

# Lucas Collocation Method for the Solution of Differential Difference Equations

Deniz Elmaci<sup>1</sup> and Nurcan Baykus Savaseneril<sup>2</sup>

<sup>1</sup>Dokuz Eylul University, Bergama Vocational School, Izmir, Turkey.

<sup>2</sup>Dokuz Eylul University, Izmir Vocational School, Izmir, Turkey.

Received: 9 August 2022, Accepted: 28 September 2022

Published online: 5 December 2022.

**Abstract:** In order to solve  $m$ th-order linear differential difference equations with variable coefficients under mixed conditions, this research suggests a combined operational matrix approach based on Lucas polynomials. The simplicity of the proposed method's application is a benefit. The technique simplifies the provided problem by turning it into a matrix equation. Absolute errors are used to validate illustrative examples. The solutions are enhanced by residual error estimates. The results, which are displayed in graphs and tables, are contrasted with the accepted approaches in the literature.

**Keywords:** Differential-difference equations, Lucas polynomials, collocation and matrix methods.

## 1 Introduction

Differential-difference equations find applications in areas that are not covered by classical models of mathematical physics, namely, in models of nonlinear optics, in nonclassical diffusion models, in biomathematical applications, and in the theory of multilayered plates and shells [1]. This is explained by the nonlocal nature of functional-differential equations: unlike in classical differential equations, where all derivatives of the unknown function are related at the same point, in functional-differential equations, these terms can be related at different points, thus substantially expanding the generality of the model [2,3,4,5,6].

Consider the following differential-difference equation,

$$\sum_{k=0}^m P_k(t)y^{(k)}(t) + \sum_{j=0}^J Q_j(t)y^{(j)}(\alpha t + \beta) = g(t) \quad (1)$$

with the mixed condition

$$\sum_{k=0}^{m-1} \left( a_{ks}y^{(k)}(-1) + b_{ks}y^{(k)}(0) + c_{ks}y^{(k)}(1) \right) = \lambda_s, \quad s = 0, 1, \dots, m-1 \quad (2)$$

where  $P_k(t)$ ,  $Q_j(t)$  and  $g(t)$  are functions defined on the interval  $-1 \leq t \leq 1$ ;  $\alpha$ ,  $\beta$ ,  $a_{ks}$ ,  $b_{ks}$ ,  $c_{ks}$  and  $\lambda_s$  are appropriate constants;  $y(t)$  is an unknown solution function to be determined.

For our purpose, we assume the approximate solution of the problem Eq.(1)-Eq.(2) in the truncated Lucas polynomials form

\* Corresponding author e-mail: [nurcan.savaseneril@deu.edu.tr](mailto:nurcan.savaseneril@deu.edu.tr)

$$y(t) \cong y_N(t) = \sum_{n=0}^N a_n L_n(t), \quad -1 \leq t \leq 1 \tag{3}$$

where  $a_n, n = 0, 1, 2, \dots, N$  are unknown coefficients to be determined and  $L_n(t)$  indicates the Lucas polynomials which are originally studied in 1970 by Bcknell. Lucas polynomials are defined recursively as follows [7, 8, 9].

$$L_{n+1}(t) = tL_n(t) + L_{n-1}(t), \quad n \geq 1, \quad L_0(t) = 2, L_1(t) = t. \tag{4}$$

Their explicit form for  $n \geq 1$  is

$$L_n(t) = \sum_{k=0}^{\frac{n}{2}} \frac{n}{n-k} \binom{n-k}{k} t^{n-2k} \tag{5}$$

where  $x$  is the largest integer smaller than or equal to  $x$ .

By using Eq.(4) and Eq.(5) the first Lucas polynomials respectively are given by

$$\begin{aligned} L_0(t) &= 2, & L_1(t) &= t, & L_2(t) &= t^2 + 2, & L_3(t) &= t^3 + 3t, \\ L_4(t) &= t^4 + 4t^2 + 2, & L_5(t) &= t^5 + 5t^3 + 5t, & L_6(t) &= t^6 + 6t^4 + 9t^2 + 2, \end{aligned}$$

## 2 Materials and Methods

### 2.1 Matrix Relations

The following process is used in this section to convert the expressions defined in Eq.(1) and Eq.(2) into matrix forms: First, the derivatives of the function  $y(t)$  defined by Eq.(3) can be expressed in matrix form.

$$y(t) \cong y_N(t) = \mathbf{L}(t)\mathbf{A}, \quad \mathbf{L}(t) = \mathbf{T}(t)\mathbf{D}^T \tag{6}$$

where

$$\mathbf{L}(t) = [L_0(t) \ L_1(t) \ \dots \ L_N(t)], \quad \mathbf{A} = [a_0 \ a_1 \ \dots \ a_N]^T$$

$$\mathbf{T}(t) = [1 \ t \ t^2 \ \dots \ t^N].$$

If  $N$  is odd,

$$\mathbf{D} = \begin{bmatrix} 2 & 0 & 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & \frac{1}{1} \begin{pmatrix} 1 \\ 0 \end{pmatrix} & 0 & 0 & 0 & 0 & \dots & 0 \\ \frac{2}{1} \begin{pmatrix} 1 \\ 1 \end{pmatrix} & 0 & \frac{2}{2} \begin{pmatrix} 2 \\ 0 \end{pmatrix} & 0 & 0 & 0 & \dots & 0 \\ 0 & \frac{3}{2} \begin{pmatrix} 2 \\ 1 \end{pmatrix} & 0 & \frac{3}{3} \begin{pmatrix} 3 \\ 0 \end{pmatrix} & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & & & \\ \frac{N-1}{\binom{N-1}{2}} \begin{pmatrix} \frac{N-1}{2} \\ \frac{N-1}{2} \end{pmatrix} & 0 & \frac{N-1}{\binom{N+1}{2}} \begin{pmatrix} \frac{N+1}{2} \\ \frac{N-3}{2} \end{pmatrix} & 0 & \dots & \dots & \frac{N-1}{\binom{2N-2}{2}} \begin{pmatrix} \frac{2N-1}{2} \\ 0 \end{pmatrix} & 0 \\ 0 & \frac{N}{\binom{N+1}{2}} \begin{pmatrix} \frac{N+1}{2} \\ \frac{N-1}{2} \end{pmatrix} & 0 & \frac{N}{\binom{N+3}{2}} \begin{pmatrix} \frac{N+3}{2} \\ \frac{N-3}{2} \end{pmatrix} & \dots & \dots & 0 & \frac{N}{N} \begin{pmatrix} N \\ 0 \end{pmatrix} \end{bmatrix}$$

and if  $N$  is even,

$$\mathbf{D} = \begin{bmatrix} 2 & 0 & 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & \frac{1}{1} \begin{pmatrix} 1 \\ 0 \end{pmatrix} & 0 & 0 & 0 & 0 & \dots & 0 \\ \frac{2}{1} \begin{pmatrix} 1 \\ 1 \end{pmatrix} & 0 & \frac{2}{2} \begin{pmatrix} 2 \\ 0 \end{pmatrix} & 0 & 0 & 0 & \dots & 0 \\ 0 & \frac{3}{2} \begin{pmatrix} 2 \\ 1 \end{pmatrix} & 0 & \frac{3}{3} \begin{pmatrix} 3 \\ 0 \end{pmatrix} & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & \frac{N-1}{\left(\frac{N}{2}\right)} \begin{pmatrix} \frac{N}{2} \\ \frac{N-2}{2} \end{pmatrix} & 0 & \frac{N-1}{\left(\frac{N+2}{2}\right)} \begin{pmatrix} \frac{N+2}{2} \\ \frac{N-4}{2} \end{pmatrix} & \dots & \dots & 0 & \frac{N-1}{N-1} \begin{pmatrix} N-1 \\ 0 \end{pmatrix} \\ \frac{N}{\left(\frac{N}{2}\right)} \begin{pmatrix} \frac{N}{2} \\ \frac{N}{2} \end{pmatrix} & 0 & \frac{N}{\left(\frac{N+2}{2}\right)} \begin{pmatrix} \frac{N+2}{2} \\ \frac{N-2}{2} \end{pmatrix} & 0 & \dots & \dots & \frac{N}{\left(\frac{2N}{2}\right)} \begin{pmatrix} \frac{2N}{2} \\ 0 \end{pmatrix} & 0 \end{bmatrix}$$

From the matrix relations Eq. (6), it follows that

$$y_N(t) = \mathbf{T}(t) \mathbf{D}^T \mathbf{A}, \tag{7}$$

Besides, it is well known that the relation between the matrix  $\mathbf{T}(t)$  and its derivatives are

$$\mathbf{T}^{(k)}(t) = \mathbf{T}(t) \mathbf{B}^k$$

where

$$\mathbf{B} = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 2 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & N \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad \mathbf{B}^0 = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & \dots & 1 \end{bmatrix}$$

By using Eq.(6)-(7), we have the matrix relation

$$y_N^{(k)}(t) = \mathbf{T}(t) \mathbf{B}^k \mathbf{D}^T \mathbf{A}, \quad k = 0, 1, 2, \dots \tag{8}$$

By putting  $t \rightarrow \alpha t + \beta$  in the relation Eq.(8)

$$y_N^{(k)}(\alpha t + \beta) = \mathbf{T}(\alpha t + \beta) \mathbf{B}^k \mathbf{D}^T \mathbf{A}, \quad k = 0, 1, 2, \dots \tag{9}$$

$$\mathbf{T}(\alpha t + \beta) = \mathbf{T}(t) \mathbf{B}(\alpha, \beta) \tag{10}$$

where

$$\mathbf{B}(\alpha, \beta) = \begin{bmatrix} \binom{0}{0} \alpha^0 \beta^0 & \binom{1}{0} \alpha^0 \beta^1 & \binom{2}{0} \alpha^0 \beta^2 & \dots & \binom{N}{0} \alpha^0 \beta^N \\ 0 & \binom{1}{1} \alpha^1 \beta^0 & \binom{2}{1} \alpha^1 \beta^1 & \dots & \binom{N}{1} \alpha^1 \beta^{N-1} \\ 0 & 0 & \binom{2}{2} \alpha^2 \beta^0 & 0 & \binom{N}{2} \alpha^2 \beta^{N-2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \binom{N}{N} \alpha^N \beta^0 \end{bmatrix}$$

To obtain the Lucas polynomial solution of Eq.(1) in the form Eq.(3) we firstly compute the Lucas coefficients by means of the collocation points defined by

$$t_i = a + \frac{b-a}{N}i, \quad i = 0, 1, \dots, N.$$

The following steps are taken to obtain the matrix equation system:

$$\sum_{k=0}^m P_k(t_i) y^{(k)}(t_i) + \sum_{j=0}^J Q_j(t_i) y^{(j)}(\alpha t_i + \beta) = g(t_i) \quad (11)$$

It is constructed the fundamental matrix equation corresponding to Eq.(1). For this purpose, it is substituted the matrix relations Eq.(8)-(10) into Eq.(1) and simplified, obtained the fundamental matrix equation

$$\sum_{k=0}^m P_k(t_i) \mathbf{T}(t_i) \mathbf{B}^k \mathbf{D}^T \mathbf{A} + \sum_{j=0}^J Q_j(t_i) \mathbf{T}(t_i) \mathbf{B}(\alpha, \beta) \mathbf{B}^k \mathbf{D}^T \mathbf{A} = g(t_i) \quad (12)$$

or briefly,

$$\sum_{k=0}^m \mathbf{P}_k \mathbf{T} \mathbf{B}^k \mathbf{D}^T \mathbf{A} + \sum_{j=0}^J \mathbf{Q}_j \mathbf{T} \mathbf{B}(\alpha, \beta) \mathbf{B}^k \mathbf{D}^T \mathbf{A} = \mathbf{G} \quad (13)$$

where

$$\mathbf{P}_k = \begin{bmatrix} P_k(t_0) & 0 & \dots & 0 \\ 0 & P_k(t_1) & 0 & \vdots \\ \vdots & \dots & \ddots & 0 \\ 0 & \dots & 0 & P_k(t_N) \end{bmatrix}, \quad \mathbf{Q}_j = \begin{bmatrix} Q_j(t_0) & 0 & \dots & 0 \\ 0 & Q_j(t_1) & 0 & \vdots \\ \vdots & \dots & \ddots & 0 \\ 0 & \dots & 0 & Q_j(t_N) \end{bmatrix}$$

$$\mathbf{T} = \begin{bmatrix} \mathbf{T}(t_0) \\ \mathbf{T}(t_1) \\ \vdots \\ \mathbf{T}(t_N) \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} g(t_0) \\ g(t_1) \\ \vdots \\ g(t_N) \end{bmatrix}, \quad \mathbf{A} = \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_N \end{bmatrix},$$

Besides, the fundamental matrix equation Eq.(13) can be expressed in the form

$$\mathbf{W} \mathbf{A} = \mathbf{G} \quad \Leftrightarrow \quad [\mathbf{W} : \mathbf{G}] \quad (14)$$

where

$$\mathbf{W} = \sum_{k=0}^m \mathbf{P}_k \mathbf{T} \mathbf{B}^k \mathbf{D}^T + \sum_{j=0}^J \mathbf{Q}_j \mathbf{T} \mathbf{B}(\alpha, \beta) \mathbf{B}^k \mathbf{D}^T = [w_{mn}]; \quad m, n = 0, 1, \dots, N.$$

Now we can obtain the corresponding matrix form for the initial conditions Eq.(2), by means of the relation Eq.(8),

$$\mathbf{U}_s \mathbf{A} = \lambda_s \Leftrightarrow [\mathbf{U}_s : \lambda_s]; \quad s = 0, 1, \dots, m-1. \tag{15}$$

such that

$$\mathbf{U}_s = \sum_{k=0}^{m-1} (a_{jk} \mathbf{T}(a) + b_{jk} \mathbf{T}(b)) \mathbf{B}^k \mathbf{D}^T \mathbf{A} = [u_{j0} \ u_{j1} \ \dots \ u_{jN}] \tag{16}$$

After substituting any m rows of the augmented matrix (14) with the m row matrices (16), we finally get the new matrix as the answer to the problems (1)-(2).

$$\widetilde{\mathbf{W}} \mathbf{A} = \widetilde{\mathbf{G}} \Rightarrow [\widetilde{\mathbf{W}} : \widetilde{\mathbf{G}}] \tag{17}$$

In Eq.(17), if  $\text{rank} \widetilde{\mathbf{W}} = \text{rank} [\widetilde{\mathbf{W}} : \widetilde{\mathbf{G}}] = N + 1$ , then the coefficient matrix  $\mathbf{A}$  is uniquely determined and the solution of the problem Eq.(1)-(2) is obtained as

$$y_N(t) = \mathbf{L}(t) \mathbf{A} = \mathbf{T}(t) \mathbf{D}^T \mathbf{A}$$

### 3 Residual Error Analysis

By employing the residual correction method, we build an error estimation strategy for the Lucas polynomial approximations of the problem Eq.(1)-(2), and we then use this technique to improve the approximation.

To begin with, the residual function of the method is

$$R_N(t) = L[y_N(t)] - g(t) \tag{18}$$

where  $L[y_N(t)] \cong g(t)$  and  $y_N(t)$  is the Lucas polynomial solution Eq.(3) of the problems Eq.(1)-(2). For  $t = t_l \in [-1, 1]$ ,  $l = 0, 1, 2, \dots$ ;  $R_N(t_l) \leq 10^{-k_l}$  ( $k_l$  is any positive integer).

Further, the error function  $e_N(t)$  can be determined as

$$e_N(t) = y(t) - y_N(t) \tag{19}$$

where  $y(t)$  is the exact solution of the problem Eq.(1)-(2). From Eqs.(1), (2), (18) and (19), we obtain the system of the error differential equations

$$L[e_N(t)] = L[y(t)] - L[y_N(t)] = -R_N(t) \tag{20}$$

and the error problem

$$\begin{aligned} \sum_{k=0}^m P_k(t) e_N^{(k)}(t) + \sum_{j=0}^J Q_j(t) e_N^{(j)}(\alpha t + \beta) &= -R_N(t) \\ e_{jN}^{(k)}(a) &= 0, \quad j = 1, 2, \dots, J, \quad k = 0, 1, \dots, m-1 \end{aligned} \tag{21}$$

The error problem Eq.(21) can be settled by using the presented method in Section 2. So, we obtain the approximation  $e_{N,M}(t)$  to  $e_N(t)$  as follows:

$$e_{N,M}(t) = \sum_{n=0}^M a_N^* L_N(t), \quad M > N, \quad j = 1, 2, \dots, J. \tag{22}$$

As a result, using the polynomials  $y_N(t)$  and  $e_{N,M}(t)$ , the corrected Lucas polynomial solution  $y_{N,M}(t) = y_N(t) + e_{N,M}(t)$  is achieved. Additionally, the error function  $e_N(t) = y(t) - y_N(t)$ , the estimated error function  $e_{N,M}(t)$  and the corrected error function  $E_{N,M}(t) = e_N(t) - e_{N,M}(t) = y(t) - y_{N,M}(t)$  constructed [10, 11].

## 4 Numerical Examples

In order to demonstrate the correctness and efficiency of the procedure, some numerical examples of the problem Eq. (1) are provided in this section.

**Example 4.1.** Let us first consider the problem

$$y'(t) = ty(t) + \frac{1}{2}e^{-t}y(t-1) + e^t - te^t - \frac{1}{2}e^{-1}$$

with the initial condition  $y(0) = 1$  [12].

We approximate the solution  $y(t)$  by the polynomial

$$y(t) = y_N(t) = \sum_{n=0}^3 a_n L_n(t), \quad -1 \leq t \leq 1$$

$P_1(t) = 1$ ,  $P_0(t) = -t$ ,  $Q_1(t) = 0$ ,  $Q_0(t) = \frac{e^{-t}}{2}$ ,  $g(t) = e^t - te^t - \frac{1}{2}e^{-1}$   
and the collocation points for  $a = -1$ ,  $b = 1$  and  $N = 3$  are computed as

$$\left\{ t_0 = -1, \quad t_1 = -\frac{1}{3}, \quad t_2 = \frac{1}{3}, \quad t_3 = 1 \right\}.$$

Following the procedure in Section 2, the fundamental matrix equation of the given equation becomes

$$\sum_{k=0}^1 \mathbf{P}_k \mathbf{T} \mathbf{B}^k \mathbf{D}^T \mathbf{A} + \sum_{j=0}^1 \mathbf{Q}_j \mathbf{T} \mathbf{B}(1, -1) \mathbf{B}^j \mathbf{D}^T \mathbf{A} = \mathbf{G}$$

where

$$\mathbf{P}_0 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{1}{3} & 0 & 0 \\ 0 & 0 & -\frac{1}{3} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{P}_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{Q}_0 = \begin{bmatrix} \frac{e}{2} & 0 & 0 & 0 \\ 0 & \frac{e^{1/3}}{2} & 0 & 0 \\ 0 & 0 & \frac{e^{-1/3}}{2} & 0 \\ 0 & 0 & 0 & \frac{e^{-1}}{2} \end{bmatrix}, \quad \mathbf{D}^T = \begin{bmatrix} 2 & 0 & 2 & 0 \\ 0 & 1 & 0 & 3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$\mathbf{B}^0 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 3 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{B}(1, -1) = \begin{bmatrix} 1 & -1 & 1 & -1 \\ 0 & 1 & -2 & 3 \\ 0 & 0 & 1 & -3 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{T} = \begin{bmatrix} 1 & -1 & 1 & -1 \\ 1 & -\frac{1}{3} & \frac{1}{9} & -\frac{1}{27} \\ 1 & \frac{1}{3} & \frac{1}{9} & \frac{1}{27} \\ 1 & 1 & 1 & 1 \end{bmatrix},$$

$$\mathbf{W} = \begin{bmatrix} -\frac{719}{1001} & \frac{1457}{536} & -\frac{1109}{155} & \frac{3007}{143} \\ -\frac{1307}{1793} & \frac{1037}{570} & -\frac{590}{227} & \frac{4378}{589} \\ -\frac{675}{488} & \frac{980}{869} & -\frac{806}{883} & \frac{10246}{2689} \\ -\frac{3450}{1457} & 0 & -\frac{1993}{1457} & 2 \end{bmatrix}, \mathbf{G} = \begin{bmatrix} \frac{1001}{1814} \\ \frac{3257}{4222} \\ \frac{1004}{1345} \\ \frac{268}{-1457} \end{bmatrix}$$

The augmented matrix for this fundamental matrix equation is

$$[\tilde{\mathbf{W}} \ ; \ \tilde{\mathbf{G}}] = \begin{bmatrix} -\frac{719}{1001} & \frac{1457}{536} & -\frac{1109}{155} & \frac{3007}{143} & ; & \frac{1001}{1814} \\ -\frac{1307}{1793} & \frac{1037}{570} & -\frac{590}{227} & \frac{4378}{589} & ; & \frac{3257}{4222} \\ -\frac{675}{488} & \frac{980}{869} & -\frac{806}{883} & \frac{10246}{2689} & ; & \frac{1004}{1345} \\ 2 & 0 & 2 & 0 & ; & 1 \end{bmatrix}$$

Solving this system,  $\mathbf{A}$  is obtained as  $\mathbf{A} = \begin{bmatrix} -\frac{39}{23113} & \frac{461}{648} & \frac{446}{889} & \frac{145}{1382} \end{bmatrix}$

Thus, the solution of the problem becomes

$$y_3(t) = 0.10492t^3 + 0.50169t^2 + 1.02618t + 1$$

for different values of  $N$  as follows:

$$y_5(t) = 0.00582t^5 + 0.042026t^4 + 0.16898t^3 + 0.49995t^2 + 0.99963t + 1$$

$$y_8(t) = 0.00002t^8 + 0.00020t^7 + 0.00140t^6 + 0.00833t^5 + 0.04166t^4 + 0.16667t^3 + 0.5t^2 + t + 1$$

$$y_{10}(t) = 2.03 * 10^{-7}t^{10} + 2.76 * 10^{-6}t^9 + 0.00002t^8 + 0.0002t^7 + 0.0014t^6 + 0.0083t^5 + 0.0417t^4 + 0.1667t^3 + 0.5t^2 + t + 1$$

which are the approximate solution expanded for  $N = 3, 5, 8, 10$  as  $y(t) = e^t$

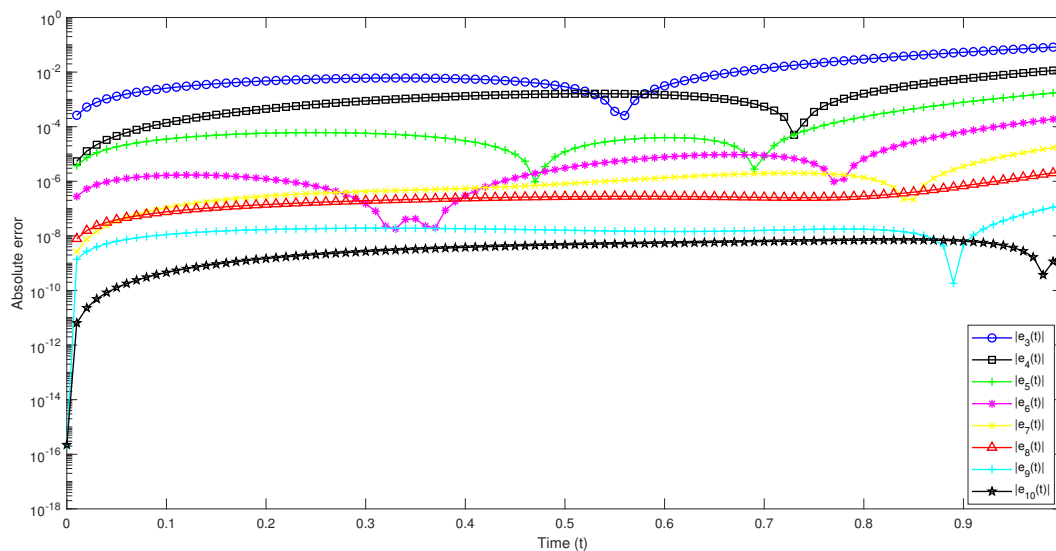
Some results from the solutions of the example are tabulated for  $N = 3, 5, 8, 10$  in Table 1. Furthermore, the results obtained by the proposed method are compared with the results of Gegenbauer polynomials given in [12]. The tables show that, the result obtained by the current approach is almost the same as the results of the exact solution. The current approach is practical and efficient as well. Fig. 1 depicts the numerical solution of the absolute errors in Example 4.1. As the integer  $N$  is increased, the error goes down.

**Table 1:** Comparisons of numerical results for N= 5, 8, 10 in Example 4.1.

t	Exact solution	Present method (N=5)	Present method (N=8)	Present method (N=10)
-1.0	0.367879	0.367550	0.367878	0.367879
-0.5	0.606531	0.606495	0.606530	0.606531
0.0	1	1	1	1
0.5	1.648721	1.648733	1.648721	1.648721
1.0	2.718282	2.716404	2.718280	2.718282
<b>CPU time</b>		0.860 s	0.883 s	0.918 s

**Table 2:** Comparisons of absolute errors in Example 4.1.

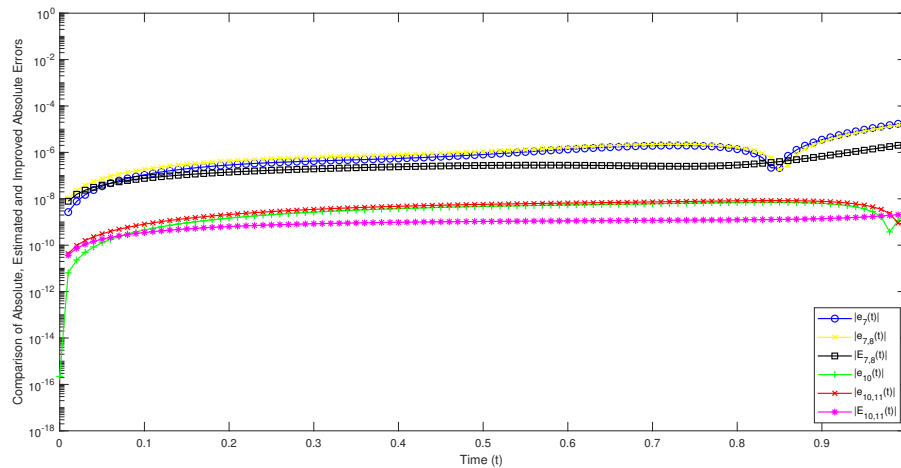
t	Gegenbauer method $e_5[12]$	Present method $e_5$	Present method $e_8$	Present method $e_{10}$
-1.0	2.7944e-04	5.8051e-05	1.5655e-06	2.4749e-10
-0.5	1.1910e-05	2.0165e-05	2.8968e-07	1.9899e-08
0.0	0	0	0	0
0.5	9.9783e-06	3.0710e-06	2.7317e-07	4.7011e-09
1.0	0.0019	0.0002	2.2649 e-06	2.9354e-09



**Fig. 1:** The absolute errors of Example 4.1 for  $3 \leq N \leq 10$ .

Additionally, the residual error analysis provides the improved numerical results as seen in Fig 2.





**Fig. 2:** Comparison of Absolute, Estimated and Improved Absolute Errors of Example 4.1.

**Example 4.2.** Consider the problem

$$y''(t) - y'(t) + y(t) - y(t + 1) + y(t + 2) = -\cos t - \sin(t + 1) + \sin(t + 2)$$

with the initial condition  $y(0) = 0, \quad y'(0) = 1$  [13].

The solution of the problem for different values of N becomes as follows:

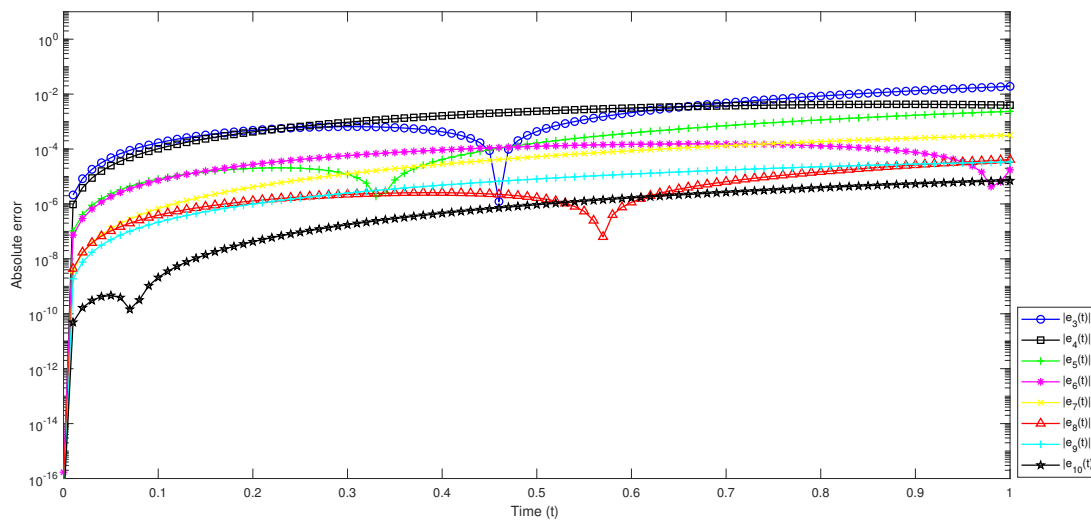
$$y_3(t) = -0.11764t^3 - 0.02174t^2 + t$$

$$y_8(t) = 0.00002t^8 - 0.00022t^7 - 6.67 * 10^{-6}t^6 + 0.00833t^5 + 0.00006t^4 - 0.16662t^3 - 0.00004t^2 + t + 7.39 * 10^{-17}$$

which are the approximate solution expanded for  $N = 3, 8$  as  $y(t) = \sin(t)$ . Table 3 shows comparison of the results obtained by our method and the Laguerre polynomials method given in [13] of Example 4.2. As can be inferred from the table, the result obtained by the present method gives better results for small values of N.

**Table 3:** Comparisons of absolute errors in Example 4.2.

t	Laguerre method $e_{10}$ [13]	Present method $e_8$	Present method $e_9$	Present method $e_{10}$
0.1	0.188e-02	3.885e-07	2.163e-07	2.096e-09
0.2	0.703e-02	1.292e-06	9.905e-07	4.190e-08
0.3	0.146e-01	2.217e-06	2.491e-06	1.738e-07
0.4	0.238e-01	2.564e-06	4.844e-06	4.537e-07
0.5	0.338e-01	1.666e-06	8.109e-06	9.332e-07
0.6	0.435e-01	1.146e-06	1.226e-05	1.654e-06
0.7	0.521e-01	6.471e-06	1.717e-05	2.638e-06
0.8	0.586e-01	1.476e-05	2.261e-05	3.888e-06



**Fig. 3:** The absolute errors of Example 4.2 for  $3 \leq N \leq 10$ .

**Example 4.3.** Consider the problem

$$y''(t) + 2y'(t) - y'(2t-1) + y'(t+1) - 3y(t-2) = 6e^t - 2e^{2t-1} + 2e^{t+1} - 6e^{t-2}$$

with the initial condition  $y(0) = 2$ ,  $y'(0) = 2$  [12].

The solution of the problem for different values of N becomes as follows:

$$y_3(t) = 0.26965t^3 + 1.16313t^2 + 2t + 2$$

$$y_6(t) = 0.00135t^6 + 0.01586t^5 + 0.08717t^4 + 0.33754t^3 + 1.00075t^2 + 2t + 2$$

$$y_{10}(t) = 2.39 \times 10^{-7}t^{10} + 4.74 \times 10^{-6}t^9 + 0.00005t^8 + 0.0004t^7 + 0.00279t^6 + 0.01666t^5 + 0.08330t^4 + 0.33332t^3 + t^2 + 2t + 2$$

which are the approximate solution expanded for  $N = 3, 6, 10$  as  $y(t) = 2e^t$

The numerical solution of the absolute errors in Example 4.3 are depicted in Fig. 4. As the integer N is increased, the error goes down.

Absolute errors of the approximate solutions, the estimated solutions and the improved approximate solutions will be given in Fig. 5.

**Example 4.4.** Consider the problem

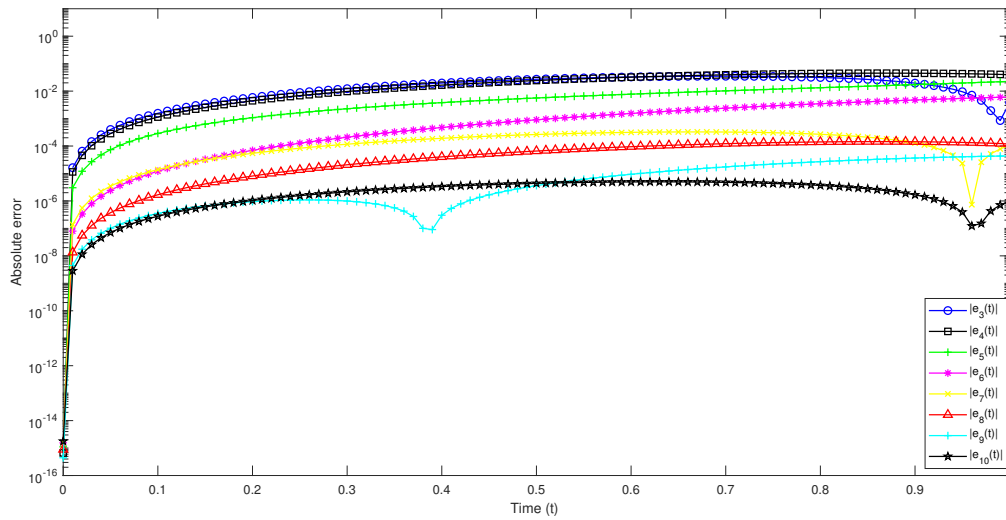
$$y'''(t) - ty''(2t) + y'(t) + y\left(\frac{t}{2}\right) = \cos(2t) + \cos\left(\frac{t}{2}\right)$$

with the initial condition  $y(0) = 1$ ,  $y'(0) = 0$ ,  $y''(0) = -1$  [12].

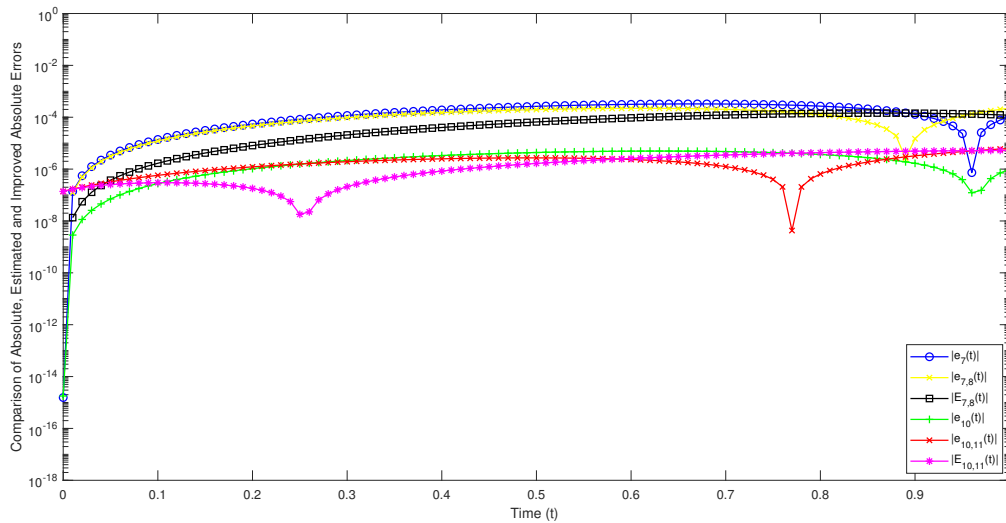
The solution of the problem for different values of N becomes as follows:

$$y_3(t) = -0.05774t^3 + 0.45323t^2 + 6.94 \times 10^{-18}t + 1$$

$$y_6(t) = -0.00110t^6 + 0.00018t^5 + 0.04155t^4 + 5.42 \times 10^{-20}t^3 - 0.53028t^2 - 1.63 \times 10^{-19}t + 1$$



**Fig. 4:** The absolute errors of Example 4.3 for  $3 \leq N \leq 10$ .



**Fig. 5:** Comparison of Absolute, Estimated and Improved Absolute Errors of Example 4.3.

$$y_{10}(t) = -2.36 * 10^{-7}t^{10} + 6.07 * 10^{-8}t^9 + 0.00002t^8 - 2.11 * 10^{-7}t^7 - 0.00139t^6 - 3.46 * 10^{-9}t^5 + 0.04167t^4 - 2.38 * 10^{-22}t^3 - 0.5t^2 + 2.25 * 10^{-22}t + 1$$

which are the approximate solution expanded for  $N = 3, 6, 10$  as  $y(t) = \cos(t)$

The absolute errors in the numerical solution of Example 4.4 are seen in Fig. 6. The error decreases when the integer N is increased.

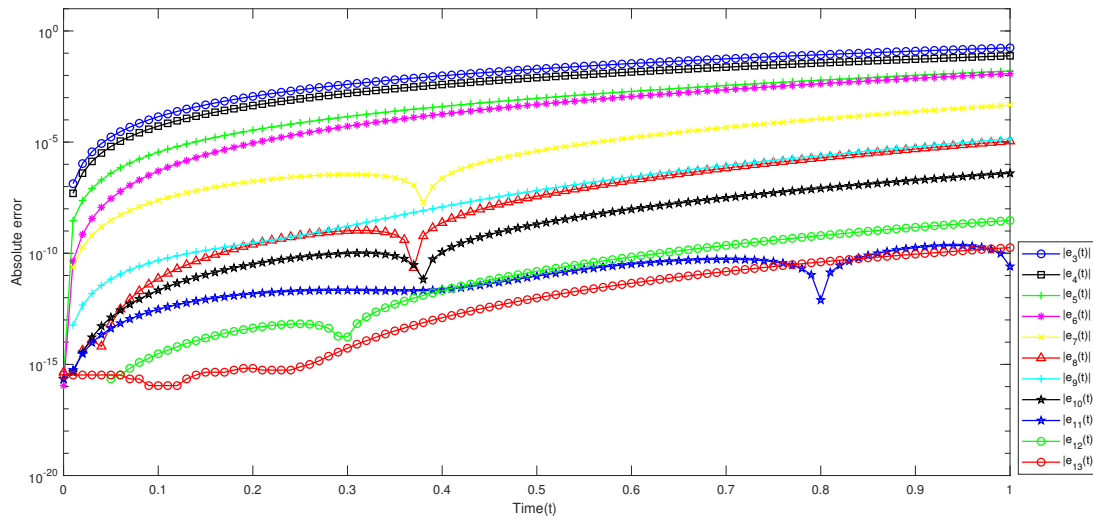


Fig. 6: The absolute errors of Example 4.4 for  $3 \leq N \leq 13$ .

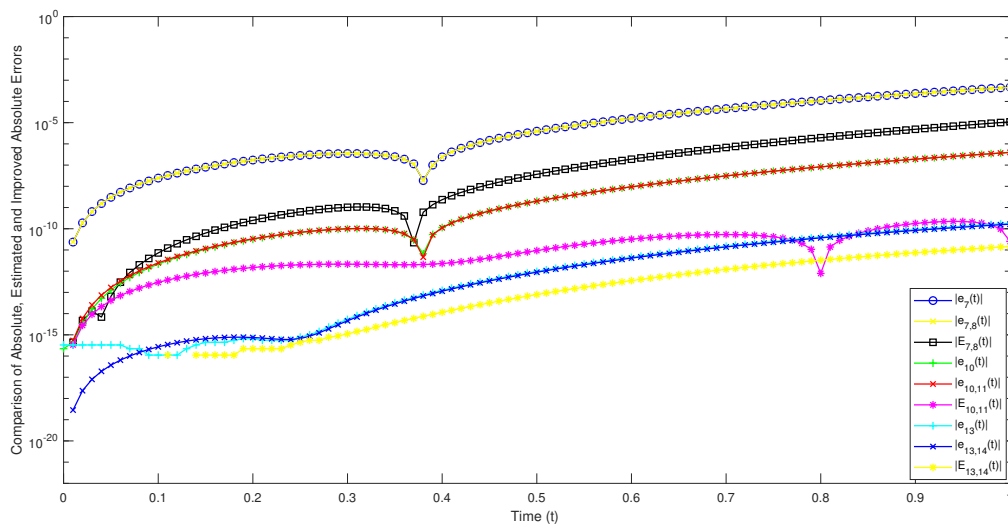


Fig. 7: Comparison of Absolute, Estimated and Improved Absolute Errors of Example 4.4.

### 5 Conclusion

A collocation calculation approach based on the Lucas polynomial is proposed in this study to solve the differential difference equations. Additionally, the control of the solutions is carried out using predetermined approaches. The residual error function also provides an estimation of the error. When results from the tables and figures are compared, it is clear that the new method is very efficient and practical. Utilizing the suggested method to assess the validity of various problem-solving approaches is another benefit. Tables and figures show that the errors drop more quickly as  $N$  increases, hence employing a large number of  $N$  is advised for better results.

## Competing interest

The authors declare that they have no competing interests.

## Authors' contribution

All of the authors contributed to the conception and development of this manuscript.

## References

- [1] Cooke, Kenneth L., *Differential-difference equations*, International symposium on nonlinear differential equations and nonlinear mechanics. Academic press, (1963).
- [2] BiÅşer K.E., Sezer M., *A computational method for solving differential equations with quadratic nonlinearity by using Bernoulli polynomials*, Thermal Science, 23-1, 275-283, doi:10.2298/TSCI181128041B, (2019).
- [3] Erdem, K., Yalcinbas, S., *Numerical approach of linear delay difference equations with variable coefficients in terms of Bernoulli polynomials*, In AIP conference proceedings, Vol. 1493, No. 1, pp. 338-344, American Institute of Physics, (2012).
- [4] Erdem, K., Yalcinbas, S., *Bernoulli polynomial approach to high-order linear differential-difference equations*, In AIP conference proceedings, Vol. 1479, No. 1, pp. 360-364, American Institute of Physics, (2012).
- [5] Yuzbasi, S., and N. Sahin, *Numerical Solution of a Class of Lane-Emden Type Differential-difference Equations*, Journal of Advanced Research in Scientific Computing 3(3) pp. 14-29, (2011).
- [6] Kurkcü, Omur Kivanc, Ersin Aslan, and Sezer, M., *A Numerical Method for Solving Functional Differential Equations with Delays*, International Students Science Congress, 1, 72-77, (2018).
- [7] Gumgum S., Baykus Savasneril N., Kurkcü O.K., Sezer M., *A numerical technique based on Lucas polynomials together with standard and Chebyshev-Lobatto collocation points for solving functional integro-differential equations involving variable delays*, Sakarya University Journal of Science, vol. 22.6, 1659-1668 doi:10.16984/saufenbilder.384592, (2018).
- [8] Baykus Savasneril N., Sezer M., *Hybrid Taylor-Lucas Collocation Method for Numerical Solution of High-Order Pantograph Type Delay Differential Equations with Variable Delays*, Appl. Math. Inf. Sci. 11, No. 6, 1795-1801, doi:10.18576/amis/110627, (2017).
- [9] Gumgum, S., Savasneril, N. B., Kurkcü, O. K., Sezer, M., *Lucas polynomial solution of nonlinear differential equations with variable delay*, Hacettepe Journal of Mathematics and Statistics, 1-12, (2019).
- [10] Kurkcü, O.K., Aslan, E., Sezer M., *A numerical method for solving some model problems arising in science and convergence analysis based on residual function*, Applied Numerical Mathematics, 121, 134-148, doi:10.1016/j.apnum.2017.06.015, (2017).
- [11] Kurkcü, O. K., Aslan, E. and Sezer, M., *A numerical approach with error estimation to solve general integro-differential-difference equations using Dickson polynomials*, Applied Mathematics and Computation, 276, 324-339, (2016).
- [12] Mollaoglu, T., *Volterra tipi gecikmeli fonksiyonel integro-diferansiyel denklemler için Gegenbauer polinom yaklaşımı*, Master thesis, Manisa Celal Bayar University, Manisa, (2017).
- [13] Gurbuz, B., Gulsu, M. and Sezer, M., *Numerical solution of differential-difference equations by Laguerre collocation method*, Erciyes universitesi Fen Bilimleri Enstitüsü Dergisi, 27(1): 75-87, (2011).