2(t) = \cos(t).$$

Problem 5(Schmitt[27])

$$y_1{}'(t) = y_2(t), \qquad y_2{}'(t) = -\left(\frac{\sin(t)}{2-\sin(t)}\right)y_1(t-\pi), \qquad 0 \le t \le 10,$$
$$y_1(t) = 2 + \sin(t), \qquad y_2(t) = \cos(t), \qquad t \le 0.$$

Exact solution is

$$y_1(t) = 2 + \sin(t), \qquad y_2(t) = \cos(t).$$

Problem 6(Seong et al.[30])

$$y_1{}'(t) = y_2(t), \qquad y_2{}'(t) = -\frac{1}{2}y_1(t) + \frac{1}{2}y_1(t-\pi), \qquad 0 \le t \le 10,$$
$$y_1(t) = 1 - \sin(t), \qquad y_2(t) = \cos(t), \qquad t \le 0.$$

Exact solution is

$$y_1(t) = 1 - \sin(t), \qquad y_2(t) = \cos(t).$$

Figures 5–16 used the following abbreviations and represents the behaviour of these numerical results in graphics form.

- **DITDRK(2,4):** Two stages DITDRK method of order four derived previously. (Ahmad et al.[2])
- **DIRKLa(3,4):** Three stages DIRK method of order four. (Lambert[20])
- **DIRKJa(4,4)**: Four stages DIRK method of order four. (Jawias et al. [15])
- **DIRKFr(4,4)**: Four stages DIRK method of order four. (Franco and Gomez [7])
- **DIRKSa(3,4)**: Three stages DIRK method of order four. (Sanz-Serna and Abia [26])
- **DIRKKa(5,4)**: Five stages DIRK method of order four. (Kalogiratou and Monavasilis [16])
- **DITDRK(3,5):** Three stages DITDRK method of order five derived previously. (Ahmad et al.[2])
- **DIRKKa(6,5):** Six stages DIRK method of order five. (Kalogiratou and Monavasilis [16])
- **DIRKKa(7,5)**: Seven stages DIRK method of order five. (Kalogiratou and Monavasilis [16])
- **DIRKAb(5,5)**: Five stages DIRK method of order five. (Ababneh et al.[1])

7 Discussion

The numerical results above show the standard features of DITDRK(2,4) and DITDRK(3,5) methods. Several well-known of same order existing DIRK methods are chosen as the comparison with the proposed method. This method has a minimized local truncation error. In Figures 5(a)-16(a), the logarithm number of both maximum global error versus function evaluations are plotted for various problems. From Figures 5(a)-10(a), it is observed that global error produced by DITDRK(2,4) and DITDRK(3,5) method method have smaller global error compared to other existing same order DIRK methods.

Next, a long period of integration of the method's efficiency and global error are plotted. The log of the maximum global error versus the CPU time is plotted as given in Figures 5(b)-16(b) to show the accuracy of the designed method. In Figures 5(b) and 9(b), DITDRK(2,4) takes slightly longer CPU time compared to DIRKLa(3,4) due to its method complexity which is caused by the existence of the extra g to be evaluated at every step. Overall, it can be concluded that DITDRK(2,4) and DITDRK(3,5) method are superior compared to other existing method even with the similar order of interpolation used for the delay term.

Looking at the stability regions of the methods, we observed that both methods give almost the same regions of Q-stability but different regions of P-stability. A smaller maximum global error which converges to its exact solution indicates that the method is stable. Hence, the value of h can be obtained from the stability region so that the methods remain stable. This can be seen in the stability test which we have carried out earlier in Tables 3–6.

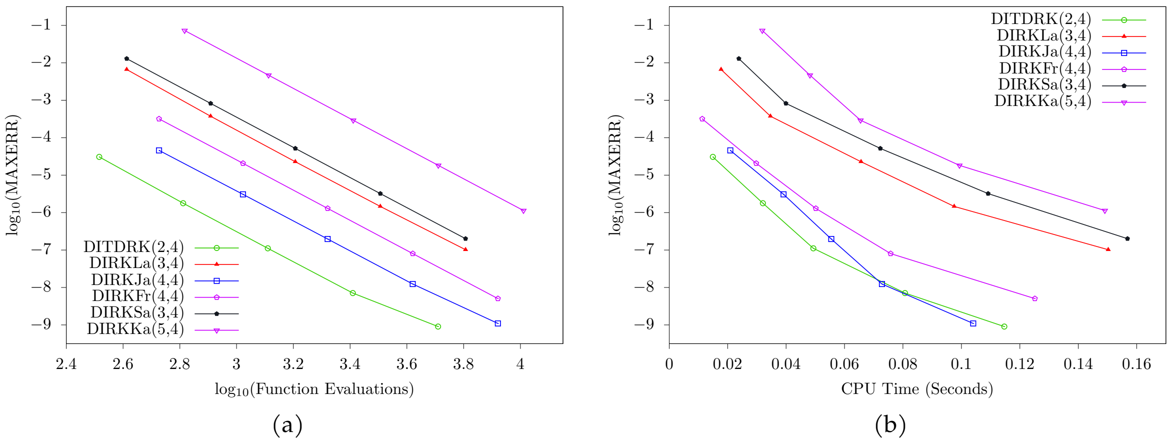


Figure 5: The error at each integration point for DITDRK(2,4) method when solving Problem 1 with $h = 0.5/2^i$ where $i = 1, \ldots, 5$.

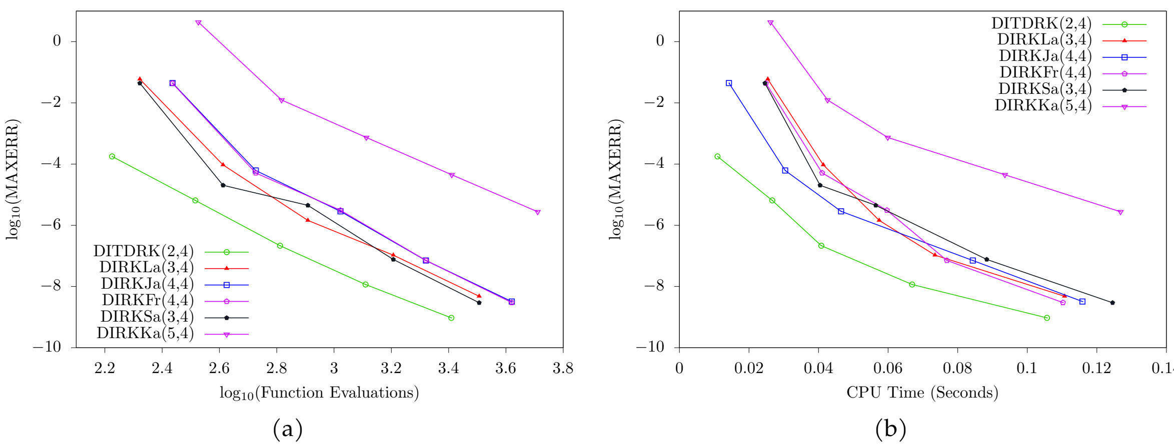
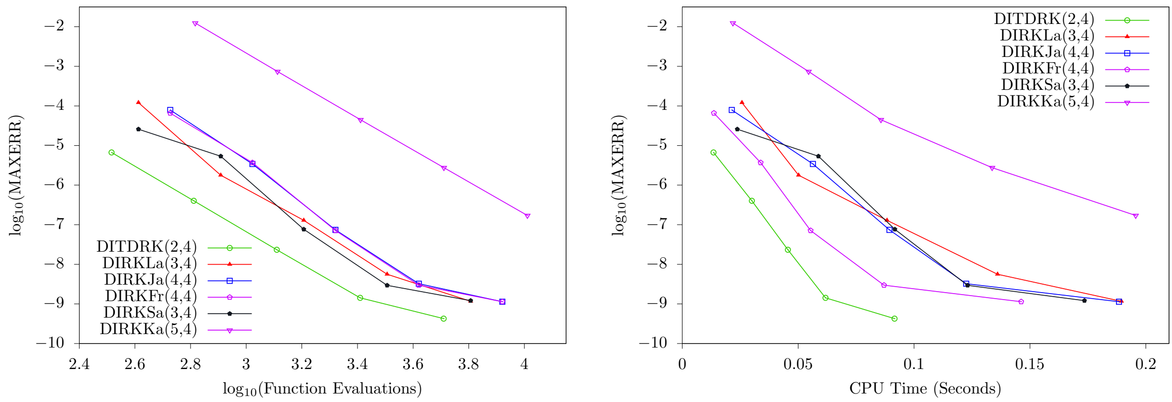
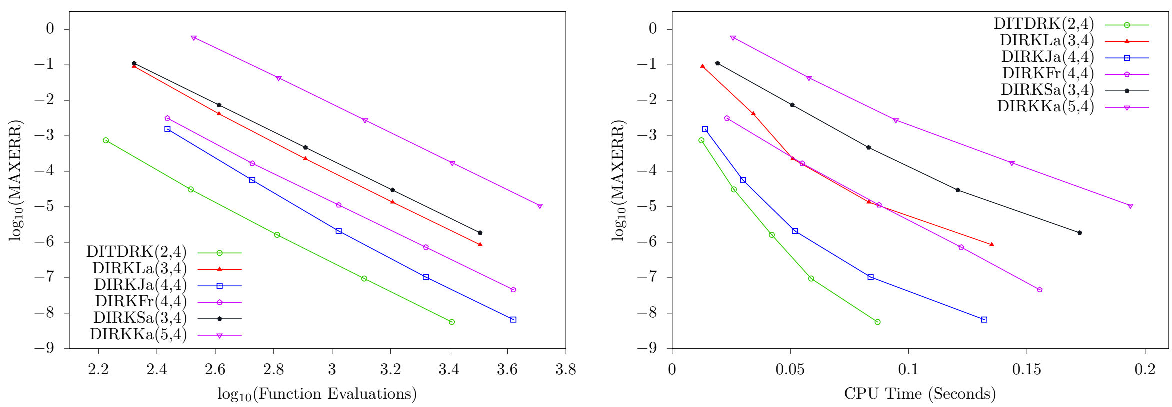


Figure 6: The error at each integration point for DITDRK(2,4) method when solving Problem 2 with $h = 1/2^i$ where $i = 1, \ldots, 5$.



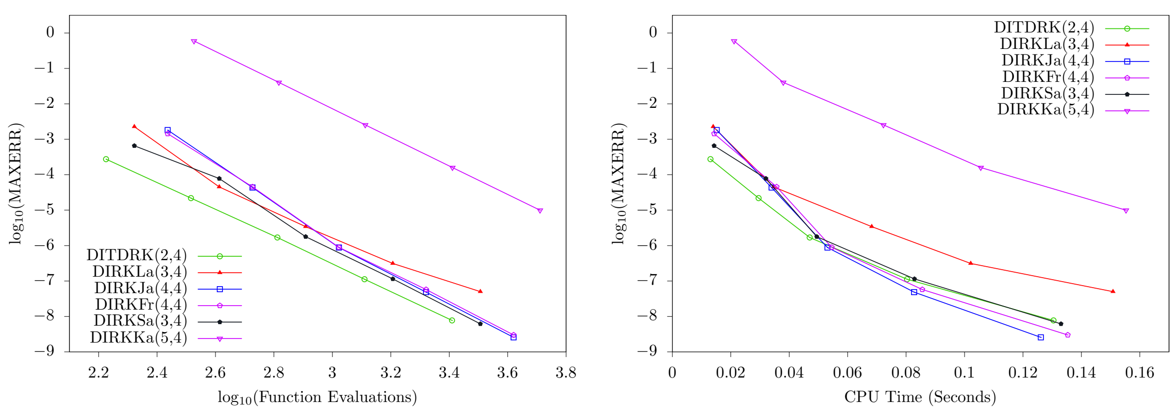
(a) (b)

Figure 7: The error at each integration point for DITDRK(2,4) method when solving Problem 3 with $h = 0.5/2^i$ where $i = 1, \dots, 5$.



(a) (b)

Figure 8: The error at each integration point for DITDRK(2,4) method when solving Problem 4 with $h = 1/2^i$ where $i = 1, \dots, 5$.



(a) (b)

Figure 9: The error at each integration point for DITDRK(2,4) method when solving Problem 5 with $h = 1/2^i$ where $i = 1, \dots, 5$.

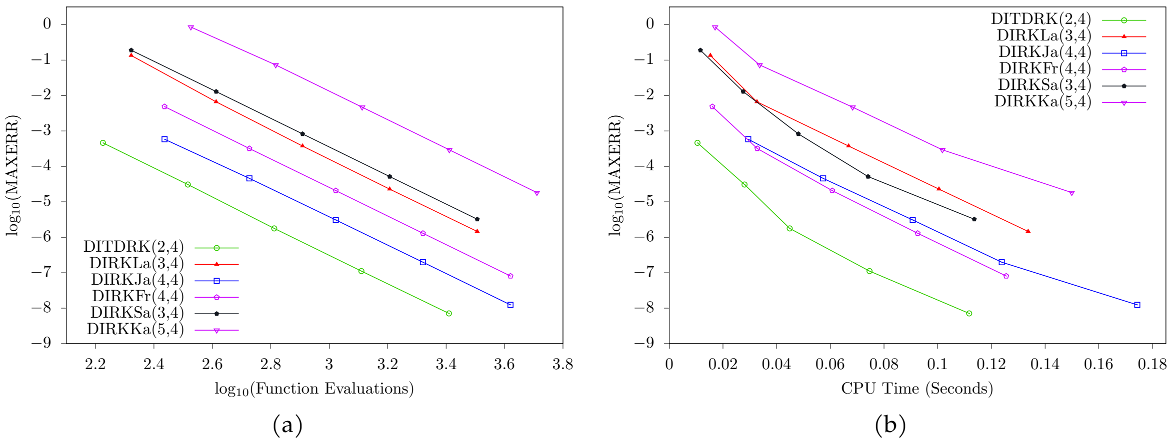


Figure 10: The error at each integration point for DITDRK(2,4) method when solving Problem 6 with $h = 1/2^i$ where $i = 1, \ldots, 5$.

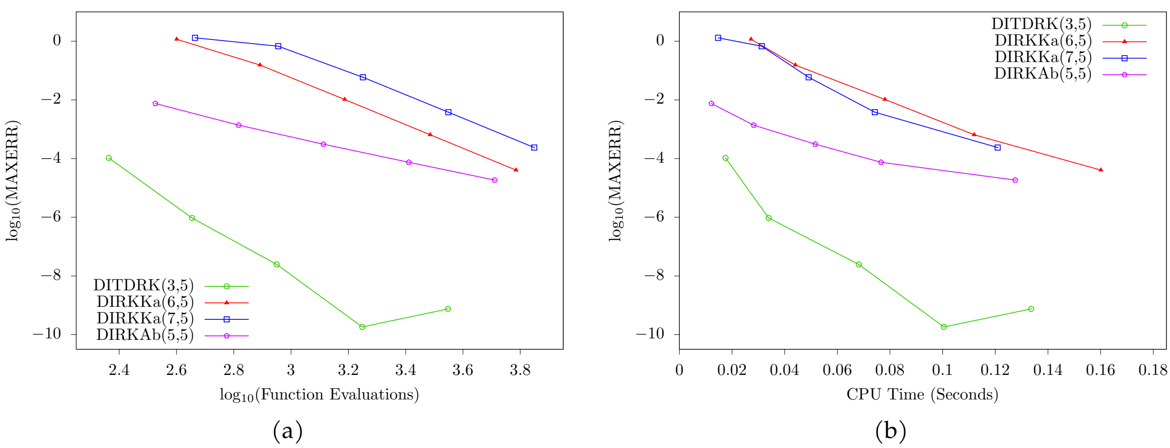


Figure 11: The error at each integration point for DITDRK(3,5) method when solving Problem 1 with $h = 1/2^i$ where $i = 1, \ldots, 5$.

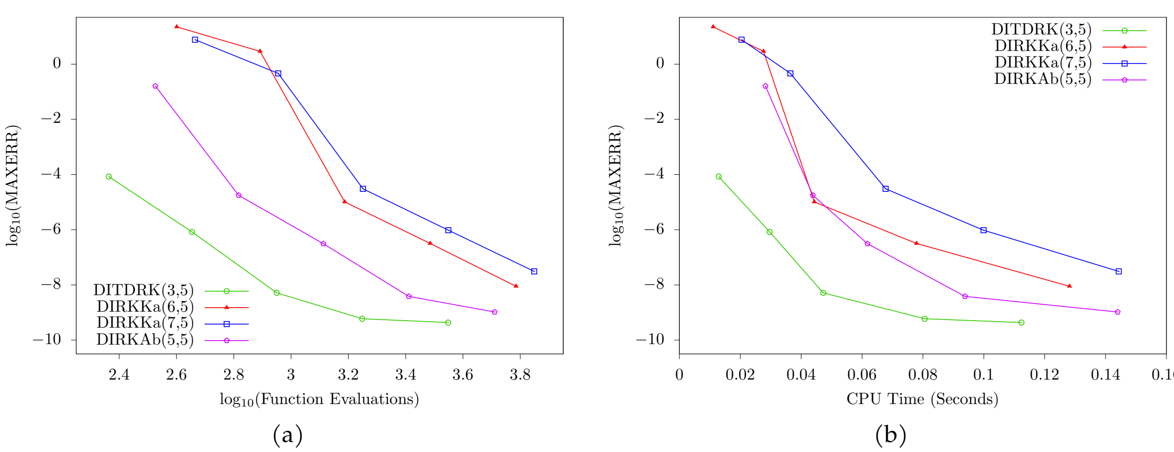


Figure 12: The error at each integration point for DITDRK(3,5) method when solving Problem 2 with $h = 1/2^i$ where $i = 1, \ldots, 5$.

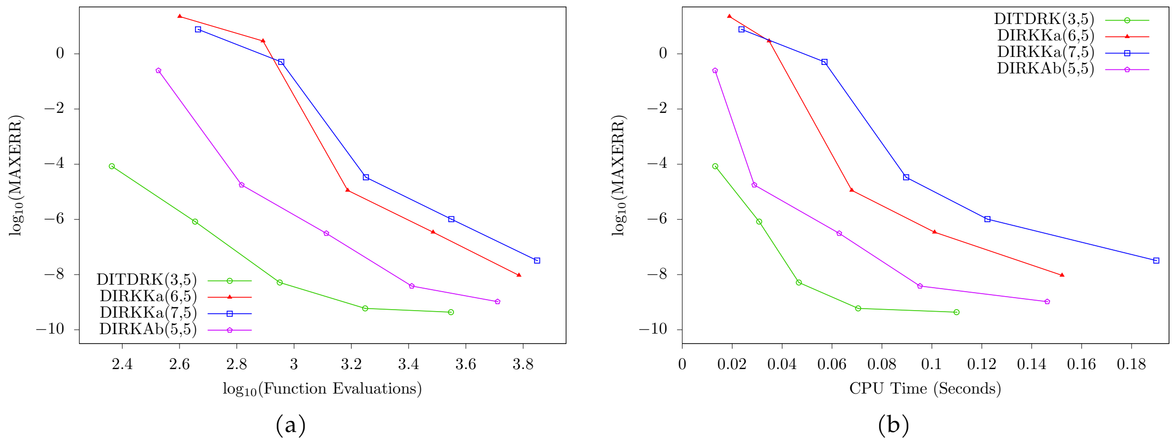


Figure 13: The error at each integration point for DITDRK(3,5) method when solving Problem 3 with $h = 1/2^i$ where $i = 1, \ldots, 5$.

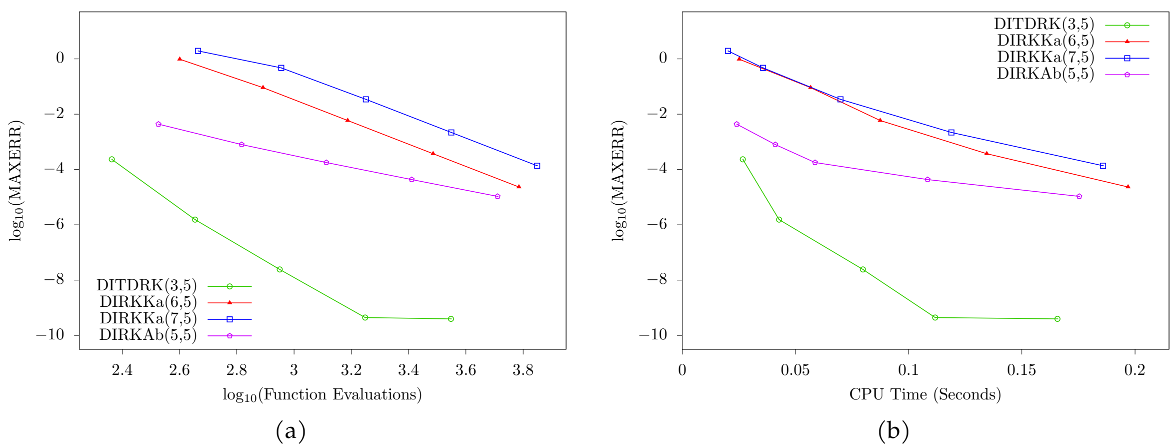


Figure 14: The error at each integration point for DITDRK(3,5) method when solving Problem 4 with $h = 1/2^i$ where $i = 1, \ldots, 5$.

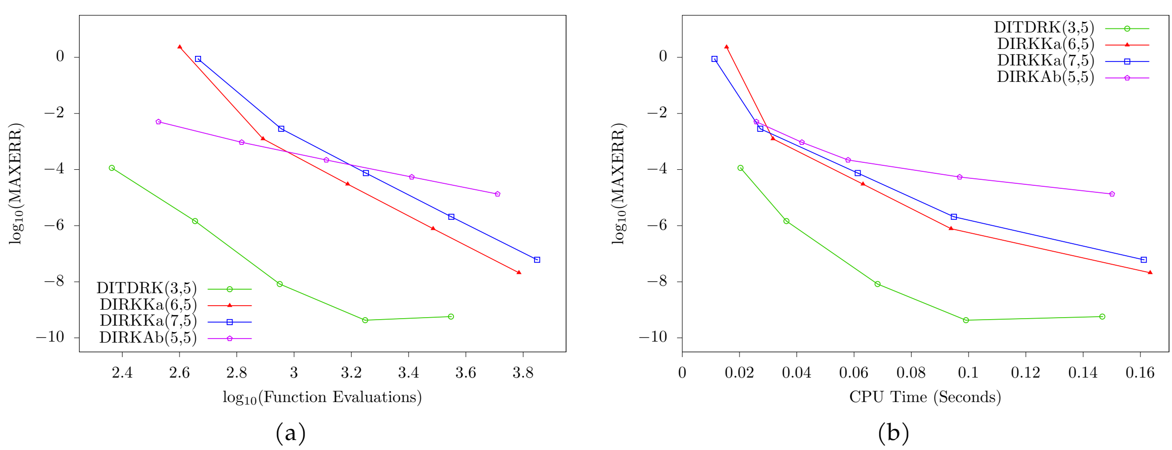
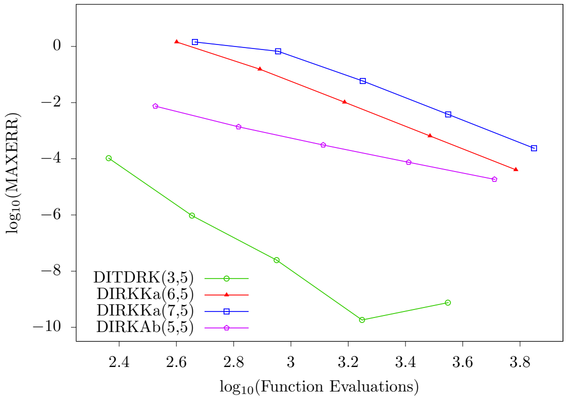


Figure 15: The error at each integration point for DITDRK(3,5) method when solving Problem 5 with $h = 1/2^i$ where $i = 1, \ldots, 5$.



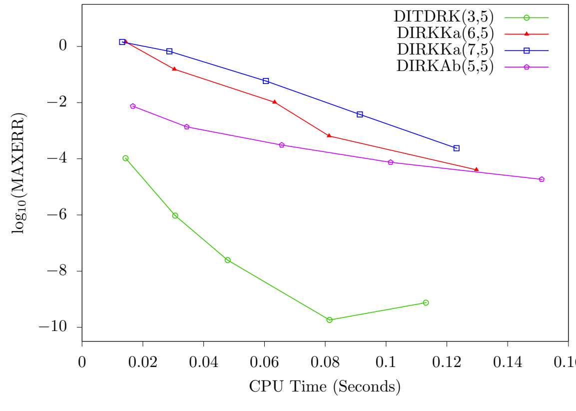


Figure 16: The error at each integration point for DITDRK(3,5) method when solving Problem 6 with $h = 1/2^i$ where $i = 1, \dots, 5$.

8 Conclusions

In this research, the P-stability and Q-stability of fourth and fifth-order DITDRK method have been investigated and their stability regions have been plotted by locating their boundary for each type of stability.

According to the numerical results gathered from the numerical experiments, it can be said that DITDRK(2,4) and DITDRK(3,5) methods are more promising compared to other excellently-known DIRK methods in terms of efficiency and accuracy as well as the total number of function evaluations. Referring to the stability regions plotted previously, these results are expected as the values of λ and μ for the set of problems above lie within the P-stability regions of the method.

Acknowledgement This study was fiscally funded by the Fundamental Research Grant Scheme (Ref. No. FRGS/1/2018/STG06/UPM/02/2) awarded by the Ministry of Education Malaysia.

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

References

[1] O. Y. Ababneh, R. Ahmad & E. S. Ismail (2009). Design of new diagonally implicit Runge-Kutta methods for stiff problems. *Applied Mathematical Sciences*, *3*(45), 2241–2253.

[2] N. A. Ahmad, N. Senu, Z. B. Ibrahim & M. Othman (2019). Diagonally implicit two derivative Runge-Kutta methods for solving first order initial value problems. *Journal of Abstract and Computational Mathematics*, *4*(1), 18–36.

[3] A. N. Al-Mutib (1984). Stability properties of numerical methods for solving delay differential equations. *Journal of Computational and Applied Mathematics*, *10*(1), 71–79.

[4] Z. Bartoszewski & Z. Jackiewicz (2002). Stability analysis of two-step Runge-Kutta methods for delay differential equations. *Computers & Mathematics with Applications*, *44*(1-2), 83–93. https://doi.org/10.1016/S0898-1221(02)00131-1.

[5] V. K. Barwell (1975). Special stability problems for functional differential equations. *BIT Numerical Mathematics*, *15*(2), 130–135. https://doi.org/10.1007/BF01932685.

[6] J. Fang & R. Zhan (2021). High order explicit exponential Runge-Kutta methods for semilinear delay differential equations. *Journal of Computational and Applied Mathematics*, *388*, 113279(1)–113279(16). https://doi.org/10.1016/j.cam.2020.113279.

[7] J. Franco & I. Gómez (2003). Fourth-order symmetric DIRK methods for periodic stiff problems. *Numerical Algorithms*, *32*(2-4), 317–336. https://doi.org/10.1023/A:1024077930017.

[8] K. In't Hout (1992). A new interpolation procedure for adapting Runge-Kutta methods to delay differential equations. *BIT Numerical Mathematics*, *32*(4), 634–649. https://doi.org/10.1007/BF01994847.

[9] F. Ishak, Z. A. Majid & M. Suleiman (2010). Two-point block method in variable stepsize technique for solving delay differential equations. *Journal of Materials Science and Engineering*, *4*(12), 86–90. .

[10] F. Ishak, M. Suleiman & Z. Omar (2008). Two-point predictor-corrector block method for solving delay differential equations. *MATEMATIKA: Malaysian Journal of Industrial and Applied Mathematics*, *24*(2), 131–140.

[11] F. Ismail, R. A. Al-Khasawneh, M. Suleiman et al. (2002). Numerical treatment of delay differential equations by Runge-Kutta method using Hermite interpolation. *MATEMATIKA: Malaysian Journal of Industrial and Applied Mathematics*, *18*(2), 79–90.

[12] F. Ismail & M. B. Suleiman (2000). The P-stability and Q-stability of singly diagonally implicit Runge-Kutta method for delay differential equations. *International Journal of Computer Mathematics*, *76*(2), 267–277. https://doi.org/10.1080/00207160008805023.

[13] N. Ismail, Z. Majid & N. Senu (2020). Solving neutral delay differential equation of pantograph type. *Malaysian Journal of Mathematical Sciences* (*ICoAIMS2019*), *14*(S), 107–121.

[14] N. T. Jaaffar, Z. A. Majid & N. Senu (2021). Numerical computation of third order delay differential equations by using direct multistep method. *Malaysian Journal of Mathematical Sciences*, *15*(3), 369–385.

[15] C. Jawias, N. Izzati, F. Ismail, M. Suleiman & A. Jaafar (2010). Fourth order four-stage diagonally implicit Runge-Kutta method for linear ordinary differential equations. *Malaysian Journal of Mathematical Sciences*, *4*(1), 95–105. .

[16] Z. Kalogiratou, T. Monovasilis & T. Simos (2012). Diagonally Implicit Symplectic Runge-Kutta methods with special properties. In *AIP Conference Proceedings*, pp. 1387–1390. American Institute of Physics, Kos, Greece.

[17] A. Karoui (1992). *On The Numerical Solution of Delay Differential Equations*. Master thesis, University of Ottawa, Canada.

[18] C. D. Kumar & A. E. K. Pushpam (2019). Higher order derivative Runge-Kutta method for solving delay differential equations. *Advances in Mathematics: Scientific Journal* (*Special Issue on ICRAPAM*), *8*(3), 26–34.

[19] C. D. Kumar & A. E. K. Pushpam (2020). Two-stage multiderivative explicit Runge-Kutta method for delay differential equations. *Advances in Mathematics: Scientific Journal*, *9*(3), 1293–1299.

[20] J. D. Lambert (1991). *Numerical methods for ordinary differential systems: The initial value problem*. John Wiley & Sons, Inc., Baffins Lane, England.

[21] S. Maset (2000). Stability of Runge-Kutta methods for linear delay differential equations. *Numerische Mathematik*, *87*(2), 355–371. http://doi.org/10.1007/s002110000179.

[22] M. Mechee, F. Ismail, N. Senu & Z. Siri (2013). Directly solving special second order delay differential equations using Runge-Kutta-Nyström method. *Mathematical Problems in Engineering*, *2013*, Article ID: 830317, 7 pages. https://doi.org/10.1155/2013/830317.

[23] K. W. Neves (1981). Control of interpolatory error in retarded differential equations. *ACM Transactions on Mathematical Software* (*TOMS*), *7*(4), 421–444. https://doi.org/10.1145/355972.355974.

[24] H. J. Oberle & H. J. Pesch (1981). Numerical treatment of delay differential equations by Hermite interpolation. *Numerische Mathematik*, *37*(2), 235–255. https://doi.org/10.1007/BF01398255.

[25] H. M. Radzi, Z. A. Majid, F. Ismail & M. Suleiman (2012). Two and three point one-step block methods for solving delay differential equations. *Journal of Quality Measurement and Analysis*, *8*(1), 29–41.

[26] J. M. Sanz-Serna & L. Abia (1991). Order conditions for canonical Runge-Kutta schemes. *SIAM Journal on Numerical Analysis*, *28*(4), 1081–1096. https://www.jstor.org/stable/2157788.

[27] K. Schmitt (1971). Comparison theorems for second order delay differential equations. *The Rocky Mountain Journal of Mathematics*, *1*(3), 459–467. https://www.jstor.org/stable/44236205.

[28] J. V. Shaalini & A. E. K. Pushpam (2020). Generalized rational multi-step method for delay differential equations. *IAENG International Journal of Applied Mathematics*, *50*(1), 1–9.

[29] O. A. Taiwo & O. S. Odetunde (2010). On the numerical approximation of delay differential equations by a decomposition method. *Asian Journal of Mathematics and Statistics*, *3*(4), 237–243. https://doi.org/10.3923/ajms.2010.237.243.

[30] H. Yann Seong, Z. Abdul Majid & F. Ismail (2013). Solving second-order delay differential equations by direct Adams-Moulton method. *Mathematical Problems in Engineering*, *2013*, Article ID: 261240, 7 pages. https://doi.org/10.1155/2013/261240.

[31] M. Zennaro (1986). P-stability properties of Runge-Kutta methods for delay differential equations. *Numerische Mathematik*, *49*(2), 305–318. https://doi.org/10.1007/BF01389632.