A New family of methods for solving delay differential equations

Yogita Mahatekar, Pallavi S. Scindia
Department of Mathematics, College of Engineering Pune, Pune - 411005, India,
yvs.maths@coep.ac.in, pus.maths@coep.ac.in

Abstract

In the present paper, we introduce a new family of $\theta-$methods for solving delay differential equations. New methods are developed using a combination of decomposition technique viz. new iterative method proposed by Daftardar Gejji and Jafari and existing implicit numerical methods. Using Butcher tableau, we observed that new methods are non Runge-Kutta methods. Further, convergence of new methods is investigated along with its stability analysis. Applications to variety of problems indicates that the proposed family of methods is more efficient than existing methods.

Keywords: New iterative method (NIM), Delay differential equation.

1 Introduction

A delay differential equation (DDE) is a differential equation in which state function is given in terms of value of the function at some previous times. Introduction of delay term in modelling allows better representation of real life phenomenon and enriches its dynamics. Due to presence of delay terms in the model, Delay differential equations (DDEs) are infinite dimentional and hence are difficult to analyse. Hence now a days, solving delay differential equations is an important area of research. Every DDE cannot be integrated analytically and hence there is a need to be dependant on numerical methods to solve DDEs. To develop efficient, stable and accurate numerical algorithms is primarily important task of research.

DDEs are receiving increasing importance in many areas of science and engineering like biological processes, population growth and decay models, epidemiology, physiology, neural networks etc [20], [6], [3]. The classical numerical methods such as Eulers method, trapezoidal method, Runge-Kutta methods are discussed in [8]. Enright and Hu [14] have developed a tool to solve DDEs with vanishing delay with iteration and interpolation technique. Karoui and Vaillancourt [19] presneted a SYSDEL code to solve DDEs using Runge kutta methods of desired convergence order. In [21], a new MATLAB program dde23 has been developed to solve wide range of DDEs with constant delays. A new Adomian decomposition method is given in [15] to solve DDEs. Further in [17], two point predictor-corrector block method for solving DDEs is described. Recently New iterative method (NIM) developed by Daftardar-Gejji and Jafari is used to develop many efficient numerical methods to solve fractional differential equations [12], partial differential equations [9], boundary value problems [10]. In [2], a solution of DDE is acheived by using Aboodh transformation method. Recently in [13] , a new numerical technique for solving fractional generalised pantograph-delay differential equations by using fractional order hybrid Bessel functions is developed.

In the present article, we use the powerful technique of NIM to generate new efficient numerical tools to solve DDEs which are reducible to solve ODEs.

The paper is organized as follows. In section 2, important preliminaries like Delay differential equations (DDEs), New iterative method (NIM) etc. are reviewed. In next section 3, we developed a new family of numerical methods to solve DDEs and system of DDEs. In section 4, error analysis of newly proposed methods is done along with its stability analysis in section 5. In section 6, some illustrative examples are solved to check the accuracy of new methods practically. In last section 7, some important observations are made on the basis of theoretical stability analysis and error analysis of newly proposed methods.

2 Preliminaries

2.1 Delay differential equations

Consider a general form of an initial value problem (IVP) representing a time delay differential equation:

$$y' = f(t, y(t), y(t-\tau));\ t \in [0, T], \tag{1}$$

$$y(t) = \phi(t),\ -\tau \leq t \leq 0, \tag{2}$$

where $\phi(t) : [-\tau, 0] \to \mathbb{R}^n;\ n \in \mathbb{N}$ is a real valued function which represents histroy of the solution in the past. Existence and uniqueness conditions for the solution of DDEs are given in [23]. For existence and uniqueness of the solution of the IVP (1-2), we assume that f is non-linear, bounded and continuous real valued operator defined on $[0, T] \times \mathbb{R}^n \times C^1(\mathbb{R}, \mathbb{R}^n)$ to $\mathbb{R}^n$ and fulfils Lipschitz conditions with respect to the second and third arguments:

$$|f(t, x_1, u) - f(t, x_2, u)| \leq L_1|x_1 - x_2|, \tag{3}$$

$$|f(t, x, u_1) - f(t, x, u_2)| \leq L_2|u_1 - u_2|, \tag{4}$$

where L_1 and L_2 are positive constants.

2.2 Approximation to the delay term $y(t - \tau)$

The approximation to the delay term $y(t_n - \tau)$ is denoted by ν_n [12].
When τ is constant $(t_n - \tau)$ may not be a grid point t_n for any n. Suppose $(m + \delta)h = \tau, m \in \mathbb{N}$ and $0 \leq \delta < 1$.
If $\delta = 0$, $mh = \tau$ and $y(t_n - \tau)$ is approximated as

$$y(t_n - \tau) \approx \nu_n = \begin{cases} y_{n-m} & \text{if} \quad n \geq m; \\ \phi(t_n - \tau) & \text{if} \quad n < m. \end{cases}$$

2.3 New iterative method (NIM)

Daftardar-Gejji and Jafari [11] have proposed a new iterative method (NIM) for solving linear/non-linear functional equations of the form

$$u = f + L(u) + N(u) \tag{5}$$

where f is a known function, L is a linear operator and N a non linear operator. This decomposition technique is used by many researchers for solving variety of problems such as fractional differential equations [12], boundary value problems [10], system of non-linear functional equations [5] etc. Recently NIM is used to develop efficient numerical algorithms to solve ordinary differential equations and time delay fractional differential equations too [22],[18].

In this method, We assume that eq.(5) has a series solution of the form $u = \sum_{i=0}^{\infty} u_i$. Since L is a linear operator, we have $L\left(\sum_{i=0}^{\infty} u_i\right) = \sum_{i=0}^{\infty} L(u_i)$ and non linear operator N is decomposed as

$$N(u) = N(u_0) + [N(u_0 + u_1) - N(u_0)] + [N(u_0 + u_1 + u_2) - N(u_0 + u_1)] + \dots$$

Let $G_0 = N(u_0)$ and $G_i = N\left(\sum_{n=0}^{i} u_n\right) - N\left(\sum_{n=0}^{i-1} u_n\right), i = 1, 2, 3, \dots$.

Observe that $N(u) = \sum_{i=0}^{\infty} G_i$.

Putting series solution $u = \sum_{i=0}^{\infty} u_i$ in eq.(5), we get

$$\sum_{i=0}^{\infty} u_i = f + \sum_{i=0}^{\infty} L(u_i) + \sum_{i=0}^{\infty} G_i.$$

Taking $u_0 = f$, and $u_n = L(u_{n-1}) + G_{n-1}, n = 1, 2, 3, \dots$
Note that

$$u = u_0 + u_1 + u_2 + \dots = f + L(u_0) + N(u_0) + L(u_1) + [N(u_0 + u_1) - N(u_0)] + \dots = f + L(u) + N(u).$$

Hence u satisfies the functional eq.(5). k-term NIM solution is given by $u = \sum_{i=0}^{k-1} u_i$.

3 New methods to solve DDEs

In this section, we represent a family of new numerical methods based on NIM and implicit numerical methods to solve delay differential equations. In [22], we developed a new algorithm to solve DDEs and ODEs by improving existing trapezoidal rule of integration. In present paper, we generalize this task by improving general $\theta-$methods to solve delay differential equations as follows.

For solving eq.(1-2) on $[0, T]$, consider the uniform grid $t_n = nh,\ n = -m, -m+1, ..., -1, 0, 1, ...N$, where m and N are integers such that $N = T/h$ and $m = \tau/h$. We integrate eq.(1) from the node t_n to t_{n+1} on both sides, which gives us

$$y(t_{n+1}) = y(t_n) + \int_{t_n}^{t_{n+1}} f(t, y(t), y(t-\tau))\ dt. \tag{6}$$

Now we approximate integration on R.H.S. in above equation as;

$$\int_{t_n}^{t_{n+1}} f(t, y(t), y(t-\tau))dt \approx h\left[(1-\theta)f(t_n, y(t_n), \nu_n) + \theta f(t_{n+1}, y(t_{n+1}), \nu_{n+1})\right], \quad (7)$$

where θ is a parameter which lies in the closed interval $[0, 1]$ and ν_n is approximation to delay term at the node $t = t_n$. Applying the approximation from eq.(7) in eq.(6), we get

$$y_{n+1} = y_n + h\left[(1-\theta)f(t_n, y_n, \nu_n) + \theta f(t_{n+1}, y_{n+1}, \nu_{n+1})\right], \; n = 0, 1, \ldots, N-1. \quad (8)$$

This is called as a family of $\theta-$ methods to solve DDEs.
In particular, when $\theta = 1$, eq.(8) reduces to

$$y_{n+1} = y_n + hf(t_{n+1}, y_{n+1}, \nu_{n+1}), \; n = 0, 1, \ldots, N-1. \quad (9)$$

This eq.(9) is referred as Implicit Euler's method. Note that when $\theta = 0$, eq.(8) yields Explicit Euler's method and when $\theta = \frac{1}{2}$ we get implicit Trapezoidal rule of integration to solve DDEs.

3.1 New Reults

To generate a new family of $\theta-$methids for solving DDEs, we note that eq.(8) is of the form $u = f + N(u)$, and hence NIM can be employed as follows:

We write,

$$\begin{aligned} u &= y_{n+1}, \\ f &= y_n + h(1-\theta)f(t_n, y_n, \nu_n), \\ N(u) &= h\theta f(t_{n+1}, y_{n+1}, \nu_{n+1}). \end{aligned}$$

For simplicity we denote, $f_n = f(t_n, y_n, \nu_n)$. Three term NIM solution of eq.(8) gives us

$$\begin{aligned} u &= u_0 + u_1 + u_2 \\ &= u_0 + N(u_0) + N(u_0 + u_1) - N(u_0) \\ &= u_0 + N(u_0 + u_1) \\ &= u_0 + N(u_0 + N(u_0)). \end{aligned}$$

That is
$y_{n+1} = y_n + h(1-\theta)f_n + N(u_0 + u_1)$
$= y_n + h(1-\theta)f_n + h\theta f(t_{n+1}, u_0 + u_1, \nu_{n+1})$
Therefore,

$$y_{n+1} = y_n + h(1-\theta)f_n + h\theta f(t_{n+1}, y_n + h(1-\theta)f_n + h\theta f(t_{n+1}, y_n + h(1-\theta)f_n, \nu_{n+1}), \nu_{n+1}) \quad (10)$$

Eq.(10) represents new family of $\theta-$methods to solve DDEs which can be expressed in the following more simple form as follows:

$$\left.\begin{aligned} k_1 &= f(t_n, y_n, \nu_n) \\ k_2 &= f(t_{n+1}, y_n + h(1-\theta)k_1, \nu_{n+1}) \\ k_3 &= f(t_{n+1}, y_n + h(1-\theta)k_1 + h\theta k_2, \nu_{n+1}) \\ \text{where, } y_{n+1} &= y_n + h(1-\theta)k_1 + h\theta k_3. \end{aligned}\right\} \quad (11)$$

Case 1: When $\theta = 1$,

$$\left.\begin{aligned} k_1 &= f(t_n, y_n, \nu_n) \\ k_2 &= f(t_{n+1}, y_n, \nu_{n+1}) \\ k_3 &= f(t_{n+1}, y_n + hk_2, \nu_{n+1}) \\ \text{where, } y_{n+1} &= y_n + hk_3. \end{aligned}\right\} \tag{12}$$

Eqs.(12) represents new improved implicit Eulers method to solve DDEs.

Case 2: When $\theta = \frac{1}{2}$,

$$\left.\begin{aligned} k_1 &= f_n = f(t_n, y_n, \nu_n) \\ k_2 &= f(t_{n+1}, y_n + \tfrac{hk_1}{2}, \nu_{n+1}) \\ k_3 &= f(t_{n+1}, y_n + \tfrac{hk_1}{2} + \tfrac{hk_2}{2}, \nu_{n+1}) \\ \text{where, } y_{n+1} &= y_n + \tfrac{hk_1}{2} + \tfrac{hk_3}{2}., \end{aligned}\right\} \tag{13}$$

Eqs.(13) represents improved trapezoidal rule to solve DDEs which is developed in [22].

Case 3: When $\theta = 0$,

$$\left.\begin{aligned} k_1 &= f_n = f(t_n, y_n, \nu_n) \\ k_2 &= f(t_{n+1}, y_n + hk_1, \nu_{n+1}) \\ k_3 &= f(t_{n+1}, y_n + hk_1, \nu_{n+1}) \\ \text{where, } y_{n+1} &= y_n + hk_1., \end{aligned}\right\} \tag{14}$$

Eqs.(14) represents a method which is usual explicit Eulers method to solve DDEs.

Case 4: When $\theta = \frac{3}{4}$,

$$\left.\begin{aligned} k_1 &= f_n = f(t_n, y_n, \nu_n) \\ k_2 &= f(t_{n+1}, y_n + \tfrac{hk_1}{4}, \nu_{n+1}) \\ k_3 &= f(t_{n+1}, y_n + \tfrac{hk_1}{4} + \tfrac{3hk_2}{4}, \nu_{n+1}) \\ \text{where, } y_{n+1} &= y_n + \tfrac{hk_1}{4} + \tfrac{3hk_3}{4}., \end{aligned}\right\} \tag{15}$$

Eqs.(15) is a new method to solve DDEs.

In above family of methods to solve DDEs, when delay term $y(t_n - \tau)$ and its approximation ν_n are zero then DDE (1) reduces to ODE without delay. In accordance with this, above family of methods gets reduced to the family of numerical methods for solving ODEs (without delay) which are developed in [1].

3.2 Non-Runge Kutta methods

General form of Runge-Kutta method is given by

$$y_{n+1} = y_n + h\sum_{i=1}^{3} b_i k_i \text{ and}$$

$$\left.\begin{aligned} k_1 &= f_n = f(t_n, y_n, \nu_n) \\ k_2 &= f(t_n + c_2 h, y_n + ha_{21}k_1, \nu_n) \\ k_3 &= f(t_n + c_3 h, y_n + h(a_{31}k_1 + a_{32}k_2), \nu_n) \end{aligned}\right\} \tag{16}$$

In newly proposed $\theta-$methods represented by eqs.(11), we have $b_1(\theta) = 1-\theta,\ b_2(\theta) = 0,\ b_3(\theta) = \theta,\ a_{21} = 1-\theta,\ a_{31} = 1-\theta,\ a_{32} = \theta,\ c_2 = 1,\ c_3 = 1$. Therefore new Family of $\theta-$methods for solving DDEs can be stated in the form of Butcher tableu as follows.

$$
\begin{array}{c|ccc}
0 & & & \\
1 & 1\text{-}\theta & & \\
1 & 1\text{-}\theta & \theta & \\
\hline
 & 1\text{-}\theta & 0 & \theta
\end{array}
$$

For a Runge Kutta method it is necessary to satisfy that $\sum_{j=1}^{i-1} a_{ij} = c_i \,\forall\, i = 2, 3$ ($cf.$[4]). From the above tableau, for $i = 2$, $\sum_{j=1}^{1} a_{2j} = c_2$ if and only if $\theta = 0$. This shows that, newly proposed family of numerical methods for solving DDEs is different from Runge Kutta methods except for $\theta = 0$.

3.3 New $\theta-$methods for solving a system of delay differential equations

The numerical algorithm presented in above section can be generalized for solving the following system of DDEs:

$$
\begin{aligned}
y_1'(t) &= f_1(t, \overline{y}(t), \overline{y}(t-\tau)),\\
y_2'(t) &= f_2(t, \overline{y}(t), \overline{y}(t-\tau)),\\
&\vdots\\
y_m'(t) &= f_m(t, \overline{y}(t), \overline{y}(t-\tau)),
\end{aligned}
$$

with initial condition

$$\overline{y}(t) = (y_1(t), y_2(t), \ldots, y_m(t)) = (\phi_1(t), \phi_2(t), ..., \phi_m(t)); -\tau \le t \le 0. \tag{17}$$

We let $\overline{y_n}$ be a vector of independent variables representing values of $(y_1, y_2, \ldots, y_m)$ at node $t = t_n$ and $\overline{\nu_n}$ is a vector approximation to $(y_1(t-\tau), y_2(t-\tau), \ldots, y_m(t-\tau))$ at $t = t_n$. We obtain a new family of $\theta-$methods to solve a system of DDEs as follows:
$k_{1,y_i} = f_i(t_n, \overline{y_n}, \overline{\nu_n})$
$k_{2,y_i} = f_i(t_{n+1}, \overline{y_n} + h(1-\theta)k_{1,y_i}, \overline{\nu_{n+1}})$
$k_{3,y_i} = f_i(t_{n+1}, \overline{y_n} + h(1-\theta)k_{1,y_i} + h\theta k_{2,y_i}, \overline{\nu_{n+1}})$
Where,

$$\overline{y_{n+1}} = \overline{y_n} + h(1-\theta)k_{1,y_i} + h\theta k_{3,y_i},\ i = 1, 2, \ldots, m.$$

4 Error analysis

Theorem 4.1 *The new family of $\theta-$methods given by eqs.(11) forms a second order numerical method for $\theta = \frac{1}{2}$ and has a first order convergence for any other value of $\theta \in [0,1]$.*

Proof: Using Taylor's series expansion,

$$y(t_{n+1}) = y(t_n + h) = y_n + hf_n + \frac{h^2}{2}(f_t + ff_y + f_\nu) + O(h^3).$$

Now consider $\theta-$method and Taylors expansions applied in $k_2,\ k_3$.

$$k_1 \quad = \quad f(t_n, y_n, \nu_n) \tag{18}$$

$$\begin{aligned} k_2 &= f(t_{n+1}, y_n + h(1-\theta)k_1, \nu_{n+1}) \\ &= f(t_n + h, y_n + h(1-\theta)k_1, \nu_n + h) \end{aligned}$$

$$\begin{aligned} &k_2 = f_n + (hf_t + h(1-\theta)k_1 f_y + hf_\nu) + \\ &\quad \frac{1}{2}\left(hf_{tt} + h^2(1-\theta)^2 k_1^2 f_{yy} + h^2 f_{\nu\nu} + 2h^2(1-\theta)k_1 f_{ty} + 2h^2 f_{t\nu} + 2h^2(1-\theta)k_1 f_{y\nu}\right) + O(h^3). \end{aligned} \tag{19}$$

$$\begin{aligned} k_3 &= f(t_{n+1}, y_n + h(1-\theta)k_1 + h\theta k_2, \nu_{n+1}) \\ &= f(t_n + h, y_n + h(1-\theta)k_1 + h\theta k_2, \nu_n + h) \end{aligned}$$

$$\begin{aligned} k_3 = f_n &+ (hf_t + h(1-\theta)k_1 f_y + h\theta k_2 f_y + hf_\nu) \\ &+ \frac{1}{2}\big(h^2 f_{tt} + (h^2(1-\theta)^2 k_1^2 + h^2\theta^2 k_2^2 + 2h^2(1-\theta)\theta k_1 k_2) f_{yy} \\ &+ h^2 f_{\nu\nu} + (2h^2(1-\theta)k_1 + 2h^2\theta k_2) f_{yt} + 2h^2 f_{t\nu} + 2(h^2(1-\theta)k_1 + h^2\theta k_2) f_{y\nu}\big) + O(h^3). \end{aligned} \tag{20}$$

Putting (18), (19), (20) in $\theta-$method given by eqs.(11), error e_{n+1} of the method is given by

$$\begin{aligned} e_{n+1} &= y(t_{n+1}) - y_{n+1} \\ &= y_n + hf_n + \frac{h^2}{2}f_t + \frac{h^2 ff_y}{2} + \frac{h^2 f_\nu}{2} + O(h^3) - \big(y_n + h(1-\theta)f_n + h\theta f_n \\ &\quad h^2\theta f_t + h^2\theta(1-\theta)k_1 f_y + h^2\theta^2 k_2 f_y + h^2\theta f_\nu\big) + O(h^3). \end{aligned} \tag{21}$$

Clearly, for $\theta = 0, 1, 3/4$ method has a linear convergence and for $\theta = 1/2$ its order of convegence is quadratic.

4.1 Error analysis of new improved implicit Euler's method for solving DDEs

Refer to Eqs.(12), For $\theta = 1$, we get new improved implicit Eulers method to solve DDEs, which is given by the formula:

$$y_{n+1} = y_n + hf[t_{n+1}, y_n + hf(t_{n+1}, y_n, \nu_{n+1}), \nu_{n+1}]. \tag{22}$$

By inserting analytical solution in the above equation, we get the truncation errorv T_n as

$$\frac{y(t_{n+1}) - y(t_n)}{h} - f[t_{n+1}, y(t_n) + hf(t_{n+1}, y(t_n), \nu_{n+1}), \nu_{n+1}] = T_n. \tag{23}$$

and by eq.(23), we get

$$\frac{y_{n+1} - y_n}{h} - f[t_{n+1}, y_n + hf(t_{n+1}, y_n, \nu_{n+1}), \nu_{n+1}] = 0. \tag{24}$$

Subtracting eq.(23) and eq.(24), we get

$$T_n = \frac{e_{n+1} - e_n}{h} - f[t_{n+1}, y(t_n) + hf(t_{n+1}, y(t_n), \nu_{n+1}), \nu_{n+1}] + f[t_{n+1}, y_n + hf(t_{n+1}, y_n, \nu_{n+1}), \nu_{n+1}]. \tag{25}$$

This implies that,

$$\begin{aligned} hT_n = e_{n+1} - e_n - h\,(&f[t_{n+1}, y(t_n) + hf(t_{n+1}, y(t_n), \nu_{n+1}), \nu_{n+1}] \\ &- f[t_{n+1}, y_n + hf(t_{n+1}, y_n, \nu_{n+1}), \nu_{n+1}]) \,. \end{aligned}$$

Therefore,

$$\begin{aligned} |e_{n+1}| &\leq |e_n| + hL_1|(y(t_n) + hf(t_{n+1}, y(t_n), \nu_{n+1}) - (y_n + hf(t_{n+1}, y_n, \nu_{n+1})| + hT_n \\ &\leq |e_n| + hL_1|e_n| + h^2L_1^2|e_n| + |T_n|h \\ &\leq (1 + hL_1 + h^2L_1^2)|e_n| + |T_n|h;\ \ n = 0, 1, \dots N. \end{aligned}$$

Let $T = \max\limits_{0 \leq n \leq (N-1)} |T_n|$ Therefore,

$$|e_{n+1}| \leq (1 + hL + h^2L^2)|e_n| + Th$$

Now by induction,

$$\begin{aligned} |e_n| &\leq (1 + hL + h^2L^2)^n|e_0| + \left(\frac{(1 + hL + h^2L^2)^{n-1} - 1}{(1 + hL + h^2L^2) - 1}\right) Th \\ &\leq (1 + hL + h^2L^2)^n|e_0| + \left(\frac{e^{nhL} - 1}{(Lh + 1)Lh}\right) Th \\ &\leq e^{nhL}|e_0| + \left(\frac{e^{nhL} - 1}{L}\right) T \\ &\leq e^{(t_n - t_0)L}|e_0| + \left(\frac{e^{(t_n - t_0)L} - 1}{L}\right) T \end{aligned}$$

Since $nh = t_n - t_0$. Noting that, Truncation error $T \le \frac{1}{2}hy''(\zeta)$ and if $y'' \le M_2$ then $T \le \frac{hM_2}{2}$. Therefore,

$$|e_n| \le e^{(t_n-t_0)L_1}|e_0| + \left(\frac{(e^{(t_n-t_0)L_1}-1)hM_2}{2L_1}\right)$$
$$\le \left(\frac{(e^{(t_n-t_0)L_1}-1)hM_2}{2L_1}\right), \text{noting that } |e_0| \text{ is zero.}$$

hence as $h \to 0$, $|e_n| \to 0$. Here L_1 is Lipschitz constant as given in eq.(3). Hence method is convergent. Similary, for $\theta = \frac{1}{2}$, we proved the convergence of the new improved trapezoidal rule for solving DDEs in [22] and for other values of θ convergenrce can be proved on similar lines.

5 Stability Analysis

Definition 5.1 *The numerical method for solving IVP eq.(1-2) is said to be zero-stable if small perturbation in the initial condition of IVP do not cause the numerical approximation to diverge from the exact solution, provided the exact solution of the IVP is bounded.*

Consider the IVP eq.(1-2) and let $\epsilon y(0) = \epsilon y_0$ be the new initial value (perturbed initial condition) obtained by making a small change in $y(0) = y_0$.

Theorem 5.1 *Let y_n be the solution obtained by new improved implicit Euler's method (Case 1 in section3) for solving DDE at the node t_n with the initial condition $y(0) = y_0$ and let ϵy_n be the solution obtained by the same numerical method with perturbed initial condition $\epsilon y_0 = y_0 + \epsilon_0;\ \epsilon_0 > 0$. We assume that $f(t, y)$ satisfies Lipschitz condition with respect to second variable and third variable with Lipschitz constant $L_1,\ L_2$ then $\exists$ positive constants k and ϵ_1 such that $|y_n - \epsilon y_n| \le k\epsilon,\ \forall nh \le T,\ h \in (0, \epsilon_1)$ whenever $|\epsilon_0| \le \epsilon$.*

Proof: We prove the result by induction.

$$|y_n - \epsilon y_n| = |y_{n-1} + hf(t_n, y_{n-1} + hf(t_n, y_{n-1}, \nu_n), \nu_n) - \epsilon y_{n-1} - hf(t_n, \epsilon y_{n-1} + hf(t_n, \epsilon y_{n-1}, \nu_n), \nu_n)|$$
$$\le |y_{n-1} - \epsilon y_{n-1}| + hL_1|y_{n-1} + hf(t_n, y_{n-1}, \nu_n) - \epsilon y_{n-1} - hf(t_n, \epsilon y_{n-1}, \nu_n)|$$
$$\le |y_{n-1} - \epsilon y_{n-1}| + hL_1|y_{n-1} - \epsilon y_{n-1}| + h^2L_1^2|y_{n-1} - \epsilon y_{n-1}|$$
$$= (1 + hL_1 + h^2L_1^2)|y_{n-1} - \epsilon y_{n-1}|$$

Therefore by induction,

$$|y_n - \epsilon y_n| \le (1 + hL_1 + h^2L_1^2)^n|y_0 - \epsilon y_0|$$
$$\le e^{nhL_1}|\epsilon_0|$$
$$= e^{T_M L_1}\epsilon = k\epsilon, \text{ where } k = e^{T_M L_1} > 0.$$

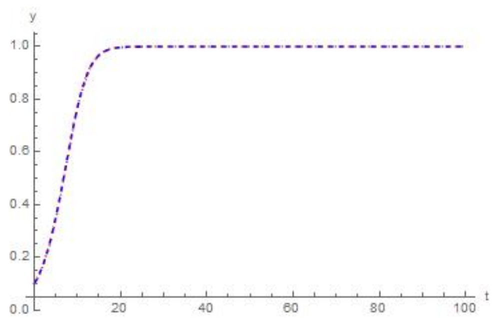
This proves that the new method for $(\theta = 1)$ is stable.

6 Illustrative examples

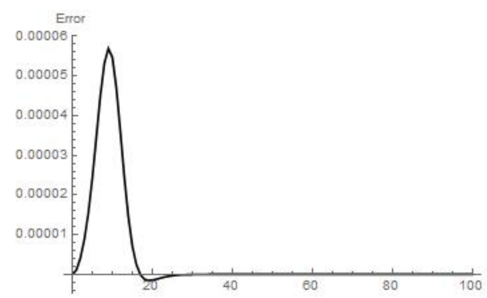
To demonstrate applicability of newly proposed methods, We present here some illustrative examples which are solved using Mathematica 12.

Example 6.1 *Consider the delay logistic differential equation*

$$y'(t) \quad = \quad 0.3y(t)(1 - y(t-1)),\ y(t \leq 0) = 0.1. \tag{26}$$



(a) Solution of (26) by new method ($\theta = 1$)



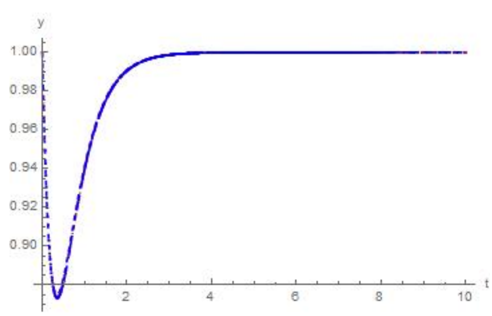
(b) Error in solution of (26) when solved using new method ($\theta = 1$)

Figure 1: Dashed graph: Solution by new method ($\theta = 1$), Dotted graph: Exact solution. It is noted that in Fig.1 (a) exact solution and approximate solution by the new method ($\theta = 1$) overlaps on each other.
Step length is taken as $h = 0.01$.

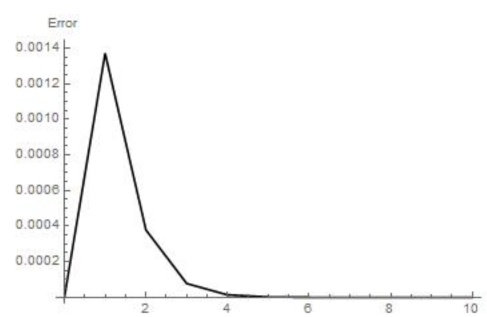
Example 6.2 *Consider the differential equation without delay [7]*

$$y'(t) \quad = \quad 2 - e^{-4t} - 2y,\ y(0) = 1,\ 0 \leq t \leq 10. \tag{27}$$

Exact solution of the differential equation (27) is $y(t) = 1 + \dfrac{e^{-4t} - e^{-2t}}{2}$.



(a) Solution of (27) by new method ($\theta = 1$)



(b) Error in solution of (27) when solved using θ method ($\theta = 1$)

Figure 2: Dashed graph: Solution by new method, Dotted graph: Exact solution. It is noted that in Fig.2 (a) exact solution and approximate solution by the new method overlaps on each other.
Step length is taken as $h = 0.01$.

The error in the new method is shown in Fig.2 (b). Thus, the new method is accurate. In Table 1, error in implicit backward Euler's method E_1 and error in new method ($\theta = 1$) E_2 (obtained by taking 3-term NIM solution) while solving (27) are compared for various values of t. It is noteworthy that E_2 is always smaller than E_1 in all cases. So the new method is more accurate than implicit backward Euler's method.

t_n	S_1	S_2	S_3	S	e_1	e_2	e_3
0	1	1	1	1	0	0	0
0.01	0.99016	0.9902	0.990196	0.990295	0.000099	0.0000954	0.0000994
0.02	0.980969	0.980976	0.980969	0.981163	0.000194	0.000187	0.0001948
0.03	0.972292	0.97230	0.972292	0.972578	0.000286	0.000275	0.0002863
0.04	0.96414	0.9641	0.96414	0.964514	0.0003739	0.00036	0.0003741
0.05	0.95648	0.9565	0.956488	0.956947	0.000458	0.0004413	0.0004584
0.06	0.949315	0.949334	0.949315	0.949854	0.000538	0.0005194	0.0005391
0.1	0.924966	0.924993	0.924966	0.925795	0.0008283	0.00080	0.0008289
0.2	0.980969	0.980976	0.980969	0.981163	0.000194	0.000187	0.0001948
0.3	0.8745	0.874563	0.874528	0.87619	0.0016622	0.001628	0.001663
0.4	0.87447	0.874498	0.87447	0.87628	0.001813	0.001785	0.001814
0.5	0.881873	0.881893	0.881872	0.883728	0.001855	0.001834	0.0018554

Table 1: Ex.(8.1)
S_1 : Solution by backward Euler's method
S_2 : Solution by new method ($\theta = 1$) obtained by taking 3-term NIM solution
S_3 : Solution by new method ($\theta = 1$) obtained by taking 4-term NIM solution
S : Exact solution
e_1 : Error in solution by backward Euler's method
e_2 : Error in solution by new method ($\theta = 1$) obtained by taking 3-term NIM solution
e_3 : Error in solution by new method ($\theta = 1$) obtained by taking 4-term NIM solution

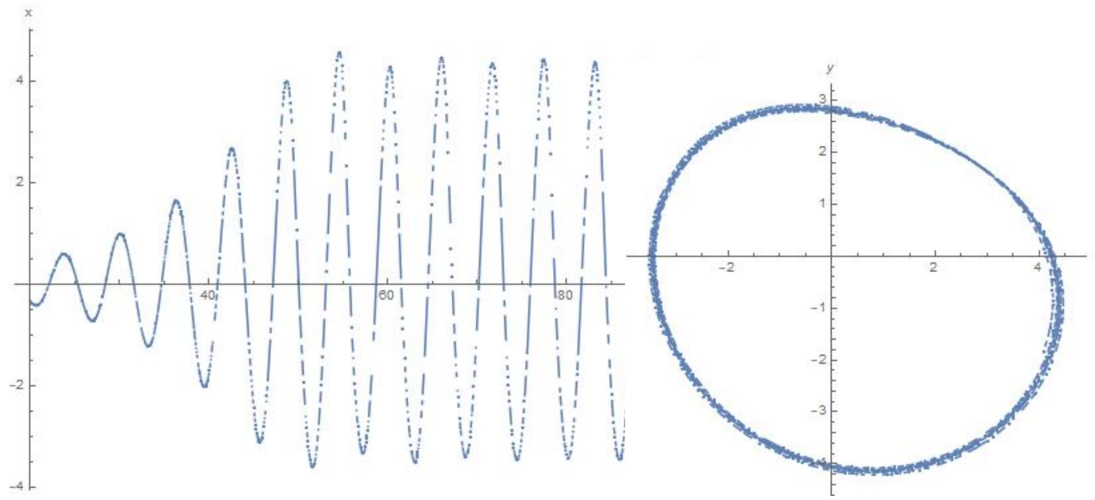
Observation: In this example, it is observed that (4-term NIM solution) new method ($\theta = 1$) method and backward Euler's method gives same error and error in these two methods is greater than (3-term NIM solution) new method ($\theta = 1$). Hence, new method with three term NIM solution gives better accuracy than implicit backward euler method and new method with 4-term NIM solution.

6.1 R$\ddot{o}$ssler System with delay

Consider the R$\ddot{o}ssler$ system [16] with delay given by the following system of differential equations.

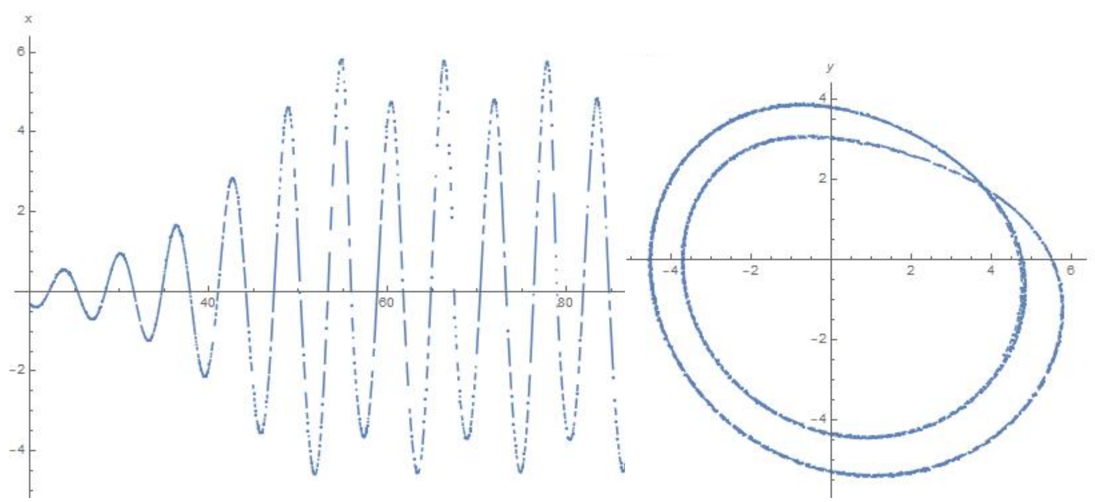
$$\begin{aligned} \dot{x} &= -y(t) - z(t), \\ \dot{y} &= x(t) + ay(t-1), \\ \dot{z} &= b + z(t)(x(t) - c). \end{aligned} \tag{28}$$

Let $a = b = 0.2,\ x(0) = y(0) = z(0) = 0.0001$. Note that c is a control parameter. Solving above system by new method for ODE we obtain the x-waveforms and x-y phase potraits which are depicted in Figs.3(a)-(b), Figs.4(a)-(b) and in Figs.5(a)-(b) for $c = 2.3$, $c = 2.9$, $c = 7.9$ respectively.



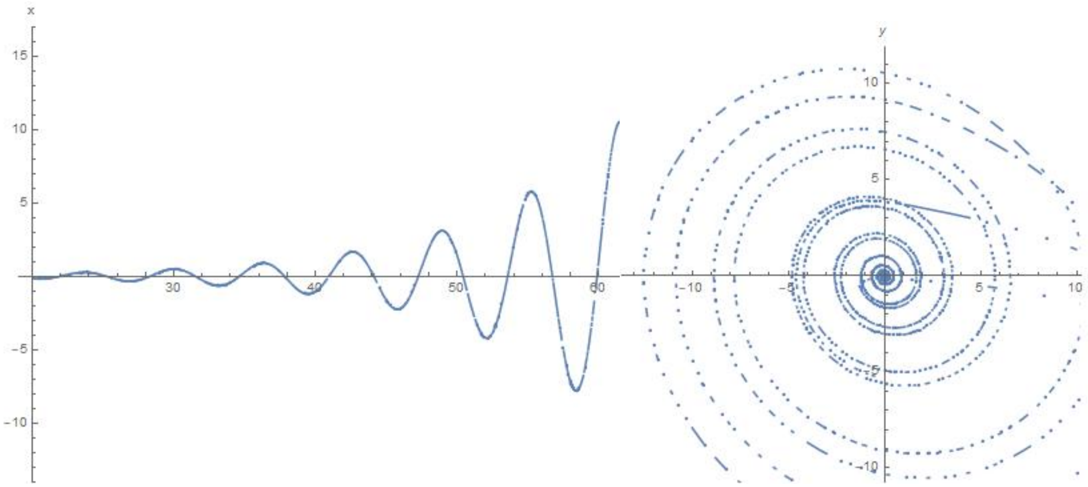
(a) x-waveform of the system (28)　　(b) xy-phase diagram of the system (28)

Figure 3: c=2.3 in system(28)



(a) x-waveform of the system (28)　　(b) xy-phase diagram of the system (28)

Figure 4: c=2.9 in system(28)



(a) x-waveform of the system (28) (b) xy-phase diagram of the system (28)

Figure 5: c=7.9 in system(28)

7 Conclusions

In the present work, a new family of methods for solving delay differential equations (DDEs) has been proposed which are reducible to solve ordinary differential equations too (without delay). Newly proposed methods are then compared with existing methods with respect to stability and accuracy. New methods formed found to be stable and accurate. Further error analysis and stability analysis of new methods is carried out. Numerous illustrative examples are solved using Mathematica 12 to demonstrate the efficiency of the method. It is observed that, new methods are non-Runge Kutta methods and are more accurate than existing numerical methods for solving DDEs.

References

[1] O. Y. Ababneh. New numerical methods for solving differential equations, 2019.

[2] K. Aboodh, R. Farah, I. Almardy, and A. Osman. Solving delay differential equations by aboodh transformation method. *International Journal of Applied Mathematics & Statistical Sciences*, 7(2):55–64, 2018.

[3] C. Baker, G. Bocharov, C. Paul, and F. Rihan. Modelling and analysis of time-lags in some basic patterns of cell proliferation. *Journal of mathematical biology*, 37(4):341–371, 1998.

[4] A. Bellen and M. Zennaro. *Numerical methods for delay differential equations*. Oxford University Press, 2013.

[5] S. Bhalekar and V. Daftardar-Gejji. Solving a system of nonlinear functional equations using revised new iterative method. *International Journal of Mathematical and Computational Sciences*, 6(8):968–972, 2012.

[6] G. A. Bocharov and F. A. Rihan. Numerical modelling in biosciences using delay differential equations. *Journal of Computational and Applied Mathematics*, 125(1-2):183–199, 2000.

[7] T. Bui. Explicit and implicit methods in solving differential equations. 2010.

[8] J. C. Butcher. Numerical methods for ordinary differential equations in the 20th century. *Journal of Computational and Applied Mathematics*, 125(1-2):1–29, 2000.

[9] V. Daftardar-Gejji and S. Bhalekar. Solving fractional diffusion-wave equations using a new iterative method. *Fractional Calculus and Applied Analysis*, 11(2):193p–202p, 2008.

[10] V. Daftardar-Gejji and S. Bhalekar. Solving fractional boundary value problems with dirichlet boundary conditions using a new iterative method. *Computers & Mathematics with Applications*, 59(5):1801–1809, 2010.

[11] V. Daftardar-Gejji and H. Jafari. An iterative method for solving nonlinear functional equations. *Journal of Mathematical Analysis and Applications*, 316(2):753–763, 2006.

[12] V. Daftardar-Gejji, Y. Sukale, and S. Bhalekar. Solving fractional delay differential equations: A new approach. *Fractional Calculus and Applied Analysis*, 18(2):400–418, 2015.

[13] H. Dehestani, Y. Ordokhani, and M. Razzaghi. Numerical technique for solving fractional generalized pantograph-delay differential equations by using fractional-order hybrid bessel functions. *International Journal of Applied and Computational Mathematics*, 6(1):1–27, 2020.

[14] W. H. Enright and M. Hu. Interpolating runge-kutta methods for vanishing delay differential equations. *Computing*, 55(3):223–236, 1995.

[15] D. J. Evans and K. Raslan. The adomian decomposition method for solving delay differential equation. *International Journal of Computer Mathematics*, 82(1):49–54, 2005.

[16] K. Ibrahim, R. Jamal, and F. Ali. Chaotic behaviour of the rossler model and its analysis by using bifurcations of limit cycles and chaotic attractors. In *J. Phys. Conf. Ser*, volume 1003, page 012099, 2018.

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

[18] A. Jhinga and V. Daftardar-Gejji. A new numerical method for solving fractional delay differential equations. *Computational and Applied Mathematics*, 38(4):1–18, 2019.

[19] A. Karoui and R. Vaillancourt. A numerical method for vanishing-lag delay differential equations. *Applied Numerical Mathematics*, 17(4):383–395, 1995.

[20] F. A. Rihan, D. Abdelrahman, F. Al-Maskari, F. Ibrahim, and M. A. Abdeen. Delay differential model for tumour-immune response with chemoimmunotherapy and optimal control. *Computational and mathematical methods in medicine*, 2014, 2014.

[21] L. F. Shampine, S. Thompson, and J. Kierzenka. Solving delay differential equations with dde23. *URL http://www. runet. edu/˜ thompson/webddes/tutorial. pdf*, 2000.

[22] Y. Sukale and V. Daftardar-Gejji. New numerical methods for solving differential equations. *International Journal of Applied and Computational Mathematics*, 3(3):1639–1660, 2017.

[23] E. Süli and D. F. Mayers. *An introduction to numerical analysis*. Cambridge university press, 2003.