DEVELOPMENT OF THE TAU METHOD FOR THE NUMERICAL STUDY OF A FOURTH-ORDER PARABOLIC PARTIAL DIFFERENTIAL EQUATION

Babak SOLTANALIZADEH [1], Hadi Roohani GHEHSAREH[2], Saeid ABBASBANDY [3]

In this paper, an approximate method based on matrix formulated algorithm is presented for numerical study of a fourth-order parabolic partial differential equation. For the numerical section, shifted Standard and shifted Chebyshev bases are utilized. Several numerical examples are presented to confirm the efficiency and accuracy of this procedure.

Keywords: fourth-order equation; parabolic equation; matrix formulation method; orthogonal polynomials.

1. Introduction

An operational technique for the numerical solution of nonlinear ordinary differential equations based on the Tau method [1] is presented by Ortiz and Samara [2]. Afterwards, many authors have been used this method and some other similar methods for solving various types of equations. In [3], this method is used for linear ordinary differential eigenvalue problems and in [4] it is used for partial differential equations. This method has been developed for different types of integral and integro-differential equations [7, 8]. Authors of [9] used this method for the system of nonlinear Volterra integro-differential equations. Some matrix formulation techniques with arbitrary polynomial bases [5] and shifted Standard and shifted Chebyshev bases [6], have been proposed for the numerical solutions of the heat and wave equations with nonlocal boundary conditions. Similar works can be found in [10, 11, 12, 13, 14].

In this paper, we focus on the following parabolic equation

$$\frac{\partial^2 u}{\partial t^2} - \alpha \frac{\partial^4 u}{\partial x^4} = f(x,t), \qquad (1)$$

with the following initial conditions

$$u(x,0) = r(x), red\, 0 \le x \le l, \qquad (2)$$

---

[1] Young Researchers Club and Elites, Ardabil Branch, Islamic Azad University, Ardabil, Iran, e-mail: babak.soltanalizadeh@gmail.com.
[2] Prof., Dept.of Mathematics, Imam Khomeini International University, Qazvin, Iran.
[3] Ph.D student., Dept.of Mathematics, Imam Khomeini International University, Qazvin, Iran.

$$u_t(x,0) = s(x), red 0 \le x \le l, \quad (3)$$

and the boundary conditions

$$u(0,t) = p(t), 0 < t \le T, \quad (4)$$

$$u(1,t) = g(t), 0 < t \le T, \quad (5)$$

$$u_{xx}(0,t) = k(t), 0 < t \le T, \quad (6)$$

$$u_{xx}(1,t) = q(t), 0 < t \le T, \quad (7)$$

where u is the transverse displacement of the beam, t is time and x is distance variable and $f(x,t)$ is dynamic driving force per unit mass. the functions $f(x,t)$, $r(x), s(x), p(t), g(t), k(t)$ and $q(t)$ and the constants α are known.

Problem (1)-(7) is the problem of undamped transverse vibrations of a flexible straight beam in such a way that it's supports do not contribute to the strain energy of the system. Recently, various authors focused on the development of numerical techniques for the solution of the Eq. (1) [15, 16, 17, 18, 19]. Aziz et. al [15] presented a three level method based on parametric quintic spline in space and finite difference discretization in time. In [16], a fifth-degree B-spline scheme is proposed for the numerical solution of the problem (1)-(7). Authors of [17, 18] proposed some finite difference schemes. Wazwaz [19], solved Eq. (1) by using the Adomian decomposition method.

2. Formulation of the method.

We assume that the functions $f(x,t)$, $r(x), s(x), p(t), g(t), k(t)$ and $q(t)$ generally are polynomials. otherwise, we can approximate these functions by one or two variate Taylor and Chebyshev series. Let $\upsilon = [\upsilon_0(x), \upsilon_1(x), \upsilon_2(x), \ldots, \upsilon_n(x)]^T$ is a polynomial basis vector given by $\upsilon = VX$ and $\omega = [\omega_0(t), \omega_1(t), \omega_2(t), \ldots, \omega_m(t)]^T$ is a polynomial basis vector given by $\omega = WT$, where V and W are nonsingular lower triangular matrices and $X = [1, x, x^2, ..., x^n]^T, T = [1, t, t^2, ..., x^m]^T$. So by using above polynomial basis vectors, we get

$$\begin{cases} f(x,t); \sum_{i=0}^{n}\sum_{j=0}^{m} f_{ij}\upsilon_i(x)\omega_j(t) = \upsilon^T F\omega, \\ r(x); \sum_{i=0}^{n} r_i\upsilon_i(x) = \upsilon^T R,\ \ s(x); \sum_{i=0}^{n} s_i\upsilon_i(x) = \upsilon^T S, \\ p(t); \sum_{j=0}^{m} p_j\omega_j(t) = P\omega,\ \ g(t); \sum_{j=0}^{m} g_j\omega_j(t) = G\omega, \\ k(t); \sum_{j=0}^{m} k_j\omega_j(t) = K\omega,\ \ q(t); \sum_{j=0}^{m} q_j\omega_j(t) = Q\omega, \end{cases} \tag{8}$$

where

$$\begin{cases} R = [r_0, r_1, r_2, ..., r_n]^T,\ S = [s_0, s_1, s_2, ..., s_n]^T,\ P = [p_0, p_1, p_2, ..., p_n], \\ G = [g_0, g_1, g_2, ..., p_m],\ K = [k_0, k_1, k_2, ..., k_m],\ Q = [q_0, q_1, q_2, ..., q_m], \\ F = [F_0, F_1, F_2, ..., F_m],\ F_j = [f_{0j}, f_{1j}, f_{2j}, ..., f_{nj}]^T, \quad j = 0,1,2,...,m. \end{cases}$$

Therefore, the approximate solution of the $u(x,t)$ can be shown as

$$U_{n,m}(x,t) = \sum_{i=0}^{n}\sum_{j=0}^{m} u_{ij}\upsilon_i(x)\omega_j(t) = \upsilon^T U\omega, \tag{9}$$

where $U = [U_0, U_1, U_2, ..., U_m]$, with $U_j = [u_{0j}, u_{1j}, u_{2j}, ..., u_{nj}]^T$.

Therefore, for finding the numerical approximation solution of $u(x,t)$ we must find the matrix U. The matrix U is an $(n+1)\times(m+1)$ matrix which contains $(n+1)\times(m+1)$ unknown coefficients. To find these $(n+1)\times(m+1)$ unknowns, we have to generate $(n+1)\times(m+1)$ equations.

Corollary 2.1. Let $g_{n,m}(x,y) = \upsilon^T G\omega,$ and $G = [G_0, G_1, G_2, ..., G_m]$ with $G_j = [g_{0,j}, g_{1,j}, g_{2,j}, ..., g_{n,j}]$, then

$$\begin{cases} \frac{d^r}{dx^r} g_{n,m}(x,y) = \upsilon^T (D_x^T)^r G\omega, \\ \frac{d^r}{dy^r} g_{n,m}(x,y) = \upsilon^T G D_y^r \omega, \end{cases} \tag{10}$$

where D is the operational derivative matrix.

Firstly, by applying Eqs. (8) and (9) in Eq. (2), we have

$$\upsilon^T U\omega(0) = \upsilon^T R,$$

which implies

$$U\omega(0) = R, \tag{11}$$

where $\omega(0) = [\omega_0(0), \omega_1(0), \omega_2(0), \dots, \omega_m(0)]^T$.

And by applying Eqs. (8), (9) and (10) in Eq. (3), we have

$$\upsilon^T U D_t \omega(0) = \upsilon^T S$$

hence

$$U D_t \omega(0) = S, \tag{12}$$

since υ is a basis vector.

Similarly, by using Eqs. (8), (9) and (10) in Eqs. (4)-(7), we have

$$\upsilon^T(0) U\omega = P\omega,$$

$$\upsilon^T(1) U\omega = G\omega,$$

$$\upsilon^T(0)(D_x^T)^2 U\omega = K\omega,$$

$$\upsilon^T(1)(D_x^T)^2 U\omega = Q\omega,$$

which implies

$$\upsilon^T(0) U = P, \tag{13}$$

$$\upsilon^T(1) U = G, \tag{14}$$

$$\upsilon^T(0)(D_x^T)^2 U = K, \tag{15}$$

$$\upsilon^T(1)(D_x^T)^2 U = Q, \tag{16}$$

where $\upsilon^T(0) = [\upsilon_0(0), \upsilon_1(0), \dots, \upsilon_n(0)]^T$ and $\upsilon^T(1) = [\upsilon_0(1), \upsilon_1(1), \dots, \upsilon_n(1)]^T$.

Finally, by applying Eqs. (8), (9) and (10) in Eq. (1), we obtain

$$\upsilon^T U D_t^2 \omega - \alpha \upsilon^T (D_x^T)^4 U\omega = \upsilon^T F\omega,$$

hence the residual $Res(x,t)$ for above equation can be written as

$$Res(x,t) = \upsilon^T H\omega,$$

where

$$H = (U D_t^2 - \alpha (D_x^T)^4 U - F),$$

since υ and ω are basic vectors.

For finding a typical matrix formulation, similar to the typical Tau method, we eliminate two last columns and four last rows of the matrix H, then we generate $(n-3)\times(m-1)$ linear algebraic equations by using the following algebraic equations:

$$H_{ij}=0,\quad i=0,1,2,\ldots,n-4,\quad j=0,1,2,\ldots,m-2. \qquad (17)$$

Therefore, we can find $2n+2$ linear algebraic equations from Eqs. (11) and (12), $(m-1)$ linear algebraic equations by choosing $(m-1)$ equations from Eq.(13) and similarly, $(m-1)$ equations from Eq. (14), $(m-1)$ equations from Eq. (15), $(m-1)$ equations from Eq. (16) and finally, $(n-3)\times(m-1)$ equations from Eq. (17). Notice that in this paper, we eliminate two last elements of Eqs.(13)-(16). Now, a system of $(n+1)\times(m+1)$ equations is generated.

3. Application on Several Bases

In this section, we introduce the shifted Standard and shifted Chebyshev bases and applied this bases for numerical computations of the method.

3.1. Shifted Standard Bases

In this section, we give some properties of shifted standard bases. Let, $\upsilon=[1,(x-\frac{l}{2}),(x-\frac{l}{2})^2,\ldots,(x-\frac{l}{2})^n]^T$ and $\omega=[1,(t-\frac{T}{2}),(t-\frac{T}{2})^2,\ldots,(t-\frac{T}{2})^m]$. Therefore we have

$$u_{n,m}(x,t)=\sum_{i=0}^{n}\sum_{j=0}^{m}u_{ij}(x-\frac{l}{2})^i(t-\frac{T}{2})^j=\upsilon^T U\omega. \qquad (18)$$

Now for computing $(n+1)\times(m+1)$ unknown coefficients u_{ij} in Eq.(18), with the matrix formulation method, we can obtain $(n+1)\times(m+1)$ linear algebraic equation from Eqs. (11)-(17). In addition, the matrices D_x and D_t are the operational derivative matrices given by

$$D_x=\begin{bmatrix}0&0&0&0&\ldots&0&0\\1&0&0&0&\ldots&0&0\\0&2&0&0&\ldots&0&0\\0&0&3&0&\ldots&0&0\\\vdots&\vdots&\vdots&\vdots&\vdots&\vdots&\vdots\\0&0&0&0&\ldots&n&0\end{bmatrix}_{(n+1)(n+1)},\quad D_t=\begin{bmatrix}0&0&0&0&\ldots&0&0\\1&0&0&0&\ldots&0&0\\0&2&0&0&\ldots&0&0\\0&0&3&0&\ldots&0&0\\\vdots&\vdots&\vdots&\vdots&\vdots&\vdots&\vdots\\0&0&0&0&\ldots&m&0\end{bmatrix}_{(m+1)(m+1)}.$$

So, by using above matrices and Eqs.(11)-(17), the unknown coefficients u_{ij} can be obtained. Then by Eq.(18), $u_{n,m}(x,t)$ can be calculated.

3.2. Shifted Chebyshev Bases

The matrix formulation method by using shifted Chebyshev bases is considered in this section. The shifted Chebyshev polynomials on the interval $[0,s]$ are defined as

$$T_0^s(x) = 1, \quad T_1^s(x) = \frac{2x-s}{s},$$

$$T_i^s(x) = 2(\frac{2x-s}{s})T_{i-1}^s(x) - T_{i-2}^s(x), \quad i = 2,3,\ldots.$$

In this case, the functions $u_{n,m}(x,t)$, $f(x,t)$, $r(x), s(x), p(t)$ and $q(t)$ are written as

$$\begin{cases} u_{n,m}(x,t);\sum_{i=0}^{n}\sum_{j=0}^{m}u_{ij}T_i^l(x)T_j^T(t) \quad f(x,t);\sum_{i=0}^{''n}\sum_{j=0}^{''m}f_{ij}T_i^l(x)T_j^T(t), \\ r(x);\sum_{i=0}^{''n}r_iT_i^l(x),\ s(x);\sum_{i=0}^{''n}s_iT_i^l(x),\ p(t);\sum_{j=0}^{''m}p_jT_j^T(t), \\ g(t);\sum_{j=0}^{''m}g_jT_j^T(t),\ k(t);\sum_{j=0}^{''m}k_jT_j^T(t),\ q(t);\sum_{j=0}^{''m}q_jT_j^T(t), \end{cases} \tag{19}$$

where the symbol $('')$ over $\sum$ indicates that the first and last terms must be halved. Therefore, suppose that $\upsilon = [T_0^l(x), T_1^l(x), \ldots, T_n^l(x)]^T$ and $\omega = [T_0^T(t), T_1^T(t), \ldots, T_m^T(t)]$. Then the matrices D_x and D_t for odd n (or m) are given as

$$D_x = \frac{2}{l}\begin{bmatrix} 0 & 0 & 0 & 0 & \ldots & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & \ldots & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 & \ldots & 0 & 0 & 0 \\ 3 & 0 & 6 & 0 & \ldots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\ 0 & 2(n-1) & 0 & 2(n-1) & \ldots & 2(n-1) & 0 & 0 \\ n & 0 & 2n & 0 & \ldots & 0 & 2n & 0 \end{bmatrix}_{(n+1)(n+1)},$$

and for even m (or n) are given as

$$D_t = \frac{2}{T}\begin{bmatrix} 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 & \dots & 0 & 0 & 0 \\ 3 & 0 & 6 & 0 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\ (m-1) & 0 & 2(m-1) & 0 & \dots & 0 & 2(m-1) & 0 \\ 0 & 2m & 0 & 2m & \dots & 2m & 0 & 2m \end{bmatrix}_{(m+1)(m+1)} .$$

Also, r_i , p_j and f_{ij} are computed by the following relations

$$r_i = (\frac{2}{n})\sum_{k=0}^{''n} r(x_k)\cos(\frac{ik\pi}{n}), \quad i = 0,1,\dots,n,$$

$$p_j = (\frac{2}{m})\sum_{s=0}^{''m} p(t_s)\cos(\frac{js\pi}{m}), \quad j = 0,1,\dots,m,$$

$$f_{ij} = (\frac{4}{n.m})\sum_{k=0}^{''n}\sum_{s=0}^{''m} f(x_k,t_s)\cos(\frac{ik\pi}{n})\cos(\frac{js\pi}{m}),$$

where

$$x_k = \frac{1}{2}[(l-0)\cos(\frac{k\pi}{n}) + (l+0)], \quad k = 0,1,\dots,n,$$

$$t_s = \frac{1}{2}[(T-0)\cos(\frac{s\pi}{m}) + (T+0)], \quad s = 0,1,\dots,m,$$

similarly, s_i and q_j, g_j and k_j can be computed.

4. Numerical results

In this section, we illustrate efficiency and accuracy of the presented method by the following numerical examples. We define some of the errors as follows:

$$\| u_{n,m} - u^* \|_\infty = \max\{| u_{n,m}(x,t) - u^*(x,t) |, \quad 0 \le x \le l, \quad 0 \le t \le T\},$$
$$\| u_{n,m} - u^* \|_{t_j,\infty} = \max\{| u_{n,m}(x,t_j) - u^*(x,t_j) |, \quad 0 \le x \le l\},$$

where $u_{n,m}(x,t)$ is the obtained approximation result for n and m and u^* is the exact solution of the problem.

Example 4.1 *Consider the following fourth-order equation*

$$\begin{cases} u_{tt} - u_{xxxx} = -2(1+x^2-x^3-x^5)+120x(1-t-t^2), \\ u(x,0) = 1+x^2-x^3-x^5,\ u_t(x,0) = -1-x^2+x^3+x^5, \\ u(0,t) = 1-t-t^2,\ u(1,t) = 0, \\ u_{xx}(0,t) = 2(1-t-t^2),\ u_{xx}(1,t) = -24(1-t-t^2). \end{cases}$$

By using the SS base and choosing $n = m = 5$, we obtain $u_{5,5}(x,t) = (1+x^2-x^3-x^5)(1-t-t^2)$, which is the exact solution of the problem.

Example 4.2 *Consider the following fourth-order equation*

$$\begin{cases} u_{tt} - u_{xxxx} = (1-\pi^4) sin(\pi x) exp(-t), \\ u(x,0) = sin(\pi x),\ u_t(x,0) = -sin(\pi x), \\ u(0,t) = u(1,t) = u_{xx}(0,t) = u_{xx}(1,t) = 0, \end{cases}$$

The exact solution of this problem is $u(x,t) = sin(\pi x) exp(-t)$.

The maximum obtained errors and some other errors by the presented method for several values of m and n, for SS and SC bases, are reported in Table 1.. Furthermore the graphs of error functions for SS and SC bases are given in Fig 1. and 2. respectively.

Table 1.

The maximum errors ($\| u_{n,m} - u^* \|_\infty$) from Example 4.2.

	$m = n = 10$	$m = n = 20$	$m = n = 30$
SS	7.055×10^{-9}	7.171×10^{-22}	6.658×10^{-39}
SC	1.027×10^{-10}	1.607×10^{-26}	7.065×10^{-42}

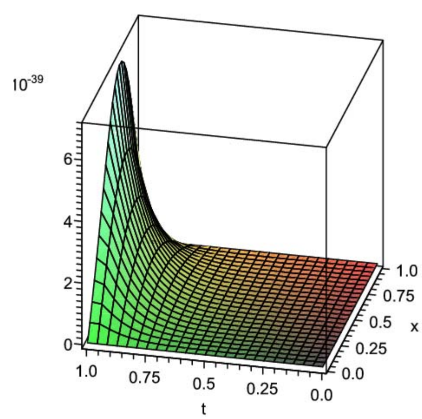

Fig 1.: Plot of error function $|u(x,t) - u_{30,30}(x,t)|$, with SS bases for Example 4.2.

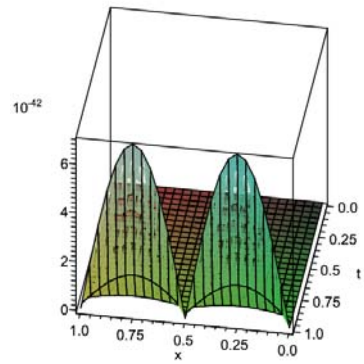

Fig 2.: Plot of error function $|u(x,t)-u_{30,30}(x,t)|$, with SC bases for Example 4.2.

Example 4.3 *Consider the following fourth-order equation*

$$\begin{cases} u_{tt}-u_{xxxx}=(1-2\pi^2)cos(\pi x)exp(t), \\ u(x,0)=cos(\pi x),\ u_t(x,0)=cos(\pi x),\ u(0,t)=exp(t), \\ u(1,t)=-exp(t),\ u_{xx}(0,t)=-\pi^2 exp(t),\ u_{xx}(1,t)=\pi^2 exp(t). \end{cases}$$

The exact solution of this problem is $u(x,t)=cos(\pi x)exp(t)$.

The obtained errors by the presented method for several values of m and n, for SS and SC bases, are reported in Table 2. and the graphs of error functions for SS and SC bases are given in Fig 3. and 4. respectively.

Table 2.

The maximum errors ($\|u_{n,m}-u^*\|_\infty$) from Example 4.3.

	$m=n=10$	$m=n=20$	$m=n=30$
SS	5.409×10^{-4}	3.271×10^{-13}	9.279×10^{-25}
SC	1.489×10^{-5}	4.455×10^{-17}	1.783×10^{-42}

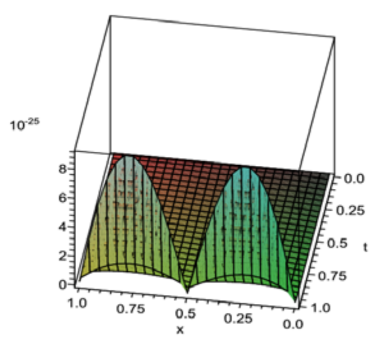

Fig 3.: Plot of error function $|u(x,t)-u_{30,30}(x,t)|$, with SS bases for Example 4.3.

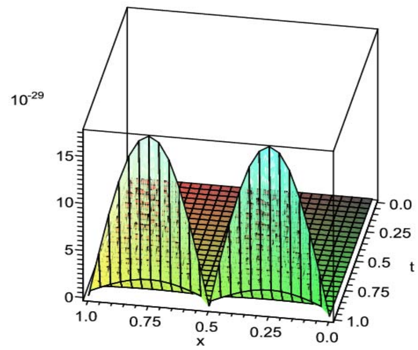

Fig 4.: Plot of error function $|u(x,t)-u_{30,30}(x,t)|$, with SC bases for Example 4.3.

Example 4.4 *Consider the following fourth-order equation*

$$\begin{cases} u_{tt}-u_{xxxx}=(\pi^4-1)sin(\pi x)cos(t), \\ u(x,0)=sin(\pi x),\ u_t(x,0)=0, \\ u(0,t)=u(1,t)=u_{xx}(0,t)=u_{xx}(1,t)=0, \end{cases}$$

The exact solution of this problem is $u(x,t)=sin(\pi x)cos(t)$.

The computational results for the Example 4.4. are presented in Table 3. In addition, the plots of corresponding error functions are shown in Fig 5. and Fig 6.

Table 3.

The maximum errors ($\|u_{n,m}-u^*\|_\infty$) from Example 4.4.

	$m=n=10$	$m=n=20$	$m=n=30$
SS	2.213×10^{-4}	9.493×10^{-15}	9.888×10^{-27}
SC	1.152×10^{-7}	4.611×10^{-21}	2.287×10^{-36}

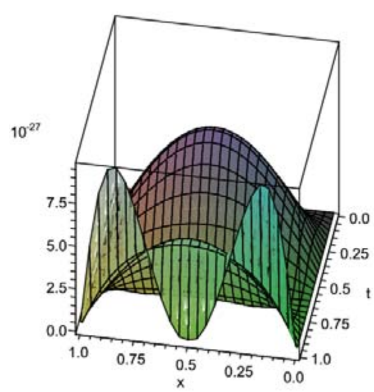

Fig. 3: Plot of error function $|u(x,t)-u_{30,30}(x,t)|$, with SS bases for Example 4.4.

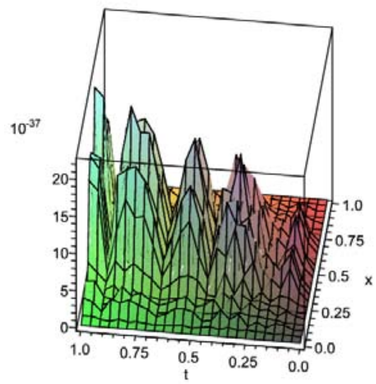

Fig. 4: Plot of error function $|u(x,t)-u_{30,30}(x,t)|$, with SC bases for Example 4.4.

6. Conclusions

In this research, A high accuracy numerical scheme is proposed for the numerical studying of a forth order parabolic partial differential equation with some initial and boundary conditions. The most important section of our method is converting the model of PDE to a linear system of algebraic equations. The method is based on finding a solution in the form of a polynomial in two variables. In addition, by increasing the number of terms in the series we can decrease the error of this process. Finally, the effectively of our method can be shown by the numerical test problems.

Acknowledgments

The authors are very grateful to the reviewers for carefully reading the paper and for their constructive comments and suggestions which have improved the paper.

R E F E R E N C E S

[1] *E.L. Ortiz*, The Tau method, SIAM. J. Numer. Anal. 6: 480-492, 1969.

[2] *E.L. Ortiz, H. Samara*, An operational approach to the Tau method for the numerical solution of nonlinear differential equations, Computing 27: 15-25, 1981.

[3] *K.M. Liu, E.L. Ortiz*, Approximation of eigenvalues defined by ordinary differential equations with the Tau method, in: B. Kagestrm, A. Ruhe (Eds.) Matrix Pencils, Springer-Verlag, Berlin, 1983.

[4] *K.M. Liu, E.L. Ortiz*, Numerical solution of eigenvalue problems for partial differential equations with the Tau-lines Method, Comput. Math. Appl. 12B: 1153-1168, 1986.

[5] *B. Soltanalizadeh*, Numerical analysis of the one-dimensional Heat equation subject to a boundary integral specification, Optics Communications, 284: 2109-2112, 2011.
[6] *H.R. Ghehsareh, B. Soltanalizadeh,S. Abbasbandy*, A matrix formulation to the wave equation with nonlocal boundary condition, Inter. J. Comput. Math., 88: 1681-1696, 2011.
[7] *S.M. Hosseini and S. Shahmorad*, Tau numerical solution of Fredholm integro-differential equations with arbitrary polynomial bases, Appl. Math. Model. 27: 145-154, 2003.
[8] A. Tari and S. Shahmorad, A computational method for solving two-dimensional linear Fredholm integral equations of the second kind, ANZIAM J. 49: 543-549, 2008.
[9] *S. Abbasbandy and E. Taati*, Numerical solution of the system of nonlinear Volterra integro-differential equations with nonlinear differential part by the operational Tau method and error estimation, J. Comput. Appl. Math. 231: 106-113, 2009.
[10] *B. Soltanalizadeh, A. Yildirim*, Application of Differential Transformation method for numerical computation of Regularized Long Wave equation, Zeitschrift fuer Naturforschung A., 67a (2012) 160-166.
[11] *B. Soltanalizade*h, Differential Transformation method for solving one-space-dimensional Telegraph equation, computational & applied mathematics, 30: 639-653, 2011.
[12] R. *Abazari, A. Borhanifar*, Numerical study of the solution of the Burgers and coupled Burgers equations by a differential transformation method, Comput. Math. Appl. 59: 2711–2722, 2010.
[13] *R. Abazari, M. Ganji*, Extended two-dimensional DTM and its application on nonlinear PDEs with proportional delay, Int. J. Comput. Math, 88: 1749–1762, 2011.
[14] *A. Gokdogö an , M. Merdan, A. Yildirim*, A multistage differential transformation method for approximate solution of Hantavirus infection model. Commun Nonlinear Sci Numer Simulat, 17: 1-8, 2012.
[15] *T. Aziz, A. Khan, J. Rashidini*, Spline methods for the solution of fourth-order parabolic partial differential equations, Appl. Math. Comput. 167: 153-166, 2005.
[16] *H. Caglar, N. Caglar*, Spline methods for the solution of fourth-order parabolic partial differential equations, Appl. Math. Comput. 201: 597-603, 2008.
[17] *M.K. Jain, S.R.K. Iyengar, A.G. Lone*, Higher order difference formulas for a fourth order parabolic partial differential equation, Int. J. Numer. Methods Eng. 10: 1357-1367, 1976.
[18*] J. Douglas*, The solution of the diffusion equation by a high order correct difference equation, J. Math. Phys. 35: 145-151, 1956.
[19] *A.M. Wazwaz*, Analytic treatment for variable coefficient fourth order parabolic partial differential equations, Appl. Math. Comput. 123: 219-227, 2001.