ORIGINAL RESEARCH

A numerical method with a control parameter for integro-differential delay equations with state-dependent bounds via generalized Mott polynomial

Ömür Kıvanç Kürkçü[1] ●

Abstract

In this paper, we introduce a numerical method to obtain an accurate approximate solution of the integro-differential delay equations with state-dependent bounds. The method is based basically on the generalized Mott polynomial with the parameter-β, Chebyshev–Lobatto collocation points and matrix structures. These matrices are gathered under a unique matrix equation and then solved algebraically, which produce the desired solution. We discuss the behavior of the solutions, controlling their parameterized form via β and so we monitor the effectiveness of the method. We improve the obtained solutions by employing the Mott-residual error estimation. In addition to comparing the results in tables, we also illustrate the solutions in figures, which are made up of the phase plane, logarithmic and standard scales. All results indicate that the present method is simple-structured, reliable and straightforward to write a computer program module on any mathematical software.

Keywords Collocation points · Error estimation · Matrix method · Mott polynomial · Delay

Mathematics Subject Classification 65L60 · 45J05

Introduction

Integro-differential equations (IDEs) and their delay types (IDDEs) govern many physical phenomena emerging in mathematics, mechanics, engineering, biology, economics, electrodynamics and oscillating magnetic field [1–8]. These varieties encourage many researchers in all over the world to give more attention than ever before. Because the sophisticated phenomena can be easily described with the aid of IDDEs. As these phenomena are evolved, finding the physical behavior of IDDEs becomes far more difficult. For example, state-dependent Riccati equation modeling vehicle state estimation [9], state-dependent delay Volterra equations considered in viscoelasticity theory [10] and the system of state-dependent delay differential equation describing forest growth [11] can be found in the literature. Thus, a difficult task appears while interpreting the physical responses of these

complex structures, analytically. Therefore, several numerical methods have recently been focused and established more on IDEs and their various types. To this end, Kürkçü et al. [12, 13] have solved IDEs and IDEs of difference type by means of Dickson matrix-collocation method. Reutskiy [14] has utilized the backward substitution method for solving the neutral Volterra–Fredholm IDEs. Chen ve Wang [15] have dealt with the neutral functional–differential equation with proportional delays using the variational iteration method. Bellen and Zennaro [4] have investigated the convergence and numerical solution of state-dependent delay differential equations. Gökmen et al. [16] have proposed Taylor polynomial method for solving the Volterra-type functional integral equations. Gülsu et al. [17] have used Chebyshev polynomial for delay differential equations. Savaşaneril and Sezer [18, 19] have employed Taylor and Taylor–Lucas polynomial method for searching the solution of Fredholm IDEs and pantograph-type delay differential equations, respectively. Maleknejad and Mahmoidi [20] have obtained the Taylor and block–pulse numerical solutions of Fredholm integral equation. Rohaninasab et al. [21] have established Legendre collocation method to solve Volterra–Fredholm IDEs. Yüzbaşı [22] have approached the numerical solutions of pantograph-type Volterra IDES with

✉ Ömür Kıvanç Kürkçü
  omurkivanc@outlook.com

1   Department of Mathematics, Izmir University of Economics,
    P. O. Box 35330, Izmir, Turkey

the aid of Laguerre polynomial. Gümgüm at al. [23] have obtained the Lucas polynomial solutions of functional IDEs involving variable delays.

All above studies motivate us to develop a numerical method and deal with the highly stiff problems, such as the integro-differential delay equations with state-dependent bounds in this paper. In the literature, there is no study maintaining the numerical solution of such equations. By way of this study, we can investigate their physical responses numerically, evaluating the obtained values in tables and figures. This paper is organized as follows: "Fundamental properties of Mott polynomial" section mentions some properties of the Mott polynomials. "Constructing method of solution via matrix relations" section establishes new matrix relations and method solution. "Mott-residual error estimation" section gives Mott-residual error estimation as an algorithmic sense. "Numerical examples" section includes stiff numerical examples solved with the aid of the present method. "Conclusions" section presents the discussions about the present method and its efficiency by taking into account the results in "Numerical examples" section. The functional integro-differential delay equations with state-dependent bounds are of the form

$$\sum_{r=0}^{m_1} P_r(t) y^{(r)}\big(t - \sigma_r\big)$$

$$= g(t) + \sum_{q=0}^{m_2} \lambda_q \int_{c_q y(t)}^{d_q y(t)} K_q(t, s) y\big(s - \tau_q\big) \mathrm{d}s,$$

$$a \le t, s \le b, \tag{1}$$

subject to the initial conditions

$$\sum_{k=0}^{m_1-1} y^{(k)}(a) = \psi_k, \tag{2}$$

where $y(t)$, $P_r(t)$, $g(t)$ and $K_q(t, s)$ are analytic functions on $[a, b]$; σ_r and τ_q are real constant delays $\big(\sigma_r, \tau_q \ge 0\big)$; λ_q, c_q, $d_q\big(c_q < d_q\big)$ and ψ_k are proper constants.

Our aim in this study is to efficiently obtain an accurate approximate solution of Eq. (1) by developing the Mott matrix-collocation method, which was previously introduced in [24]. Besides, the parameter-β in the generalized Mott polynomial is used as a control parameter in the numerical approximations. Hence, we can control the obtained solutions in terms of their more consistent structures. The approximate solution comes out to be in the form (see [24])

$$y(t) \cong y_N(t) = \sum_{n=0}^{N} y_n S_n(t, \beta), \tag{3}$$

where y_n, $n = 0, 1, \dots, N$ are unknown Mott coefficients to be calculated by the method and $S_n(t, \beta)$ is the generalized

Mott polynomial [25]. Chebyshev–Lobatto collocation points used in the matrix systems are defined to be (see [23])

$$t_i = \frac{a + b}{2} + \frac{a - b}{2} \cos\left(\frac{\pi i}{N}\right) \tag{4}$$

where $i = 0, 1, \dots, N$ and $a = t_0 < t_1 < \cdots < t_N = b$.

Fundamental properties of Mott polynomial

In this section, we briefly describe some fundamental properties of Mott polynomial, which is used as a basis of the matrix-collocation method. In 1932, Mott [26] originally introduced the polynomial while monitoring the roaming behaviors of electrons for a problem in the theory of electrons. After this exploration, Erdèlyi et al. [27] established the explicit formula of the polynomial $S_n(t)$ as follows:

$$S_n(t) = \left(-\frac{t}{2}\right)^n (n - 1)! \sum_{l=0}^{\lfloor n/2 \rfloor} \frac{t^{-2l}}{l!(n - l)!(n - 2l - 1)!}$$

$$= (n!)^{-1} \left(-\frac{t}{2}\right)^n {}_3F_0\left(-n, \frac{1}{2} - \frac{n}{2}, 1 - \frac{n}{2}; -4t^{-2}\right),$$

where ${}_3F_0$ is a generalized hypergeometric function.

In 1984, Roman [28] presented both an associated Sheffer sequence and a generating function for the polynomial as follows:

$$f(t) = \frac{-2t}{1 - t^2} \quad \text{and} \quad \sum_{k=0}^{\infty} \frac{S_k(t)}{k!} s^k = \exp\left(\frac{t\sqrt{1 - s^2} - t}{s}\right),$$

where $S_0(t) = 1$, $S_1(t) = -\frac{t}{2}$, $S_2(t) = \frac{t^2}{4}$, $S_3(t) = -\frac{3t}{4} - \frac{t^3}{8}$ and $S_4(t) = \frac{t^2}{2} + \frac{t^4}{16}$.

On the other hand, a triangle coefficient matrix of the polynomial can be found in A137378 of OEIS [29]. In 2014, Kruchinin [25] converted the polynomial to a generalized form with a parameter-β:

$$S_n(t, \beta) = \sum_{p=1}^{n} \sum_{q=0}^{p} (-1)^{p-q+(n+p)/2} \frac{n!\big(1 + (-1)^{n+p}\big)}{2p!} \binom{p}{q}$$

$$\binom{\beta q}{(n + p)/2} t^p, \quad n > 0,$$

where the Mott polynomial is obtained for $\beta = 0.5$. For further properties of the polynomial, the reader can refer to [25–28].

Constructing method of solution via matrix relations

In this section, the fundamental matrix relations are presented to construct method of solution. Let us first state the solution form (3) in the matrix relation [24]

$$y(t) = \boldsymbol{S}(t, \beta)\boldsymbol{Y} \text{ and } y^{(r)}(t) = \boldsymbol{S}^{(r)}(t, \beta)\boldsymbol{Y}, \tag{5}$$

where

$$\boldsymbol{S}(t, \beta) = \begin{bmatrix} S_0(t, \beta) & S_1(t, \beta) & \cdots & S_N(t, \beta) \end{bmatrix},$$
$$\boldsymbol{S}^{(r)}(t, \beta) = \begin{bmatrix} S_0^{(r)}(t, \beta) & S_1^{(r)}(t, \beta) & \cdots & S_N^{(r)}(t, \beta) \end{bmatrix} \text{ and }$$
$$\boldsymbol{Y} = \begin{bmatrix} y_0 & y_1 & \cdots & y_N \end{bmatrix}^{\mathrm{T}}.$$

Now, inserting $t \to t - \sigma_r$ into the matrix relation (5), then we get

$$y^{(r)}(t - \sigma_r) = \boldsymbol{S}^{(r)}(t - \sigma_r, \beta)\boldsymbol{Y}, \tag{6}$$

where

$$\boldsymbol{S}^{(r)}(t - \sigma_r, \beta) = \begin{bmatrix} S_0^{(r)}(t - \sigma_r, \beta) & S_1^{(r)}(t - \sigma_r, \beta) & \cdots & S_N^{(r)}(t - \sigma_r, \beta) \end{bmatrix}.$$

Similarly, it holds that

$$y(s - \tau_q) = \boldsymbol{S}(s - \tau_q, \beta)\boldsymbol{Y}, \tag{7}$$

By the matrix relation (6), the left hand side of Eq. (1) is of the matrix relation form

$$\sum_{r=0}^{m_1} P_r(t)y^{(r)}(t - \sigma_r) = \sum_{r=0}^{m_1} \boldsymbol{P}_r(t)\boldsymbol{S}^{(r)}(t - \sigma_r, \beta)\boldsymbol{Y}. \tag{8}$$

Now, the matrix relation of integral part of Eq. (1) is given. First, the kernel function $K_q(t, s)$ can be written in the truncated Taylor series form [16, 18],

$$K_q(t, s) = \sum_{m=0}^{N} \sum_{n=0}^{N} k_{mn} t^m s^n \Rightarrow K_q(t, s) = \boldsymbol{X}(t)\boldsymbol{K}_q\boldsymbol{X}^{\mathrm{T}}(s), \tag{9}$$

where

$$\boldsymbol{K}_q = \begin{bmatrix} k_{mn}^q \end{bmatrix}, \ k_{mn}^q = \frac{1}{i!j!}\frac{\partial^{i+j}K_q(0,0)}{\partial t^i \partial s^j}, \ i,j = 0, 1, \ldots, N,$$

and

$$\boldsymbol{X}(t) = \begin{bmatrix} 1 & t & \cdots & t^N \end{bmatrix}.$$

Then, it holds from the matrix relations (7) and (9) that

$$\sum_{q=0}^{m_2} \lambda_q \int_{c_q y(t)}^{d_q y(t)} K_q(t, s)y(s - \tau_q)\mathrm{d}s$$
$$= \sum_{q=0}^{m_2} \lambda_q \boldsymbol{X}(t)\boldsymbol{K}_q\boldsymbol{R}_q(t)\boldsymbol{Y}, \tag{10}$$

where

$$\boldsymbol{R}_q(t) = \int_{c_q y(t)}^{d_q y(t)} \boldsymbol{X}^{\mathrm{T}}(s)\boldsymbol{S}(s - \tau_q, \beta)\mathrm{d}s = \begin{bmatrix} r_{ij}^q(t) \end{bmatrix},$$
$$i, j = 0, 1, \ldots, N.$$

Recalling the matrix relations (8) and (10) and collocation points (4), we thus write the combined matrix relation as

$$\sum_{r=0}^{m_1} \boldsymbol{P}_r(t_i)\boldsymbol{S}^{(r)}(t_i - \sigma_r, \beta)\boldsymbol{Y}$$
$$= g(t_i) + \sum_{q=0}^{m_2} \lambda_q \boldsymbol{X}(t_i)\boldsymbol{K}_q\boldsymbol{R}_q(t_i)\boldsymbol{Y}. \tag{11}$$

More briefly, we can construct the matrix relation (11) as the fundamental matrix equation

$$\left(\sum_{r=0}^{m_1} \boldsymbol{P}_r\boldsymbol{S}^{(r)}(\beta) - \sum_{q=0}^{m_2} \lambda_q \overline{\boldsymbol{X}}\ \overline{\boldsymbol{K}_q}\ \overline{\boldsymbol{R}_q} \right)\boldsymbol{Y} = \boldsymbol{G}, \tag{12}$$

where

$$\boldsymbol{S}^{(r)}(\beta) = \begin{bmatrix} \boldsymbol{S}^{(r)}(t_0 - \sigma_r, \beta) \\ \boldsymbol{S}^{(r)}(t_1 - \sigma_r, \beta) \\ \vdots \\ \boldsymbol{S}^{(r)}(t_N - \sigma_r, \beta) \end{bmatrix}$$
$$= \begin{bmatrix} S_0^{(r)}(t_0 - \sigma_r, \beta) & S_1^{(r)}(t_0 - \sigma_r, \beta) & \cdots & S_N^{(r)}(t_0 - \sigma_r, \beta) \\ S_0^{(r)}(t_1 - \sigma_r, \beta) & S_1^{(r)}(t_1 - \sigma_r, \beta) & \cdots & S_N^{(r)}(t_1 - \sigma_r, \beta) \\ \vdots & \vdots & \ddots & \vdots \\ S_0^{(r)}(t_N - \sigma_r, \beta) & S_1^{(r)}(t_N - \sigma_r, \beta) & \cdots & S_N^{(r)}(t_N - \sigma_r, \beta) \end{bmatrix},$$
$$\overline{\boldsymbol{X}} = \begin{bmatrix} \boldsymbol{X}(t_0) & 0 & \cdots & 0 \\ 0 & \boldsymbol{X}(t_1) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \boldsymbol{X}(t_N) \end{bmatrix}_{(N+1)\times(N+1)^2},$$
$$\boldsymbol{G} = \begin{bmatrix} g(t_0) & g(t_1) & \cdots & g(t_N) \end{bmatrix}^{\mathrm{T}},$$
$$\overline{\boldsymbol{K}_q} = \begin{bmatrix} \boldsymbol{K}_q & 0 & \cdots & 0 \\ 0 & \boldsymbol{K}_q & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \boldsymbol{K}_q \end{bmatrix}_{(N+1)^2\times(N+1)^2},$$
$$\overline{\boldsymbol{R}_q} = \begin{bmatrix} \boldsymbol{R}_q(t_0) \\ \boldsymbol{R}_q(t_1) \\ \vdots \\ \boldsymbol{R}_q(t_N) \end{bmatrix}_{(N+1)^2\times(N+1)}.$$

Using the matrix relation (5), we similarly state the matrix relation of the initial conditions (2) as the following:

$$\sum_{k=0}^{m_1-1} \boldsymbol{S}^{(k)}(a, \beta)\boldsymbol{Y} = \psi_k. \tag{13}$$

By the matrix equation (12), we are now ready to constitute the method of solution

$$\underbrace{\left(\sum_{r=0}^{m_1} P_r S^{(r)}(\beta) - \sum_{q=0}^{m_2} \lambda_q \overline{X}\ \overline{K_q}\ \overline{R_q}\right)}_{W} Y = G. \quad (14)$$

Then, it follows that

$$WY = G \ \text{ or } \ [W; G].$$

On the other hand, we can construct the matrix relation of Eq. (13) as

$$U_k Y = \psi_k \Rightarrow [U_k\ ; \psi_k],\ k = 0, 1, \dots m_1 - 1, \quad (15)$$

where

$$U_k \equiv \begin{bmatrix} u_{k0} & u_{k1} & \cdots & u_{kN} \end{bmatrix}.$$

Replacing the row(s) of the matrix relation (15) by the last m_1 row(s) in W, we then obtain the augmented matrix

$$\left[\widetilde{W}\ ;\ \widetilde{G}\right] = \begin{bmatrix} w_{00} & w_{01} & \cdots & w_{0N} & ; & g(t_0) \\ w_{10} & w_{11} & \cdots & w_{1N} & ; & g(t_1) \\ \vdots & \vdots & \vdots & \vdots & :; & \vdots \\ w_{N-m_1,0} & w_{N-m_1,1} & \cdots & w_{N-m_1,N} & ; & g(t_{N-m_1}) \\ u_{00} & u_{01} & \cdots & u_{0N} & ; & \psi_0 \\ u_{10} & u_{11} & \cdots & u_{1N} & ; & \psi_1 \\ \vdots & \vdots & \vdots & \vdots & :; & \vdots \\ u_{m_1-1,0} & u_{m_1-1,1} & \cdots & u_{m_1-1,N} & ; & \psi_{m_1-1} \end{bmatrix}. \quad (16)$$

We solve the augmented matrix (16) only if $\text{rank}\widetilde{W} = \text{rank}\left[\widetilde{W}\ ;\ \widetilde{G}\right] = N + 1$. We can state $Y = \left(\widetilde{W}\right)^{-1}\widetilde{G}$. Thus, the Mott coefficients appearing in the form (3) are obtained, and then, they are substituted into the form (3); we finally reach the Mott polynomial solution with the parameter-β.

Mott-residual error estimation

The residual error analysis has successfully been employed in [7, 8, 12, 13, 16, 22, 30]. For this motivation, we introduce the Mott-residual error estimation technique including the Mott polynomial and a residual function to improve the Mott polynomial solution (3) of Eq. (1). Algorithmic procedure of this technique can be described for the present method as

Step 1: $R_N(t) \leftarrow \sum_{r=0}^{m_1} P_r(t) y_N^{(r)}\left(t - \sigma_r\right)$
$\qquad - \sum_{q=0}^{m_2} \lambda_q \int_{c_q y_N(t)}^{d_q y_N(t)} K_q(t, s) y_N\left(s - \tau_q\right) \mathrm{d}s - g(t),$

Step 2: $e_N(t) \leftarrow y(t) - y_N(t) = -R_N(t),$
Step 3: $0 \leftarrow \sum_{k=0}^{m_1-1} e_N^{(k)}(a),$
Step 4: Solve the error problem consisting of Steps 2 and 3,
Step 5: $e_{N,M}(t) \leftarrow \sum_{n=0}^{M} y_n^* S_n(t, \beta),\ (M > N),$ where $S_n(t, \beta)$ is the Mott polynomial and $e_{N,M}(t)$ is a Mott-estimated error function,
Step 6: $y_{N,M}(t) \leftarrow y_N(t) + e_{N,M}(t),$ where $y_{N,M}(t)$ is a corrected Mott polynomial solution.

Thus, we improve the Mott polynomial solution and it is worth specifying that the corrected error function is of the form $E_{N,M}(t) = y(t) - y_{N,M}(t).$

Numerical examples

In this section, we apply the present method to solve some stiff problems concerned with Eq. (1). To do this, we develop a computer program routine on Mathematica 11. The obtained solutions and numerical values are elucidated in figures and tables.

Example 1 Consider the second-order FIDE with state-dependent bounds and multi-delays

$$y''(t) - ty'(t - 1) + t^2 y(t - 0.6)$$
$$= g(t) - \int_{2(t^2+1)}^{3(t^2+1)} (t^2 + s^2) y(s - 0.5)\mathrm{d}s$$
$$+ \int_{0}^{t^2+1} (t^2 s^2) y(s - 0.1)\mathrm{d}s$$

subject to the initial conditions $y(0) = 1,\ y'(0) = 0,$ and $t, s \in [0, 1].$ Here, the constant delays are $\left\{\left\{\sigma_0 = 0.6,\ \sigma_1 = 1\right\}, \left\{\tau_0 = 0.5,\ \tau_1 = 0.1\right\}\right\}$ and

$$g(t) = -31.8667 + 2t - 174.987t^2 - 1.2t^3 - 360.69t^4$$
$$- 378.707t^6 - 198.947t^8 - 41.25t^{10} + t^{12}/5.$$

By the fundamental matrix equation (14), we construct the fundamental matrix equation as

$$\left(P_0 S^{(0)}(\beta) + P_1 S^{(1)}(\beta) + P_2 S^{(2)}(\beta)\right.$$
$$\left. + \lambda_0 \overline{X}\ \overline{K_0}\ \overline{R_0} - \lambda_1 \overline{X}\ \overline{K_1}\ \overline{R_1}\right) Y = G,$$

After applying the described procedure to the equation above, we easily get the augmented matrix

$$\left[\tilde{W} ; \tilde{G}\right] = \begin{bmatrix} -6.33 & 13.0833\beta & -25.533\beta^2 & ; & -31.87 \\ 1 & 0 & 0 & ; & 1 \\ 0 & -\beta & 0 & ; & 0 \end{bmatrix}.$$

Solving this matrix system, we get

$$Y = \begin{bmatrix} 1 & 0 & 1/\beta^2 \end{bmatrix}^{\mathrm{T}}.$$

and it holds from Eq. (3) that

$$y(t) = t^2 + 1,$$

which is the exact solution.

Example 2 Consider the fourth-order IDDE with state-dependent bounds and variable coefficients

$$y^{(iv)}(t) + \sin(t)y''(t - \sigma_2) - \cos(t)y(t) = g(t)$$
$$+ \int_{\cos(t)}^{2\cos(t)} \exp(t + s)y(s - \tau_0)\mathrm{d}s,$$

subject to the initial conditions $y(0) = 1$, $y'(0) = 0$, $y''(0) = -1$, $y'''(0) = 0$, and $t, s \in [0, L]$. Here, the exact solution is $y(t) = \cos(t)$ and for $\sigma_2 = \tau_0 = 0$,

$$g(t) = \cos(t)[1 - \cos(t) - \sin(t)]$$
$$+ 2.71828^{(t+\cos(t))}[0.5\cos(\cos(t))$$
$$+ 0.5\sin(\cos(t)) + 2.71828^{\cos(t)}(-0.5\cos(2\cos(t))$$
$$- 0.5\sin(2\cos(t)))].$$

Similarly, $g(t)$ can be calculated via the exact solution for various values of $\{\sigma_2, \tau_0\}$.

Taking different truncation limit N and $L = \{1, 10\}$, we solve the problem by using both the present method and Taylor collocation method [16, 18] to compare the obtained results. We later employ the Mott-residual error estimation to improve the solution. It is important to state that we investigate the effects of β and the delays on the Mott polynomial solutions. Therefore, the following discussion is made:

- Table 1 shows the absolute errors for fixed $\sigma_2 = \tau_0 = 0.01$ and $\beta = 1.5$. Also in there, the better numerical results are obtained in comparison with Taylor collocation method [16, 18].
- When $L = \{1, 10\}$, the oscillatory behaviors of the solutions coincide properly with the exact solution in Figs. 1 and 2 , respectively.
- L_∞ errors obtained with $N = 12$, $\beta = 1.5$ are investigated with respect to different delays σ_2 and τ_0 in Table 2. The best approximation stands for $8.92e-10$ when $\sigma_2 = \tau_0 = 1$.
- Similarly, the behavior of L_∞ errors obtained with the fixed $N = 12$ and $\sigma_2 = \tau_0 = 0.5$ is demonstrated with

respect to β in the logarithmic scaled plot shown in Fig. 3.

Example 3 Consider the second-order external forced oscillatory differential equation exposing to single time-delayed effect

$$y''(t) + 2\varepsilon y'(t - \sigma_1) + y(t) = F\cos(\omega t), \ t \in [0, L],$$

subject to the initial conditions $y(0) = 0$ and $y'(0) = 1$. A under-damped parameter $|\varepsilon| < 1$, an external force F, a non-resonance excitation $\omega \neq 1$ [5]. Here, $L = \{1, 15\}$ and also the exact solution of the problem is unknown, but it can be approached numerically with the aid of Mathematica as

NDSolve$[y''[t] + 2\varepsilon y'[t - \sigma_1]$
$+ y[t] == 2\mathrm{Cos}[2t], y[0] == 0, y'[0] == 1,$
$y[t], \{t, 0, L\}][[1, 1, 2]].$

Previously, Kalmar–Nagy and Balachandran [5] have studied the linear oscillator differential equation with external forcing, under-damped system and non-resonance excitation. They have determined the steady-state response and magnification factor. In this example, by exposing the linear oscillator equation [5] to the delayed effect σ_1, let us seek the numerical solutions for different N, M, σ_1, ε, β, and the fixed $F = \omega = 2$. Thus,

- By increasing N and M, we demonstrate the absolute errors for $\sigma_1 = 0$ and $\beta = 1$ in Table 3. This indicates that N and M enable us to enhance the accuracy of the method.
- Figs. 4 and 5 illustrate the oscillatory response of both the Mott polynomial $y_{25}(t)$ and Mathematica solutions for $L = 15$, $\sigma_1 = 0, 0.5$ and $\varepsilon = \{0.1, 0.45\}$. Figures 6 and 7 also illustrate these solutions in the phase plane.
- The decreasing L_∞ error diagram obtained with $N = 12$ is demonstrated with respect to the control parameter β in Fig. 8.

In addition, we draw the attention to the fact that both ε and σ_1 have a different effect on the Mott polynomial solution.

Table 1 Comparison of the absolute errors of Example 2 with $\sigma_2 = \tau_0 = 0.01$ for $\beta = 1.5$

t_i	$|e_5(t_i)|$	$|e_9(t_i)|$	$|E_{9,11}(t_i)|$	$|e_9(t_i)|$ [16, 18]
0.2	1.05e−05	9.16e−09	5.71e−11	1.91e−05
0.4	1.70e−04	1.50e−07	9.26e−10	3.08e−04
0.6	8.63e−04	7.66e−07	4.90e−09	1.54e−03
0.8	2.68e−03	2.40e−06	1.73e−08	4.75e−03
1.0	6.34e−03	5.71e−06	4.80e−08	1.11e−02

Table 2 Comparison of L_∞ errors ($N = 12$, $\beta = 1.5$) with respect to the delays σ_2 and τ_0 for Example 2

σ_2	− 0.5	0	0.1	0.5	0.8	**1**	2
τ_0	− 0.5	0	0.1	0.5	0.8	**1**	2
L_∞	3.02e−09	9.00e−09	1.32e−08	2.59e−09	6.88e−09	**8.92e−10**	6.92e−07

Bold values show that the method has provided the best approximation when the specific values

Fig. 1 Comparison of the Mott polynomial with the control parameter $\beta = 1.5$ and exact solutions in terms of N on [0, 1] for Example 2 with $\sigma_2 = \tau_0 = 0.5$
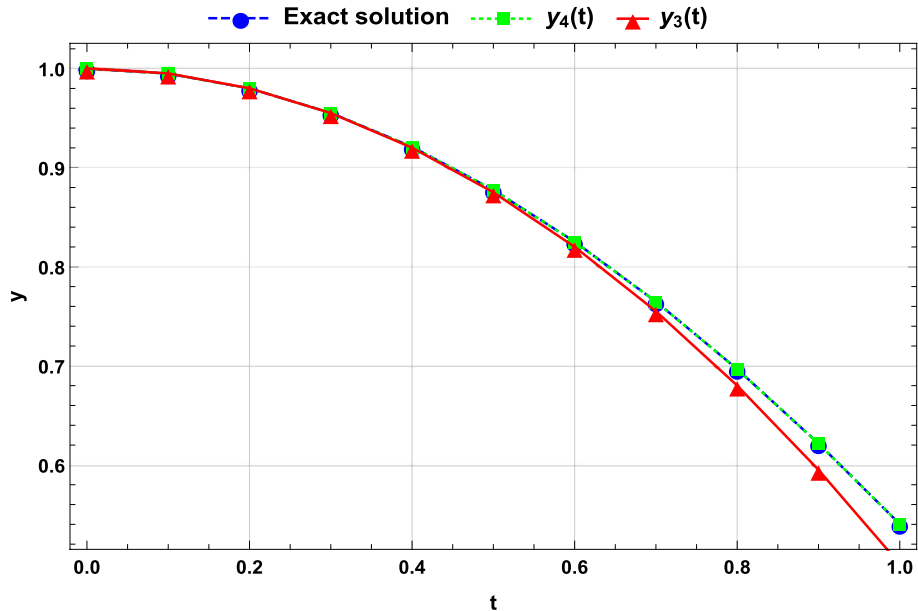
Fig. 2 Oscillatory behavior of the Mott polynomial ($\beta = 1.5$) and exact solutions on [0, 10] for Example 2 with $\sigma_2 = \tau_0 = 0.5$
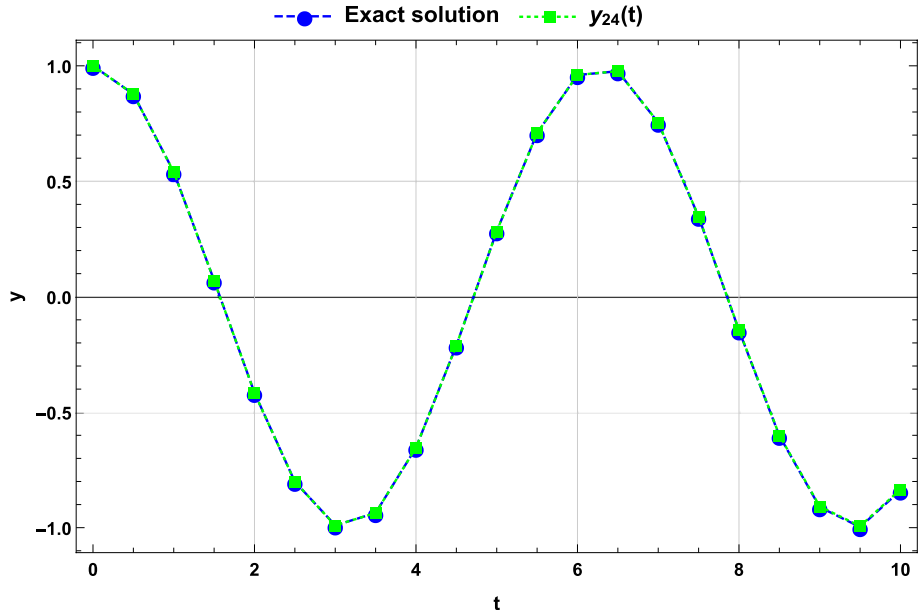
Conclusions

An efficient numerical method based on the generalized Mott polynomial, Chebyshev–Lobatto collocation points and the matrix structures has been proposed to solve stiff IDDEs with state-dependent bounds, which are introduced for the first time with this paper. Thanks to the simplicity of the present method, the obtained solutions have been accurately approximated to the exact and Mathematica solutions. Controlling the optimum value of the parameter-β in the

Fig. 3 Logarithmic scaled plot of L_∞ error with respect to β for Example 2 with $L = 1$ and $\sigma_2 = \tau_0 = 0.5$
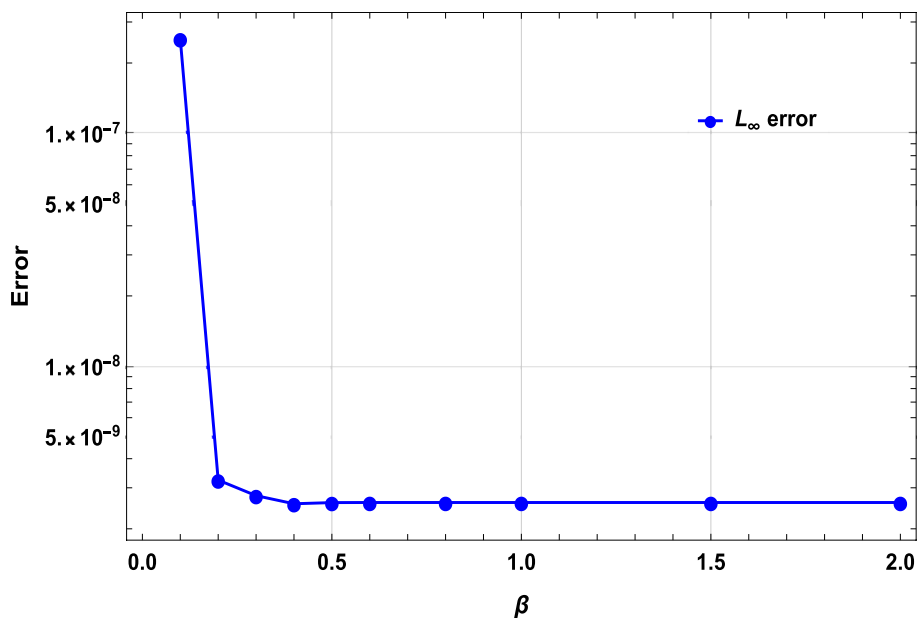

Table 3 Comparison of the absolute errors between the Mott polynomial and Mathematica solutions ($\beta = 1$) for Example 3 with $\varepsilon = 0.1$ and $\sigma_1 = 0$

t_i	$|e_6(t_i)|$	$|e_8(t_i)|$	$|E_{8,9}(t_i)|$	$|E_{8,12}(t_i)|$
0.2	3.18e−07	2.49e−09	5.85e−11	6.49e−15
0.4	1.08e−06	6.27e−09	9.60e−11	2.51e−14
0.6	4.88e−06	1.67e−08	3.92e−10	2.78e−14
0.8	1.37e−05	2.27e−08	1.49e−09	1.54e−13
1.0	2.11e−04	9.97e−07	3.46e−08	5.49e−12

solutions is of importance as can be seen in Figs. 3 and 8. Therefore, this parameter plays a specific role in numerical approximations. The Mott-residual error estimation has effectively improved the obtained solutions as seen from Tables 1 and 3 . The effects of the delays have been monitored differently. So, we here want to state from Table 2, Figs. 4 and 5 that the delays change the behavior of the problems in a physical sense. By investigating all results, as N is increased, the accuracy of the method increases. Thus, we conclude that the present method could be very applicable

Fig. 4 Oscillatory behavior of the Mott polynomial ($\beta = 1.5$) and exact solutions with respect to σ_1 in $[0, 15]$ for Example 3 with $\varepsilon = 0.1$, $F = 2$, and $\omega = 2$
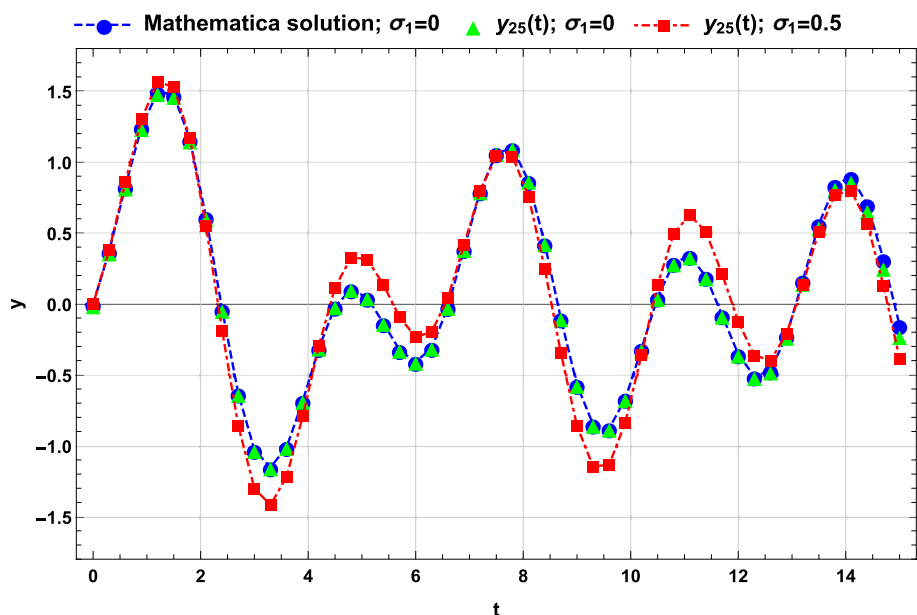
Fig. 5 Oscillatory behavior the Mott polynomial ($\beta = 1.5$) and exact solutions with respect to σ_1 in [0, 15] for Example 3 with $\varepsilon = 0.45$, $F = 2$ and $\omega = 2$
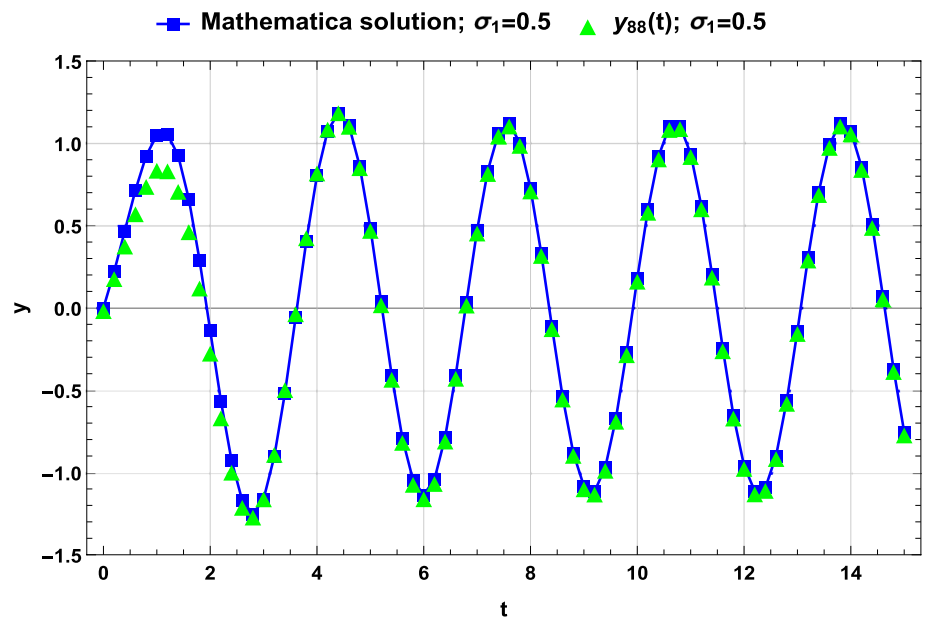

Fig. 6 Phase plane behavior of the Mott polynomial ($\beta = 1.5$) and exact solutions in phase plane for Example 3 with $\varepsilon = 0.1$, $F = 2$, $\omega = 2$ and $L = 15$
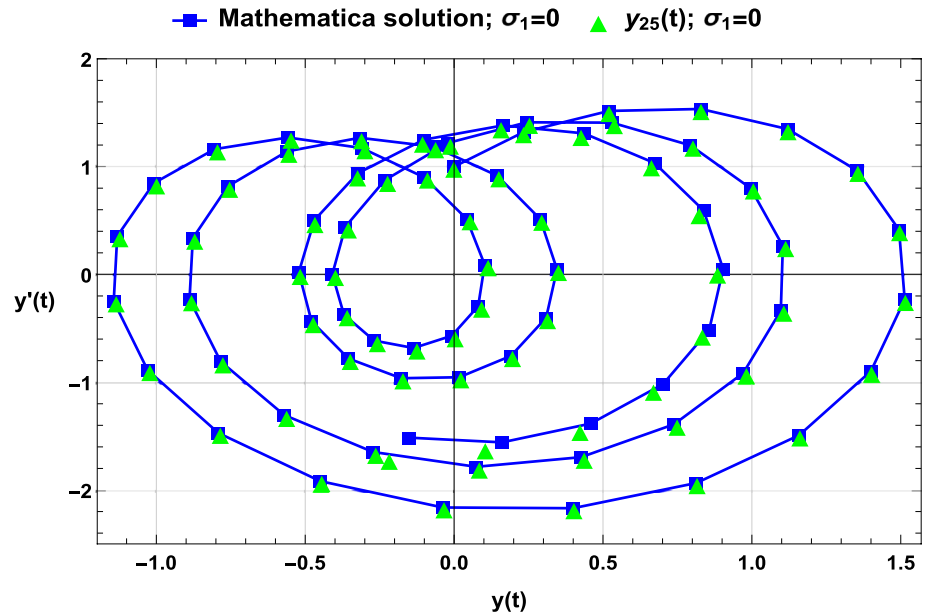
Fig. 7 Phase plane behavior of the Mott polynomial ($\beta = 1.5$) and exact solutions in phase plane for Example 3 with $\varepsilon = 0.45$, $F = 2$, $\omega = 2$ and $L = 15$
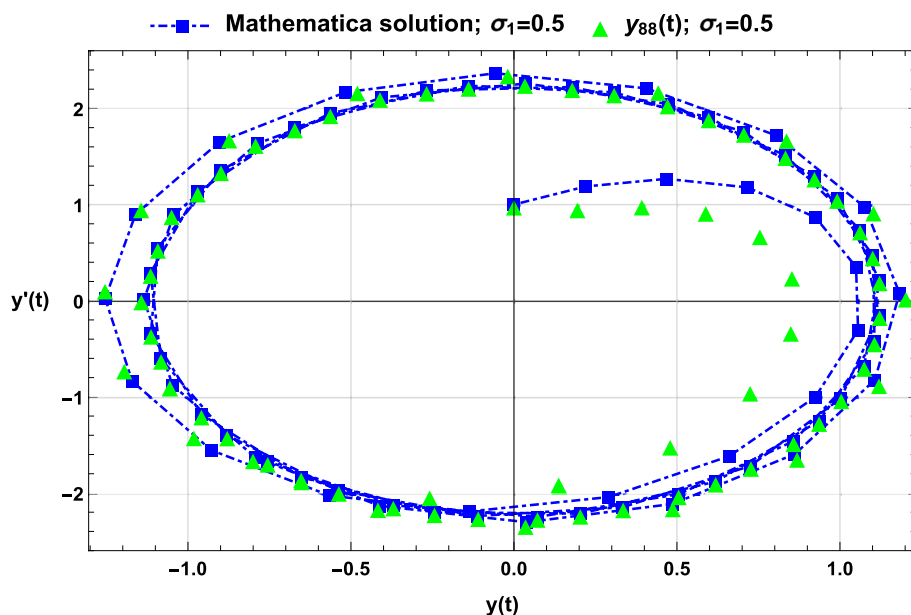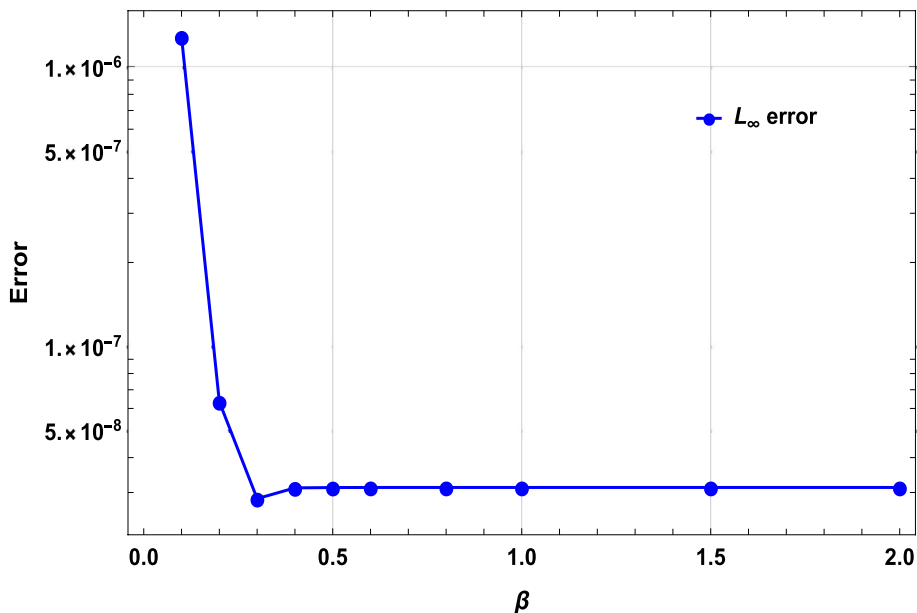
Fig. 8 Logarithmic scaled plot of L_∞ error with respect to β for Example 3 with $L = 1$, $\varepsilon = 0.1$, $\sigma_1 = 0$, $F = 2$ and $\omega = 2$

and reliable for solving other well-known phenomena, such as partial differential and fractional differential equations after making some required modifications on the proposed method.

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (http://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

References

1. Dehghan, M., Shakeri, F.: The use of the decomposition procedure of Adomian for solving a delay differential equation arising in electrodynamics. Phys. Scr. **78**, p. 11 (Article No. 065004) (2008)
2. Dehghan, M., Shakeri, F.: Solution of an integro-differential equation arising in oscillating magnetic field using He's homotopy perturbation method. Prog. Electromagn. Res. PIER **78**, 361–376 (2008)
3. Kot, M.: Elements of Mathematical Ecology. Cambridge University Press, Cambridge (2001)
4. Bellen, A., Zennaro, M.: Numerical Methods for Delay Differential Equations. Numerical Mathematics and Scientific

Computation. The Clarendon Press Oxford University Press, New York (2003)

5. Kalmar-Nagy, T., Balachandran, B.: Forced harmonic vibration of a Duffing oscillator with linear viscous damping. In: Kovacic, I., Brennan, M.J. (eds.) The Duffing Equation (2011). https://doi.org/10.1002/9780470977859.ch5

6. Kürkçü, Ö.K., Aslan, E., Sezer, M.: A numerical method for solving some model problems arising in science and convergence analysis based on residual function. Appl. Numer. Math. **121**, 134–148 (2017)

7. Yüzbaşı, Ş.: Improved Bessel collocation method for linear Volterra integro-differential equations with piecewise intervals and application of a Volterra population model. Appl. Math. Model. **40**, 5349–5363 (2016)

8. Yüzbaşı, Ş., Sezer, M., Kemanc, B.: Numerical solutions of integro-differential equations and application of a population model with an improved Legendre method. Appl. Math. Model. **37**, 2086–2101 (2013)

9. Hoek, R., Alirezaei, M., Schmeitz, A., Nijmeijer, H.: Vehicle state estimation using a state dependent Riccati equation. IFAC-PapersOnLine **50**, 3388–3393 (2017)

10. Andrade, B., Siracusa, G.: On evolutionary Volterra equations with state-dependent delay. Comput. Math. Appl. **75**, 1181–1190 (2018)

11. Magal, P., Zhang, Z.: A system of state-dependent delay differential equation modelling forest growth II: boundedness of solutions. Nonlinear Anal. Real World Appl. **42**, 334–352 (2018)

12. Kürkçü, Ö.K., Aslan, E., Sezer, M.: A numerical approach with error estimation to solve general integro-differential-difference equations using Dickson polynomials. Appl. Math. Comput. **276**, 324–339 (2016)

13. Kürkçü, Ö.K., Aslan, E., Sezer, M.: A novel collocation method based on residual error analysis for solving integro-differential equations using hybrid Dickson and Taylor polynomials. Sains Malays. **46**, 335–347 (2017)

14. Reutskiy, SYu.: The backward substitution method for multipoint problems with linear Volterra–Fredholm integro-differential equations of the neutral type. J. Comput. Appl. Math. **296**, 724–738 (2016)

15. Chen, X., Wang, L.: The variational iteration method for solving a neutral functional–differential equation with proportional delays. Comput. Math. Appl. **59**, 2696–2702 (2010)

16. Gökmen, E., Yuksel, G., Sezer, M.: A numerical approach for solving Volterra type functional integral equations with variable bounds and mixed delays. J. Comput. Appl. Math. **311**, 354–363 (2017)

17. Gülsu, M., Öztürk, Y., Sezer, M.: A new Chebyshev polynomial approximation for solving delay differential equations. J. Differ. Equ. Appl. **18**, 1043–1065 (2012)

18. Baykuş, N., Sezer, M.: Solution of high-order linear Fredholm integro-differential equations with piecewise intervals. Numer. Methods Partial Differ. Equ. **27**, 1327–1339 (2011)

19. Savaşaneril, N.B., Sezer, M.: Hybrid Taylor–Lucas collocation method for numerical solution of high-order pantograph type delay differential equations with variables delays. Appl. Math. Inf. Sci. **11**, 1795–1801 (2017)

20. Maleknejad, K., Mahmoidi, Y.: Numerical solution of linear Fredholm integral equation by using hybrid Taylor and block-pulse functions. Appl. Math. Comput. **149**, 799–806 (2004)

21. Rohaninasab, N., Maleknejad, K., Ezzati, R.: Numerical solution of high-order Volterra–Fredholm integro-differential equations by using Legendre collocation method. Appl. Math. Comput. **328**, 171–188 (2018)

22. Yüzbaşı, Ş.: Laguerre approach for solving pantograph-type Volterra integro-differential equations. Appl. Math. Comput. **232**, 1183–1199 (2014)

23. Gümgüm, S., Savaşaneril, N.B., Kürkçü, Ö.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 Univ. J. Sci. **22**, 1659–1668 (2018). https://doi.org/10.16984/saufenbilder.384592

24. Kürkçü, Ö.K.: A new numerical method for solving delay integral equations with variable bounds by using generalized Mott polynomials. Eskişehir Technical University. J. Sci. Tech. A Appl. Sci. Eng. 19, 844–857 (2018). https://doi.org/10.18038/aubtda.409056

25. Kruchinin, D.V.: Explicit formula for generalized Mott polynomials. Adv. Stud. Contemp. Math. **24**, 327–332 (2014)

26. Mott, N.F.: The polarisation of electrons by double scattering. Proc. R. Soc. Lond. A **135**, 429–458 (1932)

27. Erdèlyi, A., Magnus, W., Oberhettinger, F., Tricomi, F.G.: Higher Transcendental Functions, vol. III. McGraw-Hill Book Company Inc., New York (1955)

28. Roman, S.: The Umbral Calculus. Pure and Applied Mathematics, vol. 111. Academic Press Inc., London (1984)

29. Solane, N.J.: The on-line encyclopedia of integer sequences. https://oeis.org/A137378

30. Çelik, İ.: Collocation method and residual correction using Chebyshev series. Appl. Math. Comput. **174**, 910–920 (2006)