Chebyshev operational matrix for solving fractional order delay-differential equations using spectral collocation method

Khalid K. Ali, Mohammad A. Abd El Salam and Emad M. Mohamed
Department of Mathematics, Faculty of Science, Alazhar University, Cairo, Egypt

ABSTRACT
In this article, the solution for general form of fractional order delay-differential equations (GFDEs) is introduced. The proposed GFDEs have multi-term of integer and fractional order derivatives for delayed or non delayed terms. An operational matrix is presented for all terms. The spectral collocation method is used to solve the proposed GFDEs as a matrix discretization method. The reliability and efficiency of the proposed scheme are demonstrated by some numerical experiments.

1. Introduction
Spectral methods are one of the principal methods of matrix discretization for the solutions of differential equations. The main feature of these methods lies in their accuracy for a given number of unknown (see, for instance Doha & Bhrawy, 2008; Doha, Bhrawy, & Saker, 2011; Ramos, Hussaini, Quarteroni, & Zang, 1988).

Many works considered the ordinary (ODEs) and partial (PDEs) differential equations by using different spectral methods (Khader & Saad, 2018; Parand & Razzaghi, 2004; Siyyam, 2001; Wang & Guo, 2012), and the linear and non-linear systems of differential equations (Akyüz & Sezer, 2003; Ramadan, 2016). In addition, many articles considered the integral equations, delay-differential (DDEs) equations and mixed form of differential integral difference equations with different basis functions using spectral methods (Gülsu, Öztürk, & Sezer, 2010; Yang, 2012).

In the recent decades, the appearance of fractional calculus and fractional differential equations (FDEs) opens many research points. The scientists of applied mathematics and the engineers realize that FDEs provide a better approach to describe the complex phenomena in nature, such as non-Brownian motion, signal processing, systems identification, control viscoelastic materials and polymers (see Kilbas, Srivastava, & Trujillo, 2006; Machado, Kiryakova, & Mainardi, 2011; Podlubny, 1999 and references therein).

Numerical solution of FDEs using spectral methods grow by the last ten years, the Legendre wavelet method is developed and used for solving FDEs in ur Rehman and Khan (2011). Moreover, the authors in Doha, Bhrawy, and Ezz-Eldien, (2011b), Esmaeili and Shamsi (2011), and Saadatmandi and Dehghan (2010) constructed an efficient spectral method for the numerical approximation of the multi-term FDEs based on tau and pseudo-spectral methods. Furthermore, Bhrawy, Alofi, & Ezz-Eldien (2011; Doha, Bhrawy, & Ezz-Eldien, 2011a) introduced a quadrature shifted Legendre tau method based on Gauss-Lobatto interpolation for solving the multi-order FDEs with variable coefficients. The operational matrix of fractional derivatives has been determined for some types of orthogonal basis functions, such as Chebyshev polynomials (Saadatmandi & Dehghan, 2010). Also, the operational matrix of integration has been determined for several types of orthogonal polynomials, such as Laguerre series (Hwang & Shih, 1982), Chebyshev polynomials (Paraskevopoulos, 1983), Legendre polynomials (Paraskevopoulos, 1985) and Bessel series (Paraskevopoulos, Sklavounos, & Georgiou, 1990). Recently, Singh, Singh, and Singh (2009) derived the Bernstein operational matrix of integration.

Few years ago, many authors presented numerical treatments for a kind of FDEs contain a delay term, called Fractional-Delay-differential equations (FDDEs). In Morgado, Ford, and Lima (2013) and Xu and Lin...
(2016), the existence and uniqueness of its solution presented. Legendre pseudo-spectral method (Khader & Hendy, 2012), Bernoulli wavelets (Rahimkhani, Ordokhani, & Babolian, 2017), Hermite wavelet method (Saeed & ur Rehman, 2014), collocation method with shifted Jacobi polynomials (Muthukumar & Ganesh Priya, 2017) and collocation method with Chebyshev polynomials (Muthukumar & Ganesh Priya, 2017) were presented as a spectral numerical solution for FDEs. All of the previous work considered as a single fractional term in the right hand side and a delay term in the left hand side free of integer or fractional order derivatives.

In this article, an operational matrix of fractional derivatives is presented for Chebyshev polynomials, and employs it to deal with a general form of GFDDEs. The proposed model of GFDDEs is chosen to be multi-term of fractional derivatives and also integer order derivatives, where the delayed terms are taken to be multi-term of fractional and integer order derivatives too. The presented operational matrix is used with the collocation method to solve the proposed GFDDEs as a matrix discretization method. The high accuracy of this method is verified through some numerical examples. The obtained numerical results are compared with other methods, where they show the proposed method gives good accuracy.

The article is organized as: Preliminary and notations for some of properties for Chebyshev polynomials and fractional calculus in Section 2. The introduced operational matrix of fractional derivatives is in Section 3, where the proposed model of GFDDEs and the fundamental matrix relations are in Section 4. In Section 5 the method of solution is presented. Section 6 contains the numerical results and the comparisons.

2. Preliminaries and notations

In this section, we present some definitions and some properties for the fractional derivative and the Chebyshev polynomial of the first kind.

2.1. The Caputo fractional derivative

Definition 1. The Caputo fractional derivative operator $D^\nu$ of order $\nu$ is defined in the following form:

$$ D^\nu f(x) = \frac{1}{\Gamma(m-\nu)} \int_0^x \frac{f^{(m)}(t)}{(x-t)^{m+\nu}} dt, \quad \nu > 0, \quad m-1 < \nu \leq m, \quad m \in \mathbb{N}, \quad x > 0. \quad (1) $$

where $m-1 < \nu \leq m, m \in \mathbb{N}, x > 0$.

**Properties 1.**

- $D^\nu (\lambda f(x) + \mu g(x)) = \lambda D^\nu f(x) + \mu D^\nu g(x)$, where $\lambda$ and $\mu$ are constants.
- $D^\nu C = 0$, where $C$ is a constant,
- $D^\nu x^\nu = \begin{cases} 0, & \text{for } n \in \mathbb{N} \text{ and } n < [\nu] \\ \frac{1}{\Gamma(n+1)} \frac{x^{n-\nu}}{(n-\nu)!}, & \text{for } n \in \mathbb{N} \text{ and } n \geq [\nu] \end{cases}$.

where $[\nu]$ denote to the smallest integer greater than or equal to $\nu$, and $\mathbb{N} = \{0, 1, 2, \ldots\}$.

2.2. Chebyshev polynomials of the first-kind

The Chebyshev polynomials $T_n(x)$ of the first-kind are orthogonal polynomials of degree $n$ in $x$ defined on $[-1, 1]$

$$ T_n = \cos n\theta, $$

where $x = \cos \theta$ and $\theta \in [0, \pi]$. The polynomials $T_n(x)$ be generated by using the recurrence relations

$$ T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x), \quad T_0(x) = 1, \quad T_1(x) = x, \quad n = 1, 2, \ldots. $$

The Chebyshev polynomials $T_n(x)$ can be expressed in terms of the power $x^l$ in different forms found in Ramadan, Raslan, El Danaf, and El Salam (2017), Bhrawy and Alofi (2013), one of them is

$$ T_n(x) = \sum_{k=0}^{\lfloor n/2 \rfloor} c_k^{(n)} x^{n-2k}, \quad (2) $$

where

$$ c_k^{(n)} = (-1)^k 2^{n-2k-1} \frac{n}{n-k} \binom{n-k}{k}, \quad 2k \leq n. $$

From Eq. (2) we can define that:

- If $n$ is even we find

$$ T_n(x) = T_{2l}(x) = \sum_{j=0}^{l} (-1)^{l-j} 2^{2l-1} \frac{2l}{l+j} \binom{l+j}{l-j} x^{2l}. $$

$$ (3) $$

- If $n$ is odd we can write

$$ T_n(x) = T_{2l+1}(x) = \sum_{j=0}^{l} (-1)^{l-j} 2^{2l+1} \frac{2l+1}{l+j+1} \binom{l+j+1}{l-j} x^{2l+1}. $$

$$ (4) $$

From Eqs. (3) and (4) we can write $T(x)$ as general matrix form as (Ramadan et al., 2017)

$$ T(x) = X(x)M, $$

where $T(x)$ and $X(x)$ are matrices have the form:

$$ T(x) = \begin{bmatrix} T_0(x) & T_1(x) & \ldots & T_N(x) \end{bmatrix}, $$

$$ X(x) = \begin{bmatrix} x^0 & x^1 & \ldots & x^N \end{bmatrix}, $$

and $M$ is $(N+1) \times (N+1)$ matrix given by
In this case, we are going to use the last row for odd values of \( N = 2l + 1 \), otherwise previous one will be the last row of matrix \( M \) \((N = 2l)\). Now, from Eq. (5) we can obtain the \( k \)th derivative of the matrix \( T(x) \) as:

\[
T^{(k)}(x) = X^{(k)}(x)M^T, \quad k = 0, 1, 2, \ldots
\] (6)

### 3. Operational matrices

In this section, we introduce the generalize operational matrices for \( T(x) \), \( T^{(k)}(x) \), \( T(x-\tau) \), \( T^{(s)}(x-\tau) \), \( D^\alpha T(x) \) and \( D^\alpha T(x-\tau) \) according to fractional calculus.

\[
B_{-\tau} = \begin{bmatrix}
0 & 1 & 0 & \ldots & 0 \\
0 & 0 & 2 & \ldots & 0 \\
0 & 0 & 0 & \ldots & N \\
0 & 0 & 0 & \ldots & 0
\end{bmatrix}
\]

The row vector \( T(x-\tau) \) represents in terms of the vector \( X(x) \) in the following form (Gülsu et al., 2010):

\[
T(x-\tau) = X(x)B_{-\tau}M^T,
\]

where

\[
B_{-\tau} = \begin{bmatrix}
0 & 1 & 0 & \ldots & 0 \\
0 & 0 & 2 & \ldots & 0 \\
0 & 0 & 0 & \ldots & N \\
0 & 0 & 0 & \ldots & 0
\end{bmatrix}
\]

**Lemma 3.1.** The \( (k) \)th order derivative of the row vector \( T(x) \) can be in the following relation form (Gülsu et al., 2010):

\[
T^{(k)}(x) = X^{(k)}(x)B^kM^T,
\] (7)

where the matrix \( B \)

**Proof.** see Gülsu et al. (2010)

**Corollary 3.1.** The \( (s) \)th order derivative of the delay row vector \( T(x-\tau) \) can be represented as (Gülsu et al., 2010):
\[ T^{(t)}(x-t) = X^{(t)}(x-t)M^T = X(x)B_\nu M^T. \] (10)

According to the previous lemmas with the fractional calculus by using the Caputo fractional derivative we introduce the following theorem.

**Theorem 3.1.** The \( \nu \)th order fractional derivative of the vector \( T(x) \) can be written as

\[ D^\nu T(x) = X_\nu(x)B_\nu M^T, \] (11)

where

\[ X_\nu(x) = [0, 0, \ldots, 0, x^{\nu-1}, \ldots, x^{N-\nu}], \] (12)

if \( n-1 < \nu_i < n, \quad n \in N \).

\[ B_\nu = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix} + \frac{\Gamma(n+1)}{\Gamma(n+1-\nu_i)} \begin{bmatrix} \Gamma(n+1-\nu_i) & 0 & \cdots & 0 \\ \Gamma(n+1-\nu_i) & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \Gamma(n+1-\nu_i) & 0 & \cdots & 0 \end{bmatrix}, \] (13)

and if \( 0 < \nu_i < 1 \), then

\[ X_\nu(x) = [0, x^{1-\nu_i}, x^{2-\nu_i}, \ldots, x^{N-\nu_i}], \] (14)

\[ B_\nu = \begin{bmatrix} 0 & \Gamma(2) & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix} + \frac{\Gamma(3)}{\Gamma(3-\nu_i)} \begin{bmatrix} \Gamma(3-\nu_i) & 0 & \cdots & 0 \\ \Gamma(3-\nu_i) & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \Gamma(3-\nu_i) & 0 & \cdots & 0 \end{bmatrix}. \] (15)

**Proof.**

\[ D^\nu T(x) = D^\nu X(x)M^T = D^\nu \left[ 1, x, x^2, \ldots, x^N \right] M^T, \quad n-1 < \nu_i < n, n \in N \]

\[ = \left[ 0, \ldots, 0, \frac{\Gamma(n+1)}{\Gamma(n+1-\nu_i)} x^{\nu_i}, \ldots, \frac{\Gamma(n+1)}{\Gamma(n+1-\nu_i)} x^{N-\nu_i} \right] M^T \]

\[ = X_\nu(x)B_\nu M^T. \] (16)

\[ \square \]

**Corollary 3.2.** The \( \xi \)th order fractional derivative of the vector \( T(x-t) \) can be written in following form:

\[ D^\xi T(x-t) = X_\xi(x)B_\xi M^T. \] (17)

**Proof.** By using Eq. (11) and by putting \( x \rightarrow (x-t) \) we get

\[ D^\xi T(x-t) = D^\xi X(x-t)M^T = D^\xi X(x)B_\xi M^T = X_\xi(x)B_\xi M^T, \] (18)

where

\[ X_\xi(x) = [0, 0, \ldots, 0, x^{\xi-1}, \ldots, x^{N-\xi}], \]

if \( n-1 < \xi_i < n, \quad n \in N \).

\[ B_\xi = \begin{bmatrix} 0 & 0 & \cdots & 0 \end{bmatrix} + \frac{\Gamma(n+1)}{\Gamma(n+1-\xi_i)} \begin{bmatrix} \Gamma(n+1-\xi_i) & 0 & \cdots & 0 \end{bmatrix}, \] (19)

and if \( 0 < \xi_i < 1 \), then

\[ X_\xi(x) = [0, x^{1-\xi_i}, x^{2-\xi_i}, \ldots, x^{N-\xi_i}], \] (20)

\[ B_\xi = \begin{bmatrix} 0 & \Gamma(2) & \cdots & 0 \end{bmatrix} + \frac{\Gamma(3)}{\Gamma(3-\xi_i)} \begin{bmatrix} \Gamma(3-\xi_i) & 0 & \cdots & 0 \end{bmatrix}. \]

**4. Application to fractional order delay-differential equation**

The general form of the linear fractional order Delay-differential equation is presented in this section. In addition, the general operational matrices for all terms will be obtained.

Now, we consider the general fractional order Delay-differential equation:

\[ \sum_{i=0}^{n_1} Q_i(x) D^\nu y(x) + \sum_{j=0}^{n_2} P_j(x) D^\nu y(x-t) \]

\[ + \sum_{k=0}^{n_3} Q_k^s(x) y^{(s)}(x) + \sum_{i=0}^{n_4} P_i^s(x) y^{(i)}(x-t) = g(x), \] (21)

under the conditions

\[ y^{(i)}(0) = \mu_i, \quad i = 0, 1, 2, \ldots, m-1, \] (22)

where \( x \in [0, \ldots, 0 \text{ and } Q_i(x), Q_k^s(x), P_j(x), P_i^s(x) \) are known functions, \( i = 0, 1, \ldots, n_1; j = 0, 1, \ldots, n_2; k = 0, 1, \ldots, n_3 \) and \( s = 0, 1, \ldots, n_4 \). Also \( m \) is the greatest integer order derivative exist, or the highest integer order greater than the fractional derivative. Let us write Eq. (21) in the form:

\[ D(x) + F(x) + L(x) + H(x) = g(x), \] (23)

where

\[ D(x) = \sum_{i=0}^{n_1} Q_i(x) D^\nu y(x), \quad F(x) = \sum_{j=0}^{n_2} P_j(x) D^\nu y(x-t), \]

\[ L(x) = \sum_{k=0}^{n_3} Q_k^s(x) y^{(s)}(x), \quad H(x) = \sum_{i=0}^{n_4} P_i^s(x) y^{(i)}(x-t). \] (24)
Therefore, we consider the approximate solution in the form:

\[
y(x) \approx y_N(x) = \sum_{i=0}^{N} a_i T_i(x),
\]

(25)

where \(a_i\) are unknown Chebyshev coefficients and \(N\) is chosen any positive integer such that \(N \geq m\).

From Eq. (25) we get,

\[
y_N(x) = T(x)A,
\]

(26)

\[
y_N^{(k)}(x) = T^{(k)}(x)A,
\]

(27)

\[
D^s y_N(x) = T^{(s)}(x)A,
\]

(28)

and

\[
D^s y_N(x-\tau) = T^{(s)}(x-\tau)A, \quad i = 0, 1, \ldots, N,
\]

(29)

where

\[
A = \left[ \frac{1}{2} a_0, a_1, a_2, \ldots, a_N \right]^T.
\]

By putting \(x \rightarrow x-\tau\) in the relation Eq. (27) we obtain the matrix form

\[
y_N^{(s)}(x-\tau) = T^{(s)}(x-\tau)A. \quad \ldots \quad (30)
\]

Consequently, by substituting the matrix form Eq. (7) into (27) we have the matrix relation

\[
y_N^{(k)}(x) = X^{(k)}(x)B^sM^t A.
\]

(31)

By substituting the matrix forms Eq. (11) into Eq. (28) we have the matrix relation

\[
D^s y_N(x) = X_{n}(x)B_p M^t A.
\]

(32)

By using Eqs. (17) and (29) we get

\[
D^s y_N(x-\tau) = X_{n}(x)B_p B_{-n} M^t A.
\]

(33)

Similar form Eqs. (10) and (30) we obtain

\[
y_N^{(s)}(x-\tau) = X(x)B^sB_{-n} M^t A. \quad \ldots \quad (34)
\]

To obtain the solution Eq. (25) for Eqs. (21) and (22), we can use the collocation points defined in this form:

\[
x_i = \frac{-\tau}{2} \left( 1 + \cos \left( \frac{i\pi}{N} \right) \right), \quad i = 0, 1, 2, \ldots, N.
\]

(35)

By using Chebyshev collocation method with Eq. (35), we obtain the system Eq. (23) as:

\[
D(x_i) + F(x_i) + L(x_i) + H(x_i) = g(x_i),
\]

(36)

where

\[
D(x) = \sum_{i=0}^{n_1} Q_i(x) D^s y_N(x_i), F(x) = \sum_{i=0}^{n_1} P_i(x) D^s y_N(x_i-\tau),
\]

\[
L(x) = \sum_{i=0}^{n_1} Q_i(x) y_N^{(s)}(x_i), H(x) = \sum_{i=0}^{n_1} P_i(x) y_N^{(s)}(x_i-\tau).
\]

Then, we can write the system Eq. (36) in the form

\[
D + F + L + H = G.
\]

(37)

4.1. Matrix representation of the fractional differential part

Now, we can compute the matrix representation for all terms in Eq. (37) the first terms in Eq. (37) can be write as:

\[
D = \sum_{i=0}^{n_1} Q_i, D^sY, \quad (38)
\]

where

\[
\begin{bmatrix}
D(x_0) \\
D(x_1) \\
D(x_2) \\
\vdots \\
D(x_{n_1})
\end{bmatrix}, \quad F = \begin{bmatrix}
F(x_0) \\
F(x_1) \\
F(x_2) \\
\vdots \\
F(x_{n_1})
\end{bmatrix}, \quad L = \begin{bmatrix}
L(x_0) \\
L(x_1) \\
L(x_2) \\
\vdots \\
L(x_{n_1})
\end{bmatrix}
\]

\[
H = \begin{bmatrix}
H(x_0) \\
H(x_1) \\
H(x_2) \\
\vdots \\
H(x_{n_1})
\end{bmatrix}, \quad G = \begin{bmatrix}
g(x_0) \\
g(x_1) \\
g(x_2) \\
\vdots \\
g(x_{n_1})
\end{bmatrix}
\]

\[
Y = \begin{bmatrix}
Y^{(i)}(x_0) \\
Y^{(i)}(x_1) \\
Y^{(i)}(x_2) \\
\vdots \\
Y^{(i)}(x_{n_1})
\end{bmatrix}
\]

4.2. Matrix representation of fractional differential-delay part

We can obtain the second term \(F(x)\) in Eq. (37) by using Eqs. (33) and (35) in this form:

\[
F(x) = \sum_{i=0}^{n_2} P_j(x) X_{n_1} B_p B_{-n} M^t A.
\]

(40)

4.3. Matrix representation of differential part

From Eqs. (31) and (35) the third term \(L(x)\) in Eq. (37) can be write as:

\[
D(x) = \sum_{i=0}^{n_1} Q_i(x) X_{n_1} B_p M^t A.
\]

(39)
\[ L(x) = \sum_{k=0}^{n_1} Q_k^2(x) X(x) B^{(k)} M^T A. \] (41)

### 4.4. Matrix representation of differential-delay part

By using Eqs. (35) and (34) we can write \( H(x) \) in Eq. (37) as:

\[ H(x) = \sum_{i=0}^{n_4} P_i^* x) X(x) B^i B M^T. \] (42)

### 4.5. Matrix relation for the conditions

Finally, we can obtain the matrix form for the conditions Eq. (22) by using Eq. (26) on the form:

\[ X(0) B M^T A = \mu_i, \quad i = 0, 1, 2, \ldots, m-1, \] (43)

or

\[ U_i A = [\mu_i], \] (44)

where

\[ U_i = X(0) B M^T = [u_0, u_1, \ldots, u_N]. \]

### 5. Method of solution

Now, we ready to construct the fundamental matrix equation corresponding to Eq. (21) for this purpose, substitute Eqs. (39), (40), (41) and (42), into (21) and then simplify. Thus we have the fundamental matrix equation as:

\[
\begin{bmatrix}
\sum_{i=0}^{n_4} Q_i(x) X_i B_i M^T + \sum_{j=0}^{n_0} P_j(x) X_0 B_0 B M^T \\
+ \sum_{k=0}^{n_1} Q_k^2(x) X(k) B^{(k)} M^T + \sum_{s=0}^{n_4} P_s^* x) X(x) B^s B M^T
\end{bmatrix}
A = G. \] (45)

Relation Eq. (45) represents a system of \((N+1)\) algebraic equations for the \((N+1)\) unknown Chebyshev coefficients \(a_0, a_1, \ldots, a_N\). In short Eq. (45) can be written in the form:

\[ WA = G, \quad \text{or} \quad [W; G], \] (46)

where

\[ W = [W_{pq}] = \begin{bmatrix}
\sum_{i=0}^{n_4} Q_i(x) X_i B_i M^T + \sum_{j=0}^{n_0} P_j(x) X_0 B_0 B M^T \\
+ \sum_{k=0}^{n_1} Q_k^2(x) X(k) B^{(k)} M^T + \sum_{s=0}^{n_4} P_s^* x) X(x) B^s B M^T
\end{bmatrix}, \]

\( p = 0, 1, 2, \ldots, n, \) and \( q = 0, 1, 2, \ldots, n. \)

To obtain the solution of Eq. (21) under conditions Eq. (22), the following augmented matrix is constructed by replacing the last \( m \) rows in Eq. (46) by the rows matrix Eq. (44) so, we have new augmented matrix as:

\[
[W; G] = \begin{bmatrix}
w_{00} & w_{01} & \cdots & w_{0N} & \vdots & g(x_0) \\
w_{10} & w_{11} & \cdots & w_{1N} & \vdots & g(x_1) \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
w_{m-1,0} & w_{m-1,1} & \cdots & w_{m-1,N} & \vdots & g(x_m)
\end{bmatrix}.
\]

\[
\begin{bmatrix}
u_0 \\
u_{10} \\
\vdots \\
u_{m-1,0}
\end{bmatrix}
= \begin{bmatrix}
\mu_0 \\
\mu_1 \\
\vdots \\
\mu_{m-1}
\end{bmatrix}.
\]

- If \( \text{rank } W = \text{rank}[W; G] = N + 1 \) then we can write:

\[ A = W^{-1} G. \] (48)

#### 5.1. Error estimation

The collocation method is the most common spectral methods used to solve differential equations, it is easy to implement once the differentiation matrices are computed. The coefficient matrix of the collocation system is always full with a condition number behaving like \( O(N^m) \) \((m \text{ is the order of the differential equation})\) see Du (2016), Shen, Tang, and Wang (2011), and Wang, Zhang, and Zhang (2014). In addition, if the exact solution exists then the error is estimated form:

\[ E_N(x) = |y(x) - y_N(x)|, \] (49)

where \( y(x) \) the exact solution and \( y_N(x) \) approximate solution. We can easily check the accuracy of the suggested method. Since the truncated Chebyshev series Eq. (25) is an approximate solution of (21), when the solution \( y_N(x) \) and its derivatives are substituted in Eq. (21), the resulting relation must be satisfied approximately; that is, for \( x = x_p \in [0, 1], p = 0, 1, 2, \ldots, \)

\[
E(x_p) = \sum_{i=0}^{n_0} Q_i(x_p) D^i y_N(x_p) + \sum_{j=0}^{n_0} P_j(x_p) D^j y_N(x_p - \tau)
+ \sum_{k=0}^{n_1} Q_k^2(x_p) y_N^{(k)}(x_p - \tau) + \sum_{s=0}^{n_4} P_s^* x_p) y_N^{(s)}(x_p - \tau) - g(x_p) \cong 0, \]

\[ E(x_p) \leq 10^{-6} (E_p \text{ positive integer}). \] (50)

and \( E(x_p) \leq 10^{-6} (E_p \text{ positive integer}). \) If max \( 10^{-6} \) is prescribed, then the truncation limit \( N \) is increased until the difference \( E(x_p) \) at each of the points becomes smaller than the prescribed \( 10^{-6} \). On the other hand, the error can be estimated by the function

\[
E_N(x) = \sum_{i=0}^{n_0} Q_i(x) D^i y(x) + \sum_{j=0}^{n_0} P_j(x) D^j y(x - \tau)
+ \sum_{k=0}^{n_1} Q_k^2(x) y^{(k)}(x) + \sum_{s=0}^{n_4} P_s^* x) y^{(s)}(x - \tau) - g(x).
\] (51)

If \( E_N(x) \to 0 \), when \( N \) is sufficiently large enough, then the error decreases.
6. Applications and numerical results

In this section, we introduce some numerical examples for fractional order delay-differential equation to illustrate the above results. All results are obtained by using Mathematica 7 programming.

**Example 1.** Consider the fractional order-delay-differential equation:

\[ D^\alpha y(x) + D^\beta y(x-1) + y''(x) = 2 - 2.22848x^\delta \]

\[ Q_0 = Q_0 = P_0 = \text{identity}, \quad B = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 4 \\ 0 & 0 & 0 & 0 & 5 \end{bmatrix}, \quad M = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ -1 & 0 & 2 & 0 & 0 \\ 0 & -3 & 0 & 4 & 0 \\ -1 & 0 & -8 & 0 & 8 \\ 1 & 0 & -20 & 0 & 16 \end{bmatrix} \]

\[ X = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 1 \end{bmatrix}, \quad B_{0.5} = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}, \quad B_{0.7} = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}, \quad B_{-1} = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} \]

\[ X_{0.5} = \begin{bmatrix} 0.9510i \\ 0.8090i \\ 0.5877i \\ 0.3909i \\ 0.0570i + 0.7850i \end{bmatrix} \]

\[ X_{0.7} = \begin{bmatrix} 0.9510i \\ 0.8090i \\ 0.5877i \\ 0.3909i \\ 0.0570i + 0.7850i \end{bmatrix} \]

\[ X_{-1} = \begin{bmatrix} 0.9510i \\ 0.8090i \\ 0.5877i \\ 0.3909i \\ 0.0570i + 0.7850i \end{bmatrix} \]

\[ G = \begin{bmatrix} -0.317463 \\ -0.155364 \\ 0.265808 \\ 0.794654 \\ 1.30497 \end{bmatrix} \]

\[ \text{where} \quad Q_0X,B_0M^T + P_0X,B_0M^T + Q_0XB^2M^T \]

\[ = A, \quad \text{Eq. (52)} \]

The initial conditions are \( y(0) = 1, y'(0) = 0, \) and the exact solution is \( y(x) = x^2 + 1, \) where \( Q_0(x) = 1, P_0(x) = 1, g(x) = 2 - 2.25676x^4 + 3.009018547x^8. \) The fundamental matrix equation of the problem Eq. (52) is defined by

\[ \left[ Q_0X,B_0M^T + P_0X,B_0M^T + Q_0XB^2M^T \right] A = G, \quad \text{Eq. (53)} \]
After the augmented matrices of the system and condition are computed, we obtain the solution as:

\[
A = \begin{bmatrix}
3 & 0 & 1/2 & 0 & 0 & 0 \\
0 & 2 & 0 & 0 & 0 & 0 \\
1/2 & 0 & 3 & 0 & 0 & 0 \\
0 & 0 & 0 & 2 & 0 & 0 \\
0 & 0 & 0 & 0 & 2 & 0 \\
0 & 0 & 0 & 0 & 0 & 2
\end{bmatrix}. \tag{54}
\]

Then the solution of the problem Eq. (52) is.

\[
y_s(x) = \frac{3}{2}T_0(x) + \frac{1}{2}T_2(x) = x^2 + 1, \tag{55}
\]

which is the exact solution of the problem Eq. (52).

**Example 2.** Consider of the linear delay-differential equation (Gülşu & Sezer, 2006)

\[
2y''(x) + 2y'(x) - 4y(x) + y''(x-1) + y'(x-1) - 2y(x-1) = -6x^2 + 10x + 8,
\]

with the initial condition \(y(0) = 1, y'(0) = 2\) and the exact solution is \(y(x) = 2e^x + x^2 - 1\). Then for \(N = 5\) with Eqs. (25) and (35), and the fundamental matrix equation of the problem is defined by

\[
\begin{bmatrix}
Q_0^* X^T + Q_1^* X B^T M^T + Q_2^* X B^2 M^T + P_0^* X B_{-1} M^T \\
+ P_1^* X B_{-1} M^T + P_2^* X B^2 B_{-1} M^T
\end{bmatrix} A = G.
\]

Then the solution of the problem Eq. (59) is.

\[
A = \begin{bmatrix}
1/2 & 0 & 1/2 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 \\
1/2 & 0 & 1/2 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0
\end{bmatrix}. \tag{60}
\]

The approximate the solution \(y_N(x)\) introduced by present method with \(N = 5\) as:

\[
y_5(x) = \frac{3}{2}T_0(x) + \frac{1}{2}T_2(x) = x^2.
\]

This problem is mentioned in Muthukumar and Ganesh Priya (2017) using Jacobi collocation method, Table 2 shows the exact, present and Jacobi (Muthukumar & Ganesh Priya, 2017) solutions. Also, Figure 2 introduces the behaviour of the exact, approximate solutions at \(N = 6\).

**Table 1.** Comparison between exact solution, present method and Taylor method for Example 2.

| \(x\) | Exact solution | Present method \((N = 8)\) | Taylor method \((N = 8)\) |
|-------|---------------|-----------------------------|-----------------------------|
| -1    | 0.735759      | 0.735744                    | 0.735403                    |
| -0.8  | 0.538658      | 0.538665                    | 0.538445                    |
| -0.6  | 0.457623      | 0.457636                    | 0.457516                    |
| -0.4  | 0.300649      | 0.300649                    | 0.300599                    |
| -0.2  | 0.677462      | 0.677464                    | 0.77453                     |
| 0     | 1.000000      | 1.000000                    | 1.000000                    |

**Figure 1.** The behaviour of the exact solution, present method and Taylor method for at \(N = 8\).

**Figure 2.** The behaviour of the exact and approximate solutions at \(N = 6\).
Example 4. Consider the following linear initial value problem (Kazem, 2013)

\[ D^\nu y(x) + y(x) = 0. \]  \hfill (63)

The subjected initial conditions are \( y(0) = 1, y'(0) = 0 \), and the exact solution is given by:

\[ y(x) = \sum_{k=0}^{\infty} (-x^\nu)^k. \]

The fundamental matrix equation of the problem is defined by:

\[ \left[ Q(x) \quad X, B, M^T \right] A = 0. \]  \hfill (64)

By using the initial conditions we can write Eq. (64) as:

\[ A = \left[ Q(x) \quad X, B, M^T \right]^{-1} G, \]  \hfill (65)

where

\[ G = [1, \quad 0, \quad 0, \quad ..., \quad 0]^T. \]

The approximate solution given with \( N = 10 \). Table 3 gives the comparison between present method and (Kazem, 2013) with different \( \nu \) of root mean square (RMS), where RMS is given as:

\[ \text{RMS} = \sqrt{\frac{\sum_{k=1}^{M} (y(x_k) - y_N(x_k))^2}{M}}. \]

where \( y(x_k) \) and \( y_N(x_k) \) are achieved by exact and numerical solution on \( x_k \) and \( M \) is number of test points. In addition, Figure 3 shows the numerical results of \( y_N(x) \) for \( N = 10 \) and \( \nu = 0.2, 0.5, 0.8 \) and 1.2.

### Table 3. Comparison of the root mean square (RMS) error for some various \( \nu \) and at \( N = 10 \).

| \( \nu \) | RMS suggested method | RMS Jacobi method (Kazem, 2013) |
|---|---|---|
| 0.2 | \( 6.96 \times 10^{-3} \) | \( 6.14 \times 10^{-2} \) |
| 0.5 | \( 5.86 \times 10^{-3} \) | \( 5.60 \times 10^{-2} \) |
| 0.8 | \( 3.71 \times 10^{-3} \) | \( 8.00 \times 10^{-4} \) |
| 1.2 | \( 1.73 \times 10^{-3} \) | \( 3.09 \times 10^{-1} \) |
| 1.5 | \( 1.35 \times 10^{-3} \) | \( 2.03 \times 10^{-3} \) |

Example 5. Consider the fractional-delay-differential equation (Rahimkhani et al., 2017)

\[ D^\nu y(x) - y(x - \tau) + y(x) = \frac{2}{\Gamma(3 - \alpha)} x^{2-x} - \frac{1}{\Gamma(2 - \alpha)} x^{1-x} + 2\tau x - \tau, \]  \hfill (66)

with the initial condition \( y(0) = 0, y'(0) = 0 \) and the exact solution is \( y(x) = x^{2-x} \) when \( \alpha = 1, \tau = 0.01 \) by using Eqs. (25) and (35) with \( Q_0(x) = 1, P_0(x) = 1, Q_0(x) = 1, g(x) = \cdot N = 6. \) Then the fundamental matrix equation of the problem is defined by

\[ \left[ Q_0(x) \quad X, B, M^T \right] A = G. \]  \hfill (67)

Our numerical results are presented in Table 4.

Figure 4 displays the approximate solutions obtained for values of \( x = 0.5, 0.7, 0.9 \) and the exact solution with \( N = 6 \) and \( \tau = 0.01 \). From these results, it is seen that the approximate solutions converge to the exact solution.

Figure 5 displays the approximate solutions obtained for different values of \( x \) with the exact solution.

Example 6. Consider the following fractional-delay-differential equation (Rahimkhani et al., 2017; Saeed & ur Rehman, 2014):

\[ D^\nu y(x) + y(x - 0.3) = e^{-x} x^{0.3}, \quad 1 < x \leq 3 \]  \hfill (68)

### Table 4. Comparison of the values of exact, approximate solutions of the problem Eq. (66) for different values of \( x \) and the absolute errors at \( \tau = 0.01 \) for Example 5.

| \( x \) | Exact solution \( (N = 10) \) | Suggested method \( (N = 6) \) | exact error \( (N = 6) \) | absolute error \( (N = 6) \) |
|---|---|---|---|---|
| 0 | 0.000000 | 0.000000 | 0 | 0 |
| 0.2 | 0.160000 | 0.160000 | 7.5856 \times 10^{-14} | 8.2183 \times 10^{-14} |
| 0.4 | 0.240000 | 0.240000 | 3.9079 \times 10^{-14} | 1.1129 \times 10^{-16} |
| 0.6 | 0.320000 | 0.320000 | 1.4516 \times 10^{-14} | 5.1566 \times 10^{-14} |
| 0.8 | 0.400000 | 0.400000 | 7.9603 \times 10^{-14} | 3.2345 \times 10^{-14} |
| 1 | 0.400000 | 0.400000 | 0 | 0 |

Figure 3. The numerical results of \( y_N(x) \) for \( N = 10 \) and \( \nu = 0.2, 0.5, 0.8 \) and 1.2.

Figure 4. The comparison of \( y_N(x) \) for \( N = 6, \tau = 0.01 \), with \( \alpha = 0.5, 0.7, 0.9 \) and the exact solution for Example 5.
with the initial condition \( y(0) = 1, y'(0) = -1 \) and the exact solution when \( n = 3 \) is \( y(x) = e^{-x} \). Table 5 displays the approximate solutions obtained by present method, the Hermite wavelet method (Saeed & ur Rehman, 2014), Bernoulli wavelet method (Rahimkhani et al., 2017) and the exact solution. By using Eqs. (25) and (35) with \( Q_0(x) = 1, Q_0'(x) = 1, p_0(x) = 1 \) at \( N = 6 \). Then the fundamental matrix equation of the problem is defined by

\[
\left[ Q_0(x) X_b M^T + Q_0'(x) XM^T + P_0(x) XB \ldots M^T \right] A = G.
\]

(69)

After the augmented matrices of the system and condition are computed, we obtain the coefficient matrix on the form:

\[
A = \begin{bmatrix}
1 & 0 & 1 & 0 & 0 & 0 \\
0 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}.
\]

(73)

Then the solution is of Equation (71)

\[
y(x) = \frac{1}{2} T_0(x) + \frac{1}{2} T_2(x) = x^2,
\]

(74)

which is the exact solution of the boundary value problem Eq. (71).

7. Conclusion

In this work, the general form of fractional order Delay-differential equations (GFDEEs) is presented. The spectral collocation method is used for solving GFDEEs. All terms in the proposed equation reduced by an operational matrices based on Chebyshev polynomials to matrix form. The accuracy of this method is obtained by many numerical examples. Finally, we used the Mathematica 7 to calculate our results.

Disclosure Statement

No potential conflict of interest was reported by the authors.

ORCID

Khalid K. Ali  
http://orcid.org/0000-0002-7801-2760

References

Akyüz, A., & Sezer, M. (2003). Chebyshev polynomial solutions of systems of high-order linear differential equations with variable coefficients. Applied Mathematics and Computation, 144(2–3), 237–247, doi:10.1016/S0096-3003(02)00403-4

Bhrawy, A. H., & Alofi, A. S. (2013). The operational matrix of fractional integration for shifted Chebyshev polynomials. Applied Mathematics Letters, 26(1), 25–31, doi:10.1016/j.aml.2012.01.027

Bhrawy, A. H., Alofi, A. S., & Ezz-Eldien, S. S. (2011). A quadrature tau method for fractional differential equations with variable coefficients. Applied Mathematics Letters, 24(12), 2146–2152, doi:10.1016/j.aml.2011.06.016

Doha, E. H., & Bhrawy, A. H. (2008). Efficient spectral-Galerkin algorithms for direct solution of fourth-order differential equations using Jacobi polynomials. Applied Numerical Mathematics, 58(8), 1224–1244, doi:10.1016/j.apnum.2007.07.001

Table 5. Comparison of the approximate solutions obtained by present method, Hermite wavelet method (Saeed & ur Rehman, 2014), Bernoulli wavelet method (Rahimkhani et al., 2017) and the exact solution for Example 6.

| x     | Exact solution | present method (N = 6) \( n = 3 \) | (Saeed & ur Rehman, 2014) (N = 7) \( n = 3 \) | (Rahimkhani et al., 2017) (N = 7) \( n = 3 \) | present method (N = 6) \( n = 2.8 \) | present method (N = 6) \( n = 2.6 \) |
|-------|----------------|-------------------------------|---------------------------------|---------------------------------|-------------------------------|-------------------------------|
| 0     | 1.0000         | 1.0000                        | 1.0000                          | 1.0000                          | 1.0000                        | 1.0000                        |
| 0.2   | 0.8187         | 0.8187                        | 0.8187                          | 0.8187                          | 0.8185                        | 0.8185                        |
| 0.4   | 0.6703         | 0.6703                        | 0.6703                          | 0.6703                          | 0.6685                        | 0.6682                        |
| 0.6   | 0.5488         | 0.5488                        | 0.5488                          | 0.5488                          | 0.5480                        | 0.5488                        |
| 0.8   | 0.4493         | 0.4493                        | 0.4494                          | 0.4493                          | 0.4366                        | 0.4281                        |

Figure 5. Numerical solutions obtained by the proposed method at different \( x \) and exact solution at \( x = 3, N = 6 \).
Xu, M. Q., & Lin, Y. Z. (2016). Simplified reproducing kernel method for fractional differential equations with delay. *Applied Mathematics Letters, 52*, 156–161. doi:10.1016/j.aml.2015.09.004

Yang, C. (2012). Chebyshev polynomial solution of nonlinear integral equations. *Journal of the Franklin Institute, 349*(3), 947–956. doi:10.1016/j.jfranklin.2011.10.023