

# Research Article

# Directly Solving Special Second Order Delay Differential Equations Using Runge-Kutta-Nyström Method

# M. Mechee,<sup>1,2</sup> F. Ismail,<sup>3</sup> N. Senu,<sup>3</sup> and Z. Siri<sup>1</sup>

<sup>1</sup> Institute of Mathematical Sciences, University of Malaya, 50603 Kuala Lumpur, Malaysia

<sup>2</sup> Department of Mathematics, Faculty of Mathematics and Computer Sciences, University of Kufa, Najaf, Iraq

<sup>3</sup> Department of Mathematics and Institute for Mathematical Research, Universiti Putra Malaysia (UPM), 43400 Serdang, Selangor, Malaysia

Correspondence should be addressed to M. Mechee; mohsabd@siswa.um.edu.my

Received 31 December 2012; Revised 16 April 2013; Accepted 16 April 2013

Academic Editor: Yang Yi

Copyright © 2013 M. Mechee et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Runge-Kutta-Nyström (RKN) method is adapted for solving the special second order delay differential equations (DDEs). The stability polynomial is obtained when this method is used for solving linear second order delay differential equation. A standard set of test problems is solved using the method together with a cubic interpolation for evaluating the delay terms. The same set of problems is reduced to a system of first order delay differential equations and then solved using the existing Runge-Kutta (RK) method. Numerical results show that the RKN method is more efficient in terms of accuracy and computational time when compared to RK method. The methods are applied to a well-known problem involving delay differential equations, that is, the Mathieu problem. The numerical comparison shows that both methods are in a good agreement.

## 1. Introduction

A special second order differential equations (ODEs) of the form

$$y''(t) = f(t, y(t)), \quad y(t_0) = \alpha, \quad y'(t_0) = \beta, \quad t \ge t_0$$
(1)

which is not explicitly dependent on the first derivative of the solution are frequently found in many physical problems such as electromagnetic waves, thin film flow, and gravity driven flow. Most researchers, scientists, and engineers used to solve (1) by converting the second order differential equations to a system of first order equations twice the dimension. However there are also studies on numerical methods which directly solve (1) using one-step methods or multistep methods. Such work can be seen in [1–6].

Most of the methods for solving special second order ODEs can be adapted for solving special second order delay differential equations (DDEs). In recent years there has been a growing interest in numerical solutions of DDEs. This is due to the appearance of such equations in various areas such as neural network theory, epidemiology, and time lag control processes. DDEs also provide us with realistic model of many phenomena arising in real world problems. For example, DDEs can be used in modelling of population dynamics and spread of infectious diseases and two body problems of electrodynamics [7–12]. Most of the work concerning DDEs in the literature involved first order delay differential equations. Hence, in this research, we are going to focus on numerical methods for solving special second order DDEs. Special second order delay differential equations with multiple delays can be written in the following form:

$$y''(t) = f(t, y(t), y(t - \tau_1), y(t - \tau_2), ..., y(t - \tau_n)),$$
  

$$t > t_0$$
(2)

with initial conditions

$$y(t) = \varphi(t), \quad y'(t) = \varphi'(t), \quad t \le t_0.$$
 (3)

Runge-Kutta-Nyström method of order four will be adapted for solving second order DDEs (2). Stability polynomial of the method is also presented when applied to linear second order DDE. Numerical results on a set of test problems are given and compared with the numerical results when the problems are reduced to a system of first order DDEs and solved using Runge-Kutta methods. We are also going to solve a well-known problem in engineering which involves second order delay differential equations, that is, the Mathieu problem.

### 2. Numerical Methods for Second Order ODEs

Runge-Kutta methods are designed for special second order differential equations:

$$y'' = f(t, y(t)), \quad y(t_0) = \alpha, \quad y'(t_0) = \beta, \quad t > t_0$$
(4)

and are usually termed as Runge-Kutta-Nyström formula (RKN) since their introduction in 1925 by Nyström. An *s*-stage RKN method for the numerical integration of the IVP in (4) is given by

$$y_{n+1} = y_n + hy'_n + h^2 \sum_{i=1}^{s} b_i k_i,$$

$$y'_{n+1} = y'_n + h \sum_{i=1}^{s} b'_i k_i,$$
(5)

where

$$k_{i} = f\left(t_{n} + c_{i}h, y_{n} + c_{i}hy_{n}' + h^{2}\sum_{j=1}^{s}a_{ij}k_{j}\right)$$
(6)

for i = 1, 2, ..., s. The RKN parameters  $c_j, a_{ij}, b_j$ , and  $b'_j$  are assumed to be real for i, j = 1, 2, 3, ..., s and s is the number of stages of the method. Let us introduce the *s*-dimensional vectors c, b, and b'; moreover, the matrix A is  $s \times s$ , where  $c = [c_1, c_2, c_3, ..., c_s]$ ,  $b = [b_1, b_2, b_3, ..., b_s]$ ,  $b' = [b'_1, b'_2, b'_3, ..., b'_s]$ , and  $A = [a_{ij}]$ , respectively. RKN method can be expressed in Butcher notation using the table of coefficients as follows:

$$\begin{array}{c|c}c & A \\ \hline b^T \\ b'^T \\ b'^T \end{array} \tag{7}$$

RKN methods can be divided into two classes explicit methods when  $a_{ij} = 0$  for  $i \le j$  and implicit methods otherwise.

#### 3. Runge-Kutta-Nyström for DDEs

Consider second order DDE

$$y'' = f(t, y(t), y(t - \tau_1), y(t - \tau_2), y(t - \tau_3), \dots, y(t - \tau_m)), \quad t \ge t_0$$
(8)

with initial conditions

$$y(t) = \varphi(t), \quad y'(t) = \varphi'(t), \quad t \le t_0$$
 (9)

 $y_n$  is the numerical solution of (8) for all  $t_n > t_0$ ,  $n = 1, 2, 3, \dots$ 

The RKN for second order ODE has been adapted to solve second order DDE and the formula can be written as follows:

$$y_{n+1} = y_n + hy'_n$$

$$+ h^2 \sum_{i=1}^{s} b_i k_i \left( t_n + c_i h, Y_i, y \left( t_n + c_i h - \tau_1 \right), y \left( t_n + c_i h - \tau_2 \right), \dots, y \left( t_n + c_i h - \tau_m \right) \right),$$

$$y'_{n+1} = y'_n$$

$$+ h \sum_{i=1}^{s} b'_i k_i \left( t_n + c_i h, Y_i, y \left( t_n + c_i h - \tau_1 \right), y \left( t_n + c_i h - \tau_2 \right), \dots, y \left( t_n + c_i h - \tau_m \right) \right),$$
(10)

where

$$k_{i} = f\left(t_{n} + c_{i}h, y_{n} + c_{i}hy_{n}' + h^{2}\sum_{j=1}^{s}a_{ij}k_{j}, y\left(x_{n} + c_{i}h - \tau_{1}\right), y\left(t_{n} + c_{i}h - \tau_{2}\right), \dots, y\left(t_{n} + c_{i}h - \tau_{m}\right)\right)$$
(11)

for i = 1, 2, ..., s. It can be written as follows:

$$y_{n+1} = y_n + hy'_n + h^2 \sum_{i=1}^{s} b_i f(t_n + c_i h, Y_i, z_i),$$

$$y'_{n+1} = y'_n + h \sum_{i=1}^{s} b'_i f(t_n + c_i h, Y_i, z_i),$$
(12)

where

$$Y_{i} = y_{n} + c_{i}hy_{i}' + h^{2}\sum_{j=1}^{s} a_{ij}k_{j}(t_{n} + c_{i}h, Y_{j}, z_{i}),$$

$$z_{i} = (y(t_{n} + c_{i}h - \tau_{1}), y(t_{n} + c_{i}h - \tau_{2}), \dots,$$

$$y(t_{n} + c_{i}h - \tau_{m})).$$
(13)

3.1. Time Delay Interpolation. Let the interval of the definition of delay differential equation (8) be I = [a, b] and  $t_i = a + ih$  for i = 0, 1, ..., n and h = (b - a)/n, where n is the number of points in interval I. The numerical method approximates the solutions at point  $t_i$  for i = 1, 2, ..., n, to approximate the solution  $y_{i+1}$  at point  $t_{i+1}$  for i = 0, 1, ..., n - 1. Hence,

$$y_{i+1} = y_{i+1} \left( t_i, y \left( t_i - \tau_1 \right), y \left( t_i - \tau_2 \right), \dots, y \left( t_i - \tau_m \right) \right).$$
(14)

The method of interpolation has the following cases.

*Case 1.* If time delay  $\tau_1, \tau_2, \ldots, \tau_m$  are constants and suppose  $\tau_{i_1} < \tau_{i_2} < \cdots < \tau_{i_m}$  for  $i_1, i_2, \ldots, i_m = 1, 2, \ldots, m$ , therefore we can interpolate  $y(t_i - \tau_1), y(t_i - \tau_2), \ldots, y(t_i - \tau_m)$  as follows:

$$y(t_{i} - \tau_{1}) = f_{\tau_{1}}(t_{i-d}, t_{i-d+1}, \dots, t_{i}, y(t_{i-d}), y(t_{i-d+1}), \dots, y(t_{i})),$$

$$y(t_{i} - \tau_{2}) = f_{\tau_{2}}(t_{i-d+1}, t_{i-d+2}, \dots, t_{i}, y(t_{i-d+1}), y(t_{i-d+2}), \dots, y(t_{i}), y(t_{i} - \tau_{1})),$$

$$y(t_{i} - \tau_{3}) = f_{\tau_{3}}(t_{i-d+2}, t_{i-d+3}, \dots, t_{i}, y(t_{i-d+2}), y(t_{i-d+3}), \dots, y(t_{i}), y(t_{i} - \tau_{1}), y(t_{i} - \tau_{2})),$$

$$\vdots \qquad \vdots \qquad \vdots$$

$$y(t_{i} - \tau_{d}) = f_{\tau_{d}}(t_{i-1}, t_{i}, y(t_{i-1}), y(t_{i}), y(t_{i} - \tau_{1}), y(t_{i} - \tau_{2}), \dots, y(t_{i} - \tau_{2}), \dots, y(t_{i} - \tau_{d-1})),$$

$$y(t_{i} - \tau_{j}) = f_{\tau_{j}}(t_{i}, y(t_{i}), y(t_{i} - \tau_{1}), y(t_{i} - \tau_{2}), \dots, y(t_{i} - \tau_{d})), \dots, y(t_{i} - \tau_{d})),$$

$$m \ge j \ge d.$$
(15)

The functions  $f_{\tau_j}$  for j = 1, 2, ..., m depend on the interpolation which is used in the numerical method which has degree *d*.

*Case 2.* If  $\tau_1, \tau_2, ..., \tau_m$  are the variables of time delays, that is,  $\tau_j = \tau_j(t)$  for j = 1, 2, ..., m, however, we can consider

$$y(t_{i} - \tau_{j}) = f_{\tau_{j}}(t_{i-d}, t_{i-d+1}, \dots, t_{i}, y(t_{i-d}), y(t_{i-d+1}), \dots, y(t_{i}))$$
(16)

for j = 1, 2, ..., m, knowing that  $y_k = y(t_k)$  for k = i - d, i - d + 1, ..., i. The function  $f_{\tau_j}$  depends on the type and degree of the interpolation. In this paper, we consider the cubic interpolation to approximate DDEs with three time delays.

#### 4. Stability of the Method

Stability aspect of numerical methods for delay differential equations has been introduced in [13–17]. To study the stability of numerical method (10), consider the linear test equation

$$y'' = \lambda^2 y(t) + \mu^2 y(t - \tau).$$
 (17)

When the method is applied to the linear test equation (17), we have

$$y_{n+1} = y_n + hy'_n + h^2 \sum_{i=1}^{s} b_i f(t_n + c_i h, Y_i, y(t_n + c_i h - \tau))$$
  
=  $y_n + hy'_n + h^2 \sum_{i=1}^{s} b_i (\lambda^2 Y_i + \mu^2 y(t_n + c_i h - \tau)),$ 

$$y'_{n+1} = y'_{n} + h \sum_{i=1}^{s} b'_{i} k_{i} \left( t_{n} + c_{i}h, Y_{i}, y \left( t_{n} + c_{i}h - \tau \right) \right)$$
$$= y'_{n} + h \sum_{i=1}^{s} b'_{i} k_{i} \left( \lambda^{2} Y_{i} + \mu^{2} y \left( t_{n} + c_{i}h - \tau \right) \right),$$
(18)

where

$$Y_{i} = y_{n} + c_{i}hy_{n}' + h^{2}\sum_{j=1}^{s} a_{ij}f\left(t_{n} + c_{i}h, Y_{j}, y\left(t_{n} + c_{i}h - \tau\right)\right)$$
  
$$= y_{n} + c_{i}hy_{n}' + h^{2}\sum_{j=1}^{s} a_{ij}\left(\lambda^{2}Y_{j} + \mu^{2}y\left(t_{n} + c_{i}h - \tau\right)\right),$$
  
$$Z_{n+1} = TZ_{n} + \lambda^{2}h^{2}BY + \mu^{2}h^{2}BZ_{n}\left(\tau\right)$$
(19)

such that

$$Z_{n} = \begin{pmatrix} y_{n} \\ hy'_{n} \end{pmatrix}, \quad T = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}, \quad B = \begin{pmatrix} b_{1} & b_{2} & \cdots & b_{s} \\ b'_{1} & b'_{2} & \cdots & b'_{s} \end{pmatrix},$$
$$Z_{n}(\tau) = \begin{pmatrix} y(t_{n} + c_{1}h - \tau) \\ y(t_{n} + c_{2}h - \tau) \\ \vdots \\ y(t_{n} + c_{s}h - \tau) \end{pmatrix}.$$
(20)

So

$$\begin{pmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_s \end{pmatrix} = \begin{pmatrix} y_n \\ y_n \\ \vdots \\ y_n \end{pmatrix} + \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_s \end{pmatrix} hy'_n$$

$$+ h^2 \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1s} \\ a_{21} & a_{22} & \cdots & a_{2s} \\ \vdots & \ddots & \vdots \\ a_{s1} & a_{s2} & \cdots & a_{ss} \end{pmatrix}$$

$$\times \begin{pmatrix} \lambda^2 Y_1 + \mu^2 y (t_n + c_1 h - \tau) \\ \lambda^2 Y_2 + \mu^2 y (t_n + c_2 h - \tau) \\ \vdots \\ \lambda^2 Y_s + \mu^2 y (t_n + c_s h - \tau) \end{pmatrix}$$

$$= \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix} y_n + \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_s \end{pmatrix} hy'_n$$

$$+ h^2 \lambda^2 A \begin{pmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_s \end{pmatrix}$$

$$+ \mu^{2} h^{2} A \begin{pmatrix} \mu^{2} y (t_{n} + c_{1} h - \tau) \\ \mu^{2} y (t_{n} + c_{2} h - \tau) \\ \vdots \\ \mu^{2} y (x_{n} + c_{s} h - \tau) \end{pmatrix}.$$
(21)

This implies that

$$(I - H_{\lambda}A)\begin{pmatrix}Y_{1}\\Y_{2}\\\vdots\\Y_{s}\end{pmatrix} = \begin{pmatrix}1 & c_{1}\\1 & c_{2}\\\vdots & \vdots\\1 & c_{s}\end{pmatrix}Z_{n}$$
$$+ H_{\mu}A\begin{pmatrix}\mu^{2}y(t_{n} + c_{1}h - \tau)\\\mu^{2}y(t_{n} + c_{2}h - \tau)\\\vdots\\\mu^{2}y(x_{n} + c_{s}h - \tau)\end{pmatrix},$$
(22)

where  $H_{\lambda} = (\lambda h)^2$ ,  $H_{\mu} = (\mu h)^2$ . So

$$\begin{pmatrix} Y_{1} \\ Y_{2} \\ \vdots \\ Y_{s} \end{pmatrix} = (I - H_{\lambda}A)^{-1} \times \left( CZ_{n} + H_{\mu}A \begin{pmatrix} \mu^{2}y(t_{n} + c_{1}h - \tau) \\ \mu^{2}y(t_{n} + c_{2}h - \tau) \\ \vdots \\ \mu^{2}y(t_{n} + c_{s}h - \tau) \end{pmatrix} \right),$$
(23)

where

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1s} \\ a_{21} & a_{22} & \cdots & a_{2s} \\ \vdots & \ddots & \vdots \\ a_{s1} & a_{s2} & \cdots & a_{ss} \end{pmatrix},$$

$$C = \begin{pmatrix} 1 & c_1 \\ 1 & c_2 \\ \vdots & \vdots \\ 1 & c_s \end{pmatrix}.$$
(24)

Hence, the stability polynomial of the method is

$$\begin{split} Z_{n+1} &= TZ_n + H_{\lambda} B \big( I - H_{\lambda} A \big)^{-1} \\ &\times \big( CZ_n + H_{\mu} A Z_n \left( \tau \right) \big) + H_{\mu} B Z_n \left( \tau \right), \\ Z_{n+1} &= T_1 Z_n + T_2 Z_n \left( \tau \right), \end{split}$$

$$Z_{n}(\tau) = \begin{pmatrix} y(t_{n} + c_{1}h - \tau) \\ y(t_{n} + c_{2}h - \tau) \\ \vdots \\ y(x_{n} + c_{s}h - \tau) \end{pmatrix},$$
  

$$T_{1} = T + H_{\lambda}B(I - H_{\lambda}A)^{-1}C,$$
  

$$T_{2} = H_{\mu}B\left[H_{\lambda}(I - H_{\lambda}A)^{-1}A + I\right].$$
(25)

# 5. Numerical Results

In this section, some of the problems involving second order DDEs are solved using RKN methods. Then the same set of problems is reduced to a first order DDEs system and solved using RK methods of the same order. Then numerical results are given.

The following notations are used in Figures 1, 2, 3, 4, 5, and 6:

- (i) *h*: stepsize used.
- (ii) *RKN4-N*: the existing Rung-Kutta-Nyrström method of order four derived by Senu et al. [18].
- (iii) *RKN4-D*: the existing Rung-Kutta-Nyrström method of order four as in [19].
- (iv) *RK4*: existing Runge-Kutta method order four as given in [20].
- (v) DOPRI: existing Runge-Kutta method order fifth as in Dormand [19].
- (vi) *Total time*: the total time in second to solve the problems.
- (vii) *MAX ERROR*:  $Max_n|y(x_n) y_n|$  Absolute value of the true solution minus the computed solution.

Problem 1 (nonlinear). Consider the following:

$$y''(t) = e^{-2\tau} \frac{y^2(t-\tau)}{y(t)}, \quad t \ge t_0,$$
  
$$y(t) = e^{-t}, \quad y'(t) = -e^{-t}, \quad t \le t_0,$$
  
$$\tau = \frac{h}{10}.$$
 (26)

Exact solution:  $y(t) = e^{-t}$ .

Problem 2 (nonlinear). Consider the following:

$$y''(t) = y^{2}(t-\tau) - \frac{1}{4\sqrt{(1+t)^{3}}} - (1+t) + \tau, \quad t \ge t_{0},$$
$$y(t) = \sqrt{1+t}, \quad y'(t) = \frac{1}{2\sqrt{1+t}}, \quad t \le t_{0},$$
$$\tau = \frac{h}{10}.$$
(27)

Exact solution:  $y(t) = \sqrt{1+t}$ .



FIGURE 1: The efficiency curves for all methods for Problem 1 with  $t_{end} = 1$  and  $h = 1/4^i$ , i = 1, ..., 5.



FIGURE 2: The efficiency curves for all methods for Problem 2 with  $t_{end} = 1$  and  $h = 1/4^i$ , i = 1, ..., 5.

Problem 3 (linear example). Consider the following:

$$y''(t) = y(t - \tau) - \frac{1}{(1 + t)^2} - \ln(1 + t - \tau), \quad t \ge t_0,$$
$$y(t) = \ln(1 + t), \quad y'(t) = \frac{1}{1 + t}, \quad t \le t_0,$$
$$\tau = \frac{h}{10}.$$
(28)



FIGURE 3: The efficiency curves for all methods for Problem 3 with  $t_{end} = 1$  and  $h = 1/4^i$ , i = 1, ..., 5.



FIGURE 4: The efficiency curves for all methods for Problem 4 with  $t_{end} = 1$  and  $h = 1/4^i$ , i = 1, ..., 5.

Problem 4 (nonlinear). Consider the following:

$$y''(t) = \frac{1}{3} \left( e^{-\tau_1} y \left( t - \tau_1 \right) + e^{-\tau_2} y \left( t - \tau_2 \right) \right. \\ \left. + e^{-\tau_3} y \left( t - \tau_3 \right) \right), \quad t \ge t_0, \\ y(t) = e^{-t}, \quad y'(t) = -e^{-t}, \quad t \le t_0, \\ \tau_1 = \frac{h}{10}, \qquad \tau_2 = \frac{h}{20}, \qquad \tau_3 = \frac{h}{30}.$$

$$(29)$$

Exact solution:  $y(t) = e^{-t}$ .

5.1. An Application to Mathieu Equation. In this section we will apply the RKN and RK methods to solve a well-known

Exact solution:  $y(t) = \ln (1 + t)$ .



FIGURE 5: The numerical solution for all methods for Linear Problem with  $t_{end} = 10$  and h = 1/10.



FIGURE 6: The numerical solution for all methods for Nonlinear Problem with  $t_{end} = 10$  and h = 1/10.

equation in engineering, the Matheiu's equation, which is defined as follows:

$$y''(t) + (\delta + a\cos t) y(t) + cy^{3}(t) = by(t - T), \quad (30)$$

which is a nonlinear delay differential equation.

Where  $\delta$ , *a*, *b*, *c*, and *T* are parameters.

 $\delta$  is the frequency squared of the simple harmonic oscillator, and *a* is the amplitude of the parametric resonance, and *b* is the amplitude of delay while *c* is the amplitude of the cubic nonlinearity and *T* is the time delay.

Equation (30) is a model for high speed milling, a kind of parametrically interrupted cutting as opposed to the selfinterrupted cutting arising in an unstable turning process. More information on the problem can be found in [21]. Various special cases of (30) have been studied, depending on which parameters are zero. When  $\delta = a = b = 1$  and c = 0 we obtained the following linear Mathieu equation:

$$y''(t) = (1 + \cos t) y(t) = y(t - T), \quad t \in [0, 10],$$
  
$$y(t) = \sin(t), \quad (31)$$
  
$$y'(t) = \cos(t), \quad t \le 0,$$

where  $T = \tau = h/10$  is the delay term, the exact solution does not exist.

When  $\delta = a = b = c = 1$ , we obtained the following nonlinear Mathieu equation:

$$y''(t) = (1 + \cos t) y(t) + y^{3}(t)$$
  
=  $y(t - T), \quad t \in [0, 10],$   
 $y(t) = \sin(t),$   
 $y'(t) = \cos(t), \quad t \le 0,$   
(32)

where  $T = \tau = h/10$  is the delay term, the exact solution does not exist.

Both the linear and nonlinear Mathieu equations are solved using RKN and RK methods and the results are plotted in Figures 5 and 6.

#### 6. Discussion and Conclusion

In this paper we adapted the Runge-Kutta-Nyström method for solving special second order delay differential equations in which cubic interpolation is used to evaluate the delay term. We also presented the stability of the method when applied to linear second order DDEs. We solved a set of DDEs using RKN4-N by Senu et al. [6] and RKN4-D by Dormand [19]. For comparison purposes the same set of problems a reduced to a system of first order DDEs and solved using classical fourth order Runge-Kutta method and fifth order Runge-Kutta method by Dormand [19]. The log of maximum error is plotted against time. From the numerical results, we observed that both RKN methods are just as efficient. However they are more efficient compared to the fourth order classical Runge-Kutta method followed by the fifth order Runge-Kutta method by Dormand [19]. This is because RKN method directly solved the equations whereas in RK method the equations are reduced to a system of first order DDEs. The fifth order RK method has more stages compared to the fourth order RK method which required more function evaluations at each step. Both RKN and RK methods are also used to solve the linear and nonlinear Mathieu equations. Since they do not have the exact solution, we just plot the numerical results in which both methods show a good agreement.

#### Acknowledgment

The authors would like to thank the reviewers for their constructive comments and their careful reading of the paper.

## References

- J. M. Franco, "A class of explicit two-step hybrid methods for second-order IVPs," *Journal of Computational and Applied Mathematics*, vol. 187, no. 1, pp. 41–57, 2006.
- [2] P. J. van der Houwen and B. P. Sommeijer, "Explicit Runge-Kutta (-Nyström) methods with reduced phase errors for computing oscillating solutions," *SIAM Journal on Numerical Analysis*, vol. 24, no. 3, pp. 595–617, 1987.
- [3] S. O. Imoni, F. O. Otunta, and T. R. Ramamohan, "Embedded implicit Runge-Kutta Nyström method for solving secondorder differential equations," *International Journal of Computer Mathematics*, vol. 83, no. 11, pp. 777–784, 2006.
- [4] S. N. Jator, "On a class of hybrid methods for y" = f(x, y, y')," *International Journal of Pure and Applied Mathematics*, vol. 59, no. 4, pp. 381–395, 2010.
- [5] P. J. van der Houwen and B. P. Sommeijer, "Diagonally implicit Runge-Kutta-Nyström methods for oscillatory problems," *SIAM Journal on Numerical Analysis*, vol. 26, no. 2, pp. 414–429, 1989.
- [6] N. Senu, M. Suleiman, and F. Ismail, "An embedded explicit rungekutta-nyström method for solving oscillatory problems," *Physica Scripta*, vol. 80, no. 1, Article ID 015005, 2009.
- [7] H. Smith, An Introduction to Delay Differential Equations with Applications to the Life Sciences, vol. 57 of Texts in Applied Mathematics, Springer, New York, NY, USA, 2011.
- [8] J. E. Forde, Delay differential equation models in mathematical biol-ogy [Ph.D. thesis], The University of Michigan, 2005.
- [9] A. Bellen and M. Zennaro, Numerical Methods for Delay Differential Equations, Oxford University Press, New York, NY, USA, 2003.
- [10] T. Erneux, Applied Delay Differential Equations, vol. 3 of Surveys and Tutorials in the Applied Mathematical Sciences, Springer, New York, NY, USA, 2009.
- [11] Y. Kuang, Delay Differential Equations with Applications in Population Dynamics, vol. 191 of Mathematics in Science and Engineering, Academic Press, Boston, Mass, USA, 1993.
- [12] R. D. Driver, Ordinary and Delay Differential Equations, Springer, New York, NY, USA, 1977.
- [13] K. Jiaoxun and C. Yuhao, Stability of Numerical Methods for Delay Differentialequations, Science Press, 2005.
- [14] H. J. Tian and J. X. Kuang, "The stability of the θ-methods in the numerical solution of delay differential equations with several delay terms," *Journal of Computational and Applied Mathematics*, vol. 58, no. 2, pp. 171–181, 1995.
- [15] A. N. Al-Mutib, "Stability properties of numerical methods for solving delay differential equations," *Journal of Computational and Applied Mathematics*, vol. 10, no. 1, pp. 71–79, 1984.
- [16] M. Z. Liu and M. N. Spijker, "The stability of the θ-methods in the numerical solution of delay differential equations," *IMA Journal of Numerical Analysis*, vol. 10, no. 1, pp. 31–48, 1990.
- [17] A. Bellen and M. Zennaro, Numerical Methods for Delay Differential Equations, Numerical Mathematics and Scientific Computation, The Clarendon Press Oxford University Press, New York, NY, USA, 2003.
- [18] N. Senu, M. Suleiman, F. Ismail, and M. Othman, "A zero-dissipative Runge-Kutta-Nyström method with minimal phaselag," *Mathematical Problems in Engineering*, Article ID 591341, 15 pages, 2010.
- [19] J. R. Dormand, Numerical Methods for Differential Equations A Computational Approach, Library of Engineering Mathematics, CRC Press, Boca Raton, Fla, USA, 1996.

- [20] J. D. Lambert, *Computational Methods in Ordinary Differential Equations*, John Wiley-Sons, 1972.
- [21] T. M. Morrison and R. H. Rand, "2 : 1 resonance in the delayed nonlinear Mathieu equation," *Nonlinear Dynamics: An International Journal of Nonlinear Dynamics and Chaos in Engineering Systems*, vol. 50, no. 1-2, pp. 341–352, 2007.



The Scientific World Journal





**Decision Sciences** 







Journal of Probability and Statistics



Hindawi Submit your manuscripts at





International Journal of Differential Equations





International Journal of Combinatorics





Mathematical Problems in Engineering



Abstract and Applied Analysis



Discrete Dynamics in Nature and Society







Journal of Function Spaces



International Journal of Stochastic Analysis



Journal of Optimization