

Article

An Efficient Scheme for Time-Dependent Emden-Fowler Type Equations Based on Two-Dimensional Bernstein Polynomials

Ahmad Sami Bataineh ^{1,2}, Osman Rasit Isik ³, Abedel-Karrem Alomari ⁴,
Mohammad Shatnawi ⁵ and Ishak Hashim ^{2,*}

¹ Department of Mathematics, Faculty of Science, Al-Balqa' Applied University, Al Salt 19117, Jordan; a_s_bataineh@bau.edu.jo

² Department of Mathematical Sciences, Faculty of Science & Technology, Universiti Kebangsaan Malaysia, Bangi 43600 UKM, Selangor, Malaysia

³ Elementary Mathematics Education Program, Faculty of Education, Mugla Sitki Kocman University, Mugla 48000, Turkey; osmanrasit@mu.edu.tr

⁴ Department of Mathematics, Faculty of Science, Yarmouk University, Irbid 21163, Jordan; Abdalkareem@yu.edu.jo

⁵ Department of Basic Science, Al-Huson University College, Al-Balqa' Applied University, P.O. Box 50, Al-Huson, Irbid 21510, Jordan; taib.Shatnawi@bau.edu.jo

* Correspondence: ishak_h@ukm.edu.my

Received: 29 June 2020; Accepted: 27 August 2020; Published: 1 September 2020



Abstract: In this study, we introduce an efficient computational method to obtain an approximate solution of the time-dependent Emden-Fowler type equations. The method is based on the 2D-Bernstein polynomials (2D-BPs) and their operational matrices. In the cases of time-dependent Lane–Emden type problems and wave-type equations which are the special cases of the problem, the method converts the problem to a linear system of algebraic equations. If the problem has a nonlinear part, the final system is nonlinear. We analyzed the error and give a theorem for the convergence. To estimate the error for the numerical solutions and then obtain more accurate approximate solutions, we give the residual correction procedure for the method. To show the effectiveness of the method, we apply the method to some test examples. The method gives more accurate results whenever increasing n, m for linear problems. For the nonlinear problems, the method also works well. For linear and nonlinear cases, the residual correction procedure estimates the error and yields the corrected approximations that give good approximation results. We compare the results with the results of the methods, the homotopy analysis method, homotopy perturbation method, Adomian decomposition method, and variational iteration method, on the nodes. Numerical results reveal that the method using 2D-BPs is more effective and simple for obtaining approximate solutions of the time-dependent Emden-Fowler type equations and the method presents a good accuracy.

Keywords: Bernstein polynomials; operational matrices; Emden-Fowler equation

1. Introduction

The heat equation which expresses the diffusion of heat can be given by the following form [1–3]:

$$\frac{\partial^2 y}{\partial x^2}(x, t) + \frac{r}{x} \frac{\partial y}{\partial x}(x, t) + af(x, t)g(y(x, t)) + h(x, t) = \frac{\partial y}{\partial t}(x, t), \quad 0 < x \leq L, \quad 0 < t < T, \quad (1)$$

subject to

$$y(0, t) = \alpha, \quad y_x(0, t) = 0, \quad (2)$$

where $y(x, t)$ represents the temperature and t the time, $r > 0$ and a is an integer, α is a constant and the term $f(x, t)g(y) + h(x, t)$ is the nonlinear heat source, where f, g and h are some smooth functions. For the steady case with $h(x, t) = 0$ and $r = 2$, Equation (1) is the Emden-Fowler equation [4–6], that is,

$$\frac{\partial^2 y}{\partial x^2} + \frac{2}{x} \frac{\partial y}{\partial x} + af(x)g(y) = 0. \quad (3)$$

In addition, if $f(x) = 1$, then Equation (3) becomes the Lane–Emden equation of mathematical physics [7,8].

In this work, we study the numerical solution, which is a simply applied and effective method, of the time-dependent Emden-Fowler type equations. First, the heat-type Equation (1) is considered. To solve Equation (1) by the proposed numerical technique, an operational matrix based on two-dimensional Bernstein polynomials is constructed. Second, with the same notations, we obtain the approximate solutions of the following wave-type equation:

$$\frac{\partial^2 y}{\partial x^2}(x, t) + \frac{r}{x} \frac{\partial y}{\partial x}(x, t) + af(x, t)g(y(x, t)) + h(x, t) = \frac{\partial^2 y}{\partial t^2}(x, t), \quad (4)$$

subject to

$$y(0, t) = \alpha, \quad y_x(0, t) = 0. \quad (5)$$

The main issue arisen in the analysis of Equations (1) and (4) is treating the singularity at $x = 0$. To overcome this singularity behavior at this point, there are some semi-analytical methods which are used to solve nonlinear problems in the literature such as Adomian's decomposition method (ADM) [1,5], homotopy analysis method (HAM) [9–12], variational iteration method (VIM) [4,13–17], and homotopy perturbation method (HPM) [18–22]. Wazwaz [5] used ADM to get approximate analytic solutions of (3). He obtained more accurate results for several examples. Another study by Wazwaz [1] for time-dependent Emden-Fowler equation was given with a generalization of the previous work. He employed ADM to solve Equations (1) and (4) and obtained convergent results for several examples. Chowdhury and Hashim [3] applied HPM to get approximate analytical solutions of (1) and (4). Bataineh et al. [6] applied HAM to solve the problems (1) and (4). Belal et al. [23] used VIM to solve this problem.

Since the operational matrices method can achieve the singularity behavior at the point $x = 0$, it has been applied to some singular problems. Yousefi and Behroozifar [24] applied the Bernstein operational matrix to solve the Emden-Fowler equation for the steady-state case. The same problem was considered by Gupta and Sharma [25] and solved by the Taylor series method. Another work to solve the steady-state problem was given by Wazwaz et al. [26]. They applied VIM to the problem. When only the Lane–Emden equation is considered, there are many numerical methods that depend on ADM [27,28], VIM [29–32], HPM [8,33], and operational matrices method [34] or Bernstein collocation method [35] were used to get analytic or numerical solutions.

The methods based on operational matrix of differentiation were given in various forms in the literature. Some of them were constituted by Chebyshev polynomials [36], Legendre polynomials [37], and Bernstein polynomials [38–44]. One of the common properties of these polynomials is the basis of polynomial spaces. On the other hand, because of Bernstein polynomials are dense in $L^2(\Omega)$ [45] and thus yields good approximation results, they have been used often recently to solve the problems in two-dimensions. The Bernstein operational matrices method is also known as the Bernstein matrix method [44] and Bernstein series solution [35].

In this work, we shall develop a method based Bernstein operational matrices to solve the nonlinear problem 1. The method is applied easily and presents a good accuracy. We first give Bernstein polynomials in 1D and 2D in Section 2. The matrix representations of the approximate functions and their operational matrices for both dimensions are given next. Section 3 presents the method of solution for heat-type and wave-type equations. In Section 4, we constitute the residual correction procedure

both to estimate the absolute error and to get the corrected approximate solutions. The most important property of the procedure can be applied even if the exact solution is unknown. Section 5 contains some examples including linear and nonlinear models to demonstrate the applicability of the method. First, we apply the method to time-dependent Lane–Emden type problems and singular wave-type equations. We perform the method for different number of nodes. The numerical results show that increasing number of nodes gives a sequence of approximations, which converges to the exact solution, for linear problems. The method also gives good results for nonlinear problems. Residual correction procedure estimates the error in general. On the other hand, the corrected approximate solutions can be obtained and its error is less than the approximate solution. We compare the approximate solutions obtained by the method with the approximate solutions obtained by ADM, HPM, HAM, and VIM by calculating the error on some nodes in $[0, L] \times [0, T]$. In the last section, we summarize the results.

2. 2D Bernstein Polynomials (2D-BPs) and Their Operational Matrices

In this section, because of defining two-dimensional Bernstein polynomials easily, we will give one-dimensional, Bernstein polynomials first. The Bernstein polynomials of degree m (1D-BPs) on $[0, 1]$ are defined as [46]

$$B_{i,m}(x) = \binom{m}{i} x^i (1-x)^{m-i}, \quad i = 0, 1, \dots, m, \tag{6}$$

where $\binom{m}{i} = \frac{m!}{i!(m-i)!}$ with $B_{i,m} = 0$ if $i < 0$ or $i > m$. Equation (6) can be rewritten as

$$\begin{aligned} B_{i,m}(x) &= \binom{m}{i} x^i (1-x)^{m-i} \\ &= \binom{m}{i} x^i \left(\sum_{k=0}^{m-i} (-1)^k \binom{m-i}{k} x^k \right) \\ &= \sum_{k=0}^{m-i} (-1)^k \binom{m}{i} \binom{m-i}{k} x^{k+i}, \quad i = 0, 1, \dots, m. \end{aligned}$$

Let $y : I \rightarrow \mathbb{R}$ be any real-valued continuous function. The truncated Bernstein polynomials approximation of y is

$$y(x) \simeq y_m(x) = \sum_{i=0}^m c_i B_{i,m}(x) = C_m \Phi_m(x), \tag{7}$$

where

$$\begin{aligned} C_m &= [c_0, c_1, \dots, c_m], \\ \Phi_m(x) &= [B_{0,m}(x), B_{1,m}(x), \dots, B_{m,m}(x)]^T. \end{aligned}$$

To ease computational complexity, the matrix representations of

$$\{y'_m(x), y''_m(x), \dots, y_m^{(k)}(x)\}$$

are to be determined. Let $(A_{i+1})_{1,j}$ and $T_m(x)$ be $1 \times (m + 1)$ and $(m + 1) \times 1$ matrices respectively for $j = 0, 1, \dots, m$ as follows:

$$(A_{i+1})_{1,j} = \begin{cases} 0, & j < i, \\ (-1)^{j-i} \binom{m}{i} \binom{m-i}{j-i}, & j \geq i. \end{cases}$$

$$T_m(x) = \begin{bmatrix} 1 \\ x \\ \vdots \\ x^m \end{bmatrix}. \tag{8}$$

Then, the polynomial $B_{i,m}(x)$ is given by

$$B_{i,m}(x) = A_{i+1}T_m(x), \tag{9}$$

which gives the identity:

$$\Phi(x) = AT_m(x), \tag{10}$$

where

$$A = \begin{bmatrix} A_1 \\ A_2 \\ A_3 \\ \vdots \\ A_{m+1} \end{bmatrix}. \tag{11}$$

Note that

$$T_m(x) = A^{-1}\Phi(x), \tag{12}$$

and

$$T'_m(x) = \begin{bmatrix} 0 \\ 1 \\ 2x \\ \vdots \\ mx^{m-1} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & \dots & 0 \\ 0 & 2 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & m & 0 \end{bmatrix} \begin{bmatrix} 1 \\ x \\ x^2 \\ \vdots \\ x^m \end{bmatrix} = LT_m(x), \tag{13}$$

where

$$L = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & \dots & 0 \\ 0 & 2 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & m & 0 \end{bmatrix}. \tag{14}$$

By using (10), we write $y'_m(x)$ as

$$\begin{aligned} y'_m(x) &= C\Phi'(x) = CAT'_m(x) = CALT_m(x) \\ &= CALA^{-1}\Phi(x) = CD\Phi(x), \end{aligned} \tag{15}$$

where $D = ALA^{-1}$. Similarly, for higher-order derivatives of $y_m(x)$, one has

$$\begin{aligned} y''_m(x) &= C(\Phi'(x))' = C(D\Phi'(x)) = CD^2\Phi(x), \\ &\vdots \\ y_m^{(k)}(x) &= C(\Phi'(x))^{(k-1)} = C(D\Phi(x))^{(k-1)} = \dots = CD^k\Phi(x). \end{aligned}$$

The Bernstein polynomials of degree mn (2D-BPs) on $[0, 1] \times [0, 1]$ are defined as [47,48]

$$B_{i,j}^{m,n}(x, t) = \binom{m}{i} \binom{n}{j} x^i (1-x)^{m-i} t^j (1-t)^{n-j}, \quad (i = 0, 1, \dots, m, \quad j = 0, 1, \dots, n),$$

which can be rewritten as [45]

$$B_{i,j}^{m,n}(x, t) = \sum_{k=0}^{m-i} \sum_{r=0}^{n-j} (-1)^{k+r} \binom{m}{i} \binom{n}{j} \binom{m-i}{k} \binom{n-j}{r} x^{i+k} t^{j+r}. \tag{16}$$

Theorem 1 ([45]). *Let $\Omega = [0, 1] \times [0, 1]$. The set of all two-dimensional Bernstein polynomials $\{B_{i,j}^{m,n}(x, t)\}_{i,j=0}^{\infty}$ on the box Ω is dense in $L^2(\Omega)$.*

Figure 1 shows the two components of the two-dimensional Bernstein polynomials of order (5,5).

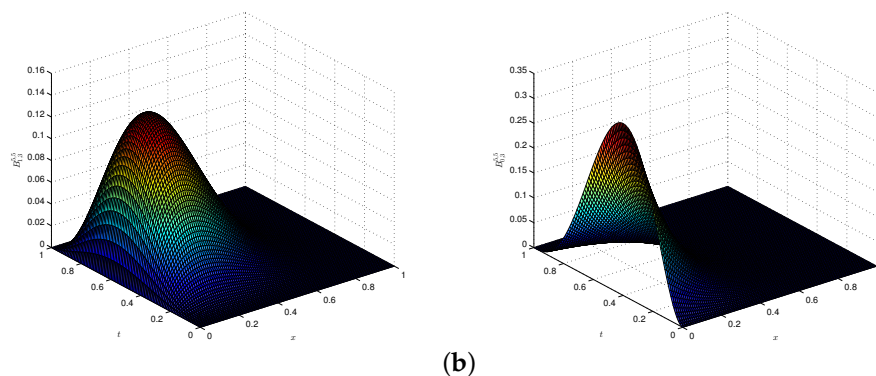


Figure 1. Two components of the two-dimensional Bernstein polynomials of order (5,5): (a) $B_{0,2}^{5,5}(x, t)$ and (b) $B_{1,3}^{5,5}(x, t)$.

Now, assume that y is a real-valued function defined on $\Omega \subset \mathbb{R}^2$. We want to approximate $y(x, t)$ by the truncated 2D-BPs series, i.e.,

$$y(x, t) \simeq y_{m,n}(x, t) = \sum_{i=0}^m \sum_{j=0}^n c_{i,j} B_{i,j}^{m,n}(x, t). \tag{17}$$

If the identity in [47] is applied to (17), we get

$$\sum_{i=0}^m \sum_{j=0}^n c_{i,j} B_{i,j}^{m,n}(x, t) = \sum_{i=0}^m \sum_{j=0}^n c_{i,j} B_{i,m}(x) B_{j,n}(t). \tag{18}$$

Then, $y_{m,n}$ in (17) can be found using

$$y_{m,n}(x, t) = C^T \Theta(x) \Psi(t), \tag{19}$$

where

$$\begin{aligned}
 C &= [c_{i,j}]_{1 \times (m+1)(n+1)}^T = [c_{0,0}, c_{0,1}, \dots, c_{0,n}, \dots, c_{m,0}, c_{m,1}, \dots, c_{m,n}]^T, \\
 \Theta(x) &= \begin{bmatrix} \Phi(x) & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \Phi(x) & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \Phi(x) \end{bmatrix}, \\
 \Psi(t) &= [\mathbf{B}_{0,n}(t) \quad \mathbf{B}_{1,n}(t) \quad \dots \quad \mathbf{B}_{n,n}(t)], \\
 \mathbf{B}_{j,n}(t) &= [B_{j,n}(t) \quad B_{j,n}(t) \quad \dots \quad B_{j,n}(t)].
 \end{aligned} \tag{20}$$

The partial derivatives $\partial y/\partial x$ and $\partial y/\partial t$ can be approximated as

$$\frac{\partial y}{\partial x}(x, t) \simeq \frac{\partial y_{m,n}}{\partial x}(x, t) = C \frac{\partial \Theta}{\partial x}(x) \Psi(t), \tag{21}$$

$$\frac{\partial y}{\partial t}(x, t) \simeq \frac{\partial y_{m,n}}{\partial t}(x, t) = C \Theta(x) \frac{\partial \Psi}{\partial t}(t). \tag{22}$$

Upon using (9), (13), and (15), one gets the identities

$$\begin{aligned}
 \frac{d\Theta}{dx}(x) &= \begin{bmatrix} \frac{\partial \Phi}{\partial x}(x) & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \frac{\partial \Phi}{\partial x}(x) & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \frac{\partial \Phi}{\partial x}(x) \end{bmatrix} \\
 &= \begin{bmatrix} D\Phi(x) & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & D\Phi(x) & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & D\Phi(x) \end{bmatrix} \\
 &= \overline{D\Phi(x)},
 \end{aligned} \tag{23}$$

$$\begin{aligned}
 \frac{d\Psi}{dt}(t) &= \left[\frac{\partial \mathbf{B}_{(0,n)}(t)}{\partial t} \quad \frac{\partial \mathbf{B}_{(1,n)}(t)}{\partial t} \quad \dots \quad \frac{\partial \mathbf{B}_{(n,n)}(t)}{\partial t} \right]^T \\
 &= [A_1 LT_n(t) \quad A_2 LT_n(t) \quad \dots \quad A_{n+1} LT_n(t)] \\
 &= \overline{ALT_n(t)},
 \end{aligned} \tag{24}$$

where

$$\begin{aligned}
 \overline{D\Phi(x)} &= \begin{bmatrix} D\Phi(x) & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & D\Phi(x) & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & D\Phi(x) \end{bmatrix}, \\
 \overline{ALT_n(t)} &= [A_1 LT_n(t) \quad A_2 LT_n(t) \quad \dots \quad A_{n+1} LT_n(t)].
 \end{aligned}$$

Hence, we get

$$\frac{\partial y_{m,n}}{\partial x}(x, t) = \overline{CD\Phi(x)\Psi(t)}, \tag{25}$$

$$\frac{\partial y_{m,n}}{\partial t}(x, t) = \overline{C\Theta(x)ALT_n(t)}. \tag{26}$$

For the higher-order derivatives, we have

$$\frac{\partial^s y_{m,n}}{\partial x^{s_1} \partial t^{s_2}}(x, t) = \overline{CD^{s_1}\Phi(x)AL^{s_2}T_n(t)}, \quad s = s_1 + s_2, s_1 \neq 0, s_2 \neq 0, \tag{27}$$

$$\frac{\partial^s y_{m,n}}{\partial x^s}(x, t) = \overline{CD^s\Phi(x)\Psi(t)},$$

$$\frac{\partial^s y_{m,n}}{\partial t^s}(x, t) = \overline{C\Theta(x)AL^sT_n(t)}.$$

3. Solving Heat-Type and Wave-Type Equations by 2D-BPs

To approximate the solution of (1) subject to the initial condition (2), one substitutes the identities in (25), (26), and (27) into (1). The residual $Re(x, t)$ are given by

$$\begin{aligned} Re(x, t) = & \overline{CD^2\Phi(x)\Psi(t)} + \frac{r}{x}\overline{CD\Phi(x)\Psi(t)} + af(x, t)g(\overline{C\Theta(x)\Psi(t)}) \\ & + h(x, t) - \overline{C\Theta(x)ALT_n(t)}. \end{aligned} \tag{28}$$

Taking the nodes

$$\{(x_i, t_i) : 0 < x_1 < \dots < x_{m-1} = 1, 0 \leq t_1 < \dots < t_{n+1} = 1\} \subset [0, 1] \times [0, 1], \tag{29}$$

in (28), we obtain $(m - 1)(n + 1)$ equations

$$\begin{aligned} Re(x_i, t_j) = & \overline{CD^2\Phi(x_i)\Psi(t_i)} + \frac{r}{x_i}\overline{CD\Phi(x_i)\Psi(t_i)} + af(x_i, t_i)g(\overline{C\Theta(x_i)\Psi(t_i)}) \\ & + h(x_i, t_i) - \overline{C\Theta(x_i)ALT_n(t_i)}, \quad i = 1, 2, \dots, m - 1, j = 1, 2, \dots, n + 1. \end{aligned} \tag{30}$$

The conditions (2) then yield

$$y(0, t_j) = \overline{C\Theta(0)\Psi(t_j)} = \alpha, \quad \frac{\partial y}{\partial x}(0, t) = \overline{CD\Phi(0)\Psi(t)} = 0. \tag{31}$$

Thus, we have $(m + 1)(n + 1)$ equations. By solving these equations by the command 'fsolve' which uses Newton's method for the unknown coefficients $C_{i,j}$, we can find the matrix C so that $y_{m,n}(x, t)$ is obtained.

In this study, we select the collocation nodes as Newton–Cotes points

$$x_i = \frac{2i - 1}{2(m + 1)}, \quad t_j = \frac{2j - 1}{2(n + 1)}, \quad i = 1, \dots, (m + 1), j = 1, \dots, (n + 1). \tag{32}$$

When applied to the wave-type Equation (4), we get the following:

$$\begin{aligned} Re(x_i, t_j) = & \overline{CD^2\Phi(x_i)\Psi(t_i)} + \frac{r}{x_i}\overline{CD\Phi(x_i)\Psi(t_i)} + af(x_i, t_i)g(\overline{C\Theta(x_i)\Psi(t_i)}) \\ & + h(x_i, t_i) - \overline{C\Theta(x_i)AL^2T_n(t_i)}, \\ & (i = 1, 2, \dots, m - 1, j = 1, 2, \dots, n + 1), \end{aligned} \tag{33}$$

$$y(0, t_j) = \overline{C\Theta(0)\Psi(t_j)} = \alpha, \quad \frac{\partial y}{\partial x}(0, t_j) = \overline{CD\Phi(0)\Psi(t_j)} = 0.$$

4. Error Analysis

In this section, we first give the Bernstein series that converges for the function in C^1 . The residual correction procedure is given to estimate the absolute error and to obtain corrected approximate solutions. To determine the convergence of the method, we define the constant M_{n+1} as follows.

Definition 1. Suppose that $\Omega = [0, 1] \times [0, 1]$ and for $k = 0, 1, \dots, n$, $f^{(k)} \in C^1(\Omega)$, then

$$\left| f(x, t) - \sum_{i=0}^n \sum_{j=0}^n \frac{\partial^{i+j} f}{\partial x^i \partial t^j}(0, 0) \frac{x^i t^j}{i! j!} \right| \leq M_{n+1} \frac{1}{(n+1)!} (x+t)^{n+1}, \tag{34}$$

where all partial derivatives of f of order $n + 1$ are bounded in magnitude M_{n+1} , which is

$$M_{n+1} = \sup_{(\zeta, \tau) \in \Omega} \left| \frac{\partial^{n+1}}{\partial \zeta^k \partial \tau^{n+1-k}} f(\zeta, \tau) \right|, \quad k = 0, 1, \dots, n + 1.$$

Now, the main theorem of function approximation using 2D-BPs are stated as follows.

Theorem 2. Consider $m, n \in \mathbb{N}$ and $\Omega = [0, 1] \times [0, 1]$, $\eta = \max\{m, n\}$. Let $\frac{\partial^k}{\partial x^k \partial t^{i-k}} f(x, t) \in C^1(\Omega)$, $k = 0, 1, \dots, \eta$, $i = 0, 1, \dots, k$. Next, suppose that

$$Y_{m,n} = \text{span} \langle B_{0,0}^{m,n}, B_{0,1}^{m,n}, \dots, B_{m,n}^{m,n} \rangle.$$

Approximating $f(x, t)$ by $f_{m,n}(x, t)$ in the space $Y_{m,n}$ as

$$f(x, t) \simeq f_{m,n}(x, t) = \sum_{i=0}^m \sum_{j=0}^n c_{i,j} B_{i,m}(x) B_{j,n}(t), \tag{35}$$

where $f_{m,n}$ is the best approximation out of $Y_{m,n}$ and supposing that

$$E_{m,n}(f) = \int_0^1 \int_0^1 [f(x, t) - f_{m,n}(x, t)]^2 dx dt, \tag{36}$$

then

$$\lim_{m,n \rightarrow \infty} E_{m,n}(f) = 0. \tag{37}$$

Proof. Let

$$\bar{f}_{m,n}(x, t) = \sum_{i=0}^m \sum_{j=0}^n \frac{\partial^{i+j} f}{\partial x^i \partial t^j}(0, 0) \frac{x^i t^j}{i! j!}. \tag{38}$$

Therefore, by considering Definition 1 and by taking $\eta = \max\{m, n\}$, we get

$$|f(x, t) - \bar{f}_{m,n}(x, t)| \leq M_{\eta+1} \frac{1}{(\eta+1)!} (x+t)^{\eta+1} \tag{39}$$

By considering Taylor’s expansion formula in Definition 1, one has

$$\begin{aligned}
 \|f(x, t) - f_{m,n}(x, t)\|_2^2 &\leq \|f(x, t) - \bar{f}_{m,n}(x, t)\| \\
 &\leq \|f(x, t) - \bar{f}_{\eta,\eta}(x, t)\| \\
 &= \int_0^1 \int_0^1 [f(x, t) - \bar{f}_{\eta,\eta}(x, t)]^2 dx dt \\
 &= \frac{M_{\eta+1}^2}{(\eta + 1)!} \int_0^1 \int_0^1 (x + t)^{2\eta+2} dx dt \\
 &= \frac{M_{\eta+1}^2}{(\eta + 1)!} \frac{2^{2\eta+4} - 2}{(2\eta + 3)(2\eta + 4)}.
 \end{aligned}
 \tag{40}$$

Hence, we get

$$\lim_{m,n \rightarrow \infty} \|f(x, t) - f_{m,n}(x, t)\|_2^2 = 0.
 \tag{41}$$

This concludes the proof. □

Now, we constitute residual correction procedure for the problem. Let $y(x, t)$ be the exact and $y_{m,n}(x, t)$ the 2D-BPs approximate solutions of Equation (1), respectively. Writing $e_{m,n}(x, t) = y(x, t) - y_{m,n}(x, t)$ as the error, then

$$\begin{aligned}
 \frac{\partial^2 e_{m,n}}{\partial x^2} + \frac{r}{x} \frac{\partial e_{m,n}}{\partial x} + af g(y_{m,n} + e_{m,n}) - \frac{\partial e_{m,n}}{\partial t} &= R(x, t), \\
 e_{m,n}(0, t) = \alpha, \quad \frac{\partial e_{m,n}}{\partial x}(0, t) &= 0,
 \end{aligned}
 \tag{42}$$

where

$$R(x, t) = \frac{\partial^2 y_{m,n}}{\partial x^2} + \frac{r}{x} \frac{\partial y_{m,n}}{\partial x} - \frac{\partial y_{m,n}}{\partial t} + h(x, t).$$

Substituting the approximate solution along with the constructed operational matrix in (42) yields the residual equation $Re_{err}(x, t)$ for the error as

$$\begin{aligned}
 Re_{err}(x, t) &= C_{err} \overline{D^2 \Phi(x)} \Psi(t) + \frac{r}{x} C_{err} \overline{D \Phi(x)} \Psi(t) + af(x, t) g [C_{err} \Theta(x) \Psi(t) \\
 &\quad + y_{m,n}(x, t)] - C_{err} \Theta(x) \overline{ALT_n(t)} + R(x, t).
 \end{aligned}
 \tag{43}$$

To solve Equation (43) numerically, we compute it at the following collocation points:

$$\{(x_i, t_i) : 0 < x_1 < \dots < x_{s-1} = 1, 0 \leq t_1 < \dots < t_{p+1} = 1\} \subset [0, 1] \times [0, 1].
 \tag{44}$$

Applying the proposed method to (43) with the given initial conditions gives us the coefficient matrix C_{err} so that we get an approximate solution $\tilde{e}_{s,p}$ for the error. Hence, $y_{m,n} + \tilde{e}_{s,p}$ is another approximate solution, which is called the corrected 2D-BPs approximate solution for the problem. In practice, selecting $s \geq m, p \geq n$ yields better approximation results.

The solution $y_{m,n}^{s,p} := y_{m,n} + \tilde{e}_{s,p}$ improves the approximation $y_{m,n}$ provided that

$$\|e_{m,n} - \tilde{e}_{s,p}\| = \|y - (y_{m,n} + \tilde{e}_{s,p})\| \leq \|y_{m,n}^{s,p}\|.
 \tag{45}$$

Similar definitions and results can be given for the wave-type Equation (4).

5. Application to Several Test Problems

The applicability of the proposed 2D-BPS method is demonstrated via several test problems. To simulate the results, three types of equations are selected. Our results are compared with the

methods of ADM, HPM, VIM, and HAM with $h = -1$. We use the Newton–Cotes points (32) and Maple 15 to simulate the numerical outlets with 50-digit precision.

5.1. Time-Dependent Lane–Emden Type Problems

Example 1. Consider

$$y_{xx} + \frac{2}{x}y_x - (6 + 4x^2 - \cos t)y = y_t, \tag{46}$$

subject to

$$y(0, t) = e^{\sin t}, \quad y_x(0, t) = 0. \tag{47}$$

The exact solution is [1–4]:

$$y(x, t) \simeq e^{\sin t} \left(1 + x^2 + \frac{x^4}{2!} + \frac{x^6}{3!} + \frac{x^8}{4!} + \dots \right) = e^{x^2 + \sin t}. \tag{48}$$

To solve the above problem via the proposed method, we get from Equations (30) and (31) and Equations (46) and (47)

$$\begin{aligned} \text{Re}(x, t) &= \overline{\text{CD}^2\Phi(x)}\Psi(t) + \frac{2}{x}\overline{\text{CD}\Phi(x)}\Psi(t) - (6 + 4x^2 - \cos t)\text{C}\Theta(x)\Psi(t) \\ &\quad - \text{C}\Theta(x)\overline{\text{AL}^s\text{T}_n(t)}, \end{aligned} \tag{49}$$

$$\text{C}\Theta(0)\Psi(t_j) = e^{\sin t_j}, \quad \text{C}^T\overline{\text{CD}\Phi(0)}\Psi(t_j) = 0. \tag{50}$$

Substituting (32) into Equations (49) and (50) for $m = n = 3$ yields a system of linear equations for the unknown coefficients $C_{i,j}$, which once computed numerically gives

$$\begin{aligned} y_{3,3}(x, t) &= 1.0057(1 - x)^3(1 - t)^3 + 3.0171x(1 - x)^2(1 - t)^3 \\ &\quad + 3.9180x^2(1 - x)(1 - t)^3 + 2.3768x^3(1 - t)^3 \\ &\quad + 3.9414(1 - x)^3t(1 - t)^2 + 11.8243x(1 - x)^2t(1 - t)^2 \\ &\quad + 15.4612x^2(1 - x)t(1 - t)^2 + 9.4559x^3t(1 - t)^2 \\ &\quad + 5.6529(1 - x)^3t^2(1 - t) + 16.9589x(1 - x)^2t^2(1 - t) \\ &\quad + 22.0396x^2(1 - x)t^2(1 - t) + 13.3483x^3t^2(1 - t) \\ &\quad + 2.3250(1 - x)^3t^3 + 6.9751x(1 - x)^2t^3 \\ &\quad + 9.0971x^2(1 - x)t^3 + 5.5510x^3t^3. \end{aligned}$$

We perform the method to the problem for various $n = m, s = p = 10$ and $s = p = 20$. The results are given in Tables 1 and 2. As seen from the tables, increasing n gives the decreasing error sequence while the computational time increases. On the other hand, residual correction procedure estimates the error well for $n \neq p$. The corrected solutions are more accurate than the solutions for $n < p$. We give the computational costs of the method in Table 3. Increasing $n = m$ yields more computational costs which means that the method needs more computational times, numbers of multiplications, summations, assignments, and storages for big n, m . As a result, we can say that the numbers n, m are selected as not too big or not too small. It seems to be suitable for selecting $n = m$ around 10 for Example 1.

The graphs of $|e_{12,12}|, |\tilde{e}_{15,15}|$ and $|y - (y_{12,12} + \tilde{e}_{15,15})|$ are plotted in Figure 2 which also depicts the absolute errors of the 5th-order approximate solutions of ADM, HPM, HAM and VIM [1–4]. We also give a comparison for the method with HAM and VIM in Table 4 for different nodes number. It can be seen from Figure 2 and Table 4 that the corrected 2D-BPs approximate solution yields better approximate results than the results obtained by ADM, HPM, HAM, and VIM. Moreover, the residual correction procedure estimates the error more accurately.

Table 1. The norms of absolute errors, the estimation of the errors by using the residual correction procedure, and the errors obtained by the corrected approximate solutions of Example 1 for $s = p = 10$.

$n = m$	3	4	5	6	7
$\ e_{n,n}\ _\infty$	0.7547	0.2304	0.0773	0.0183	4.45×10^{-3}
$\ \tilde{e}_{10,10}\ _\infty$	0.7965	0.2249	0.0739	0.0180	4.53×10^{-3}
$\ e_{n,n} - \tilde{e}_{10,10}\ _\infty$	4.17×10^{-2}	5.46×10^{-3}	3.42×10^{-3}	3.06×10^{-4}	8.09×10^{-5}
Run time (s)	0.032	0.094	0.156	0.359	0.734
n	8	9	10	11	12
$\ e_{n,n}\ _\infty$	9.28×10^{-4}	2.04×10^{-4}	3.94×10^{-5}	7.28×10^{-6}	1.19×10^{-5}
$\ \tilde{e}_{10,10}\ _\infty$	9.18×10^{-4}	1.65×10^{-4}	1.08×10^{-5}	1.69×10^{-6}	6.71×10^{-8}
$\ e_{n,n} - \tilde{e}_{10,10}\ _\infty$	9.76×10^{-6}	3.93×10^{-5}	3.94×10^{-5}	8.97×10^{-6}	1.26×10^{-6}
Run time (s)	1.357	2.543	4.540	7.675	12.293

Table 2. The results for Example 1 and $s = p = 20$.

$n = m$	4	6	8	10	12
$\ e_{n,n}\ _\infty$	0.2304	0.0183	9.28×10^{-4}	3.94×10^{-5}	1.19×10^{-6}
$\ \tilde{e}_{20,20}\ _\infty$	0.2249	0.0180	9.56×10^{-4}	3.82×10^{-5}	1.22×10^{-6}
$\ e_{n,n} - \tilde{e}_{20,20}\ _\infty$	5.42×10^{-3}	2.68×10^{-4}	2.85×10^{-5}	1.22×10^{-6}	2.94×10^{-8}
Run time (s)	0.094	0.359	1.357	4.540	13.448
n	14	16	18	20	22
$\ e_{n,n}\ _\infty$	3.22×10^{-8}	7.45×10^{-10}	1.33×10^{-11}	2.71×10^{-13}	3.39×10^{-15}
$\ \tilde{e}_{20,20}\ _\infty$	3.21×10^{-8}	7.29×10^{-10}	1.39×10^{-11}	7.73×10^{-13}	4.97×10^{-16}
$\ e_{n,n} - \tilde{e}_{20,20}\ _\infty$	1.95×10^{-10}	1.67×10^{-11}	6.18×10^{-13}	2.71×10^{-13}	3.89×10^{-15}
Run time (s)	30.389	71.090	150.509	300.521	534.194

Table 3. The computational costs of the obtaining $\text{Re}(x_i, t_j)$ of the method for $0 \leq i, j \leq n$ in the Maple code for Example 1.

$n = m$	Multiplications	Assignments	Additions	Storage
5	7827	217	3901	217
10	40,955	727	5503	727
15	309,207	1537	184,281	1537
20	851,768	2647	549,717	2647

Table 4. The results of the 2D-BPM, HAM, and VIM for Example 1.

Degree of polynomial	$n = 5$ (degree 10)	8 (degree 16)	10 (degree 20)	15 (degree 30)
2D-BPM	0.0773	9.28×10^{-4}	3.94×10^{-5}	7.45×10^{-10}
max. abs. error				
Run time (s)	0.156	1.357	4.540	71.090
	$m = 2$ (degree 8)	$m = 4$ (degree 16)	$m = 6$ (degree 24)	$m = 8$ (degree 32)
HAM	0.1933	1.22×10^{-3}	1.94×10^{-6}	2.00×10^{-9}
max. abs. error				
Run time (s)	0.188	0.327	0.499	0.717
	$m = 2$ (degree 8)	$m = 4$ (degree 16)	$m = 6$ (degree 24)	$m = 8$ (degree 32)
VIM	0.1933	1.22×10^{-3}	1.94×10^{-6}	1.09×10^{-9}
max. abs. error				
Run time (s)	0.218	0.343	0.483	0.686

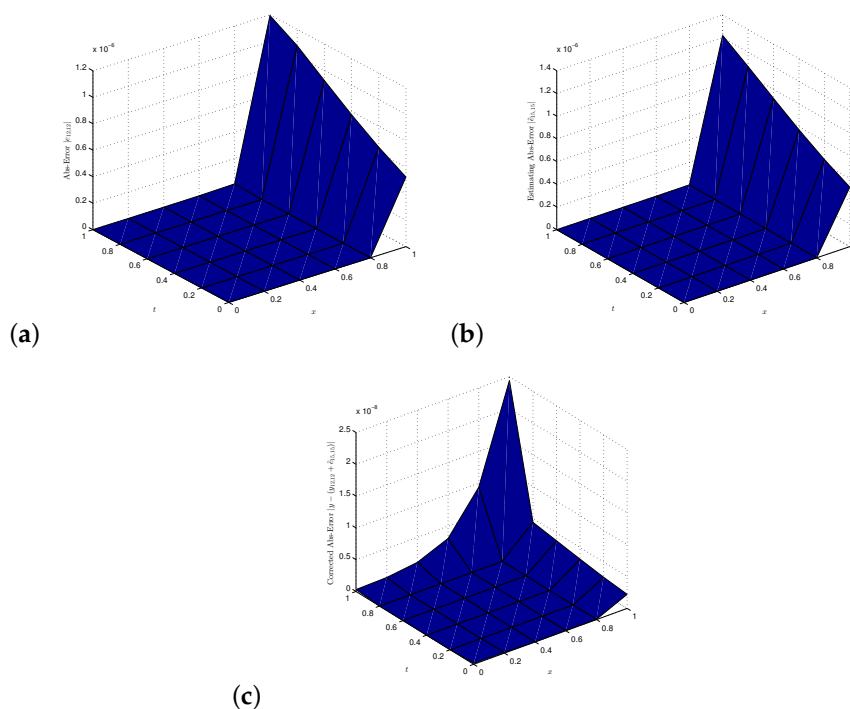


Figure 2. (a) absolute error, (b) estimated absolute error, and (c) corrected absolute error for 2D-BPs solution for Example 1.

5.2. Singular Wave-Type Equations

Example 2. We consider now

$$y_{xx} + \frac{2}{x}y_x - (5 + 4x^2)y = y_{tt} + (12x - 5x^3 - 4x^5),$$

$$y(0, t) = e^{-t}, \quad y_x(0, t) = 0,$$

where the exact solution is

$$y(x, t) = x^3 + e^{x^2-t}. \tag{51}$$

The solution in series form by ADM [1], HAM [2], and HPM [3] is

$$y(x, t) \simeq x^3 + e^{-t} \left(1 + x^2 + \frac{x^4}{2!} + \frac{x^6}{3!} + \frac{x^8}{4!} + \dots \right). \tag{52}$$

We applied the method to the problem for various $m \neq n$. We can say from Table 5 that the norm of the absolute error decreases when m, n increases. The procedure estimates the error well and the corrected solutions are better than the solutions in the norm. The absolute error and corrected absolute error for $s = p = 18$ are shown in Figure 3. We also give the solutions of the problem by ADM, HPM, and HAM [1–3] to make a comparison in the same figure. We can conclude from this figure that the corrected absolute error is better than the absolute error, also better than the absolute error of ADM, HPM, and HAM.

Table 5. The norms of absolute errors, the estimation of the errors by using the residual correction procedure, and the errors obtained by the corrected approximate solutions of Example 2 for $s = 16$ and $p = 18$.

	$m = 4, n = 5$	$m = 6, n = 5$	$m = 6, n = 8$	$m = 7, n = 9$
$\ e_{m,n}\ _\infty$	0.0993	9.38×10^{-3}	7.75×10^{-3}	1.96×10^{-3}
$\ \tilde{e}_{16,18}\ _\infty$	0.0976	7.71×10^{-3}	7.74×10^{-3}	1.96×10^{-3}
$\ e_{m,n} - \tilde{e}_{16,18}\ _\infty$	1.68×10^{-3}	1.68×10^{-3}	6.38×10^{-6}	8.23×10^{-7}
Run time (s)	0.062	0.218	0.624	1.201
	$m = 10, n = 8$	$m = 12, n = 10$	$m = 14, n = 12$	$m = 15, n = 16$
$\ e_{m,n}\ _\infty$	2.29×10^{-5}	6.23×10^{-7}	1.50E-8	2.14×10^{-9}
$\ \tilde{e}_{16,18}\ _\infty$	1.65×10^{-5}	5.25×10^{-7}	1.36×10^{-8}	1.83×10^{-9}
$\ e_{m,n} - \tilde{e}_{16,18}\ _\infty$	6.38×10^{-6}	9.81×10^{-8}	1.42×10^{-9}	3.13×10^{-10}
Run time (s)	2.480	7.051	18.096	50.279

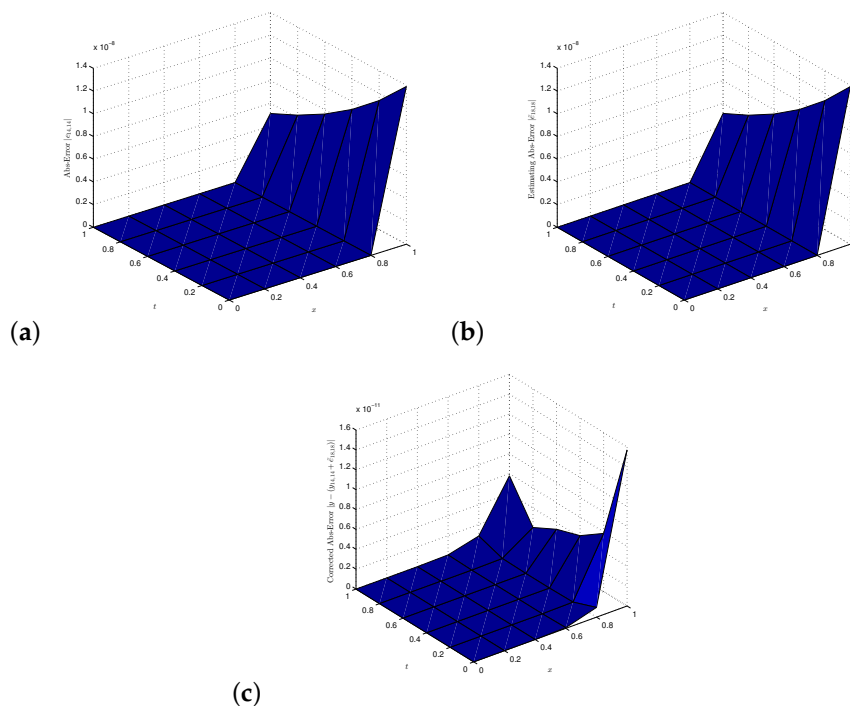


Figure 3. (a) absolute error; (b) estimated absolute error; (c) corrected absolute error for 2D-BPs solution for Example 2.

5.3. Nonlinear Models

Example 3. Consider

$$y_{xx} + \frac{5}{x}y_x + (24t + 16t^2x^2)e^y - 2x^2e^{y/2} = y_t, \tag{53}$$

$$y(0, t) = 0, \quad y_x(0, t) = 0. \tag{54}$$

The exact solution of the problem [1–4] is $y(x, t) = -2 \ln(1 + tx^2)$. We apply the method to the linear problem for various $m = n$ and apply the procedure for $s = p = 9$. The results are given in Tables 6 and 7. As seen from the tables, increasing $m = n$ yields more accurate results, whereas the computational times increase. On the other hand, the proposed method yields a much better approximate solution than the solutions obtained by ADM, HAM, and HPM [1–3], of order 5. We plot the error, the estimation of the error, and the corrected error for $m = n = 8$ and $ms = p = 9$ in Figure 4.

We can say from the figures that the estimation by using the procedure fits the error well and the corrected error is smaller than the error.

Table 6. The norms of absolute errors, the estimation of the errors by using the residual correction procedure, and the errors obtained by the corrected approximate solutions of Example 3 for $s = p = 9$.

$n = m$	5	7	9
$\ e_{n,n}\ _\infty$	2.48×10^{-4}	4.43×10^{-5}	2.66×10^{-6}
$\ \tilde{e}_{9,9}\ _\infty$	2.45×10^{-4}	4.70×10^{-5}	5.65×10^{-12}
$\ e_{n,n} - \tilde{e}_{9,9}\ _\infty$	2.66×10^{-6}	2.66×10^{-6}	2.66×10^{-6}
Run time (s)	1.638	20.155	107.251

Table 7. The results of the 2D-BPM and HAM for Example 3.

Degree of polynomial	$n = 4$ (degree 8)	6 (degree 12)	8 (degree 16)
2D-BPM	1.87×10^{-3}	1.57×10^{-4}	2.86×10^{-5}
max. abs. error			
Run time (s)	0.422	4.680	44.398
HAM	$m = 2$ (degree 8)	$m = 3$ (degree 12)	$m = 6$ (degree 24)
max. abs. error	1.1130	1.3272	3.0520
Run time (s)	1.778	1.950	2.340

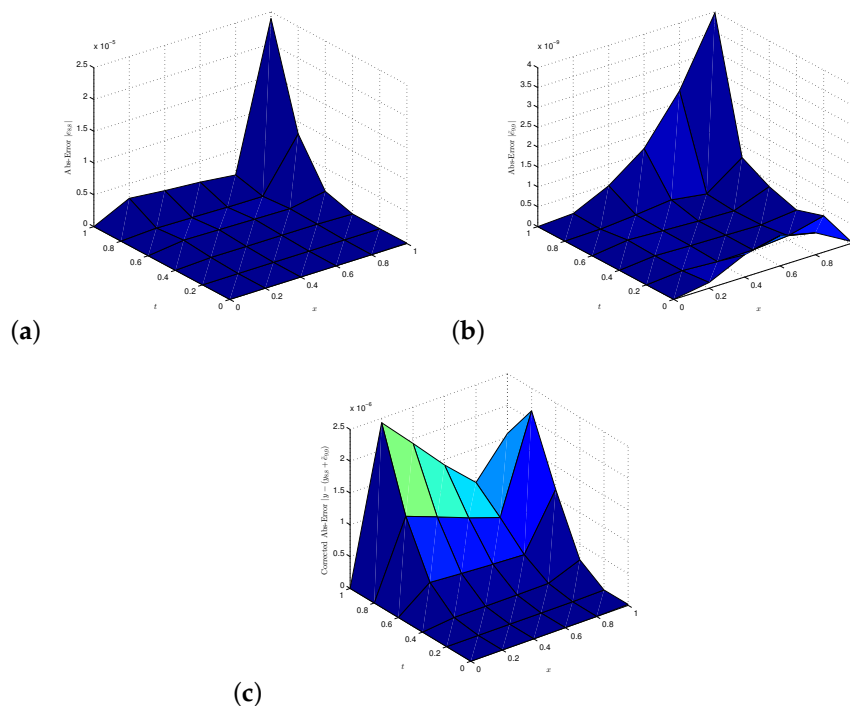


Figure 4. (a) absolute error; (b) estimated absolute error; (c) corrected absolute error for 2D-BPs solution for Example 3.

6. Conclusions

In this study, we have proposed a numerical method to solve time-dependent Emden-Fowler type equations. In the numerical scheme, the 2D-BPs are utilized. The residual correction procedure is applied to estimate the error of the approximate solution. By using the suggested procedure, we obtain a new approximate solution which is more accurate than the 2D-BPs approximate solution. The error analysis as well as convergence of the proposed method have been investigated. We applied the method to several test problems including linear and nonlinear problems and also compared with

some other methods to show the efficiency of the new method. As seen from the results of text examples, increasing the node numbers n, m yields a sequence which converges to the exact solution for all examples in the norm. The computational times and necessary operations in the code increase if n, m increases. Hence, the optimum n, m values are observed around 10. The procedure estimates the error with a good accuracy for $n, m \neq s, p$. In addition, by using the procedure, more accurate results are obtained. We can conclude that the obtained results are consistent with the results of ADM, HPM, HAM, and VIM. Especially for the nonlinear problem, the method gives better approximate solutions with respect to the semi-analytic methods.

Author Contributions: Conceptualization, A.S.B., O.R.I., A.-K.A., M.S. and I.H.; methodology, A.S.B., O.R.I., A.-K.A. and M.S.; software, A.S.B., O.R.I., A.-K.A. and M.S.; validation, A.S.B., O.R.I., A.-K.A., M.S. and I.H.; formal analysis, A.S.B., O.R.I., A.-K.A. and M.S.; investigation, A.S.B., O.R.I., A.-K.A. and M.S.; writing-original draft preparation, A.S.B., O.R.I., A.-K.A. and M.S.; writing-review and editing, A.S.B., O.R.I., A.-K.A., M.S. and I.H.; funding acquisition, I.H.. All authors have read and agreed to the published version of the manuscript

Funding: We are grateful for the financial support received from the Universiti Kebangsaan Malaysia under the research grant GP-2019-K006388.

Conflicts of Interest: All authors declare that they have no conflict of interest.

References

1. Wazwaz, A.M. Analytical solution for the time-dependent Emden-Fowler type of equations by Adomian decomposition method. *Appl. Math. Comput.* **2005**, *166*, 638–651. [[CrossRef](#)]
2. Bataineh, A.S.; Noorani, M.S.M.; Hashim, I. Solutions of time-dependent Emden-Fowler type equations by homotopy analysis method. *Phys. Letts. A* **2007**, *371*, 72–82. [[CrossRef](#)]
3. Chowdhury, M.S.H.; Hashim, I. Solutions of time-dependent Emden-Fowler type equations by homotopy-perturbation method. *Phys. Letts. A* **2007**, *368*, 305–313. [[CrossRef](#)]
4. Batiha, K. Approximate analytical solutions for time-dependent Emden-Fowler-type equations by variational iteration method. *Am. J. Appl. Sci.* **2007**, *4*, 439–443. [[CrossRef](#)]
5. Wazwaz, A.M. Adomian decomposition method for a reliable treatment of the Emden-Fowler equation. *Appl. Math. Comput.* **2005**, *161*, 543–560. [[CrossRef](#)]
6. Bataineh, A.S.; Noorani, M.S.M.; Hashim, I. Homotopy analysis method for singular IVPs of Emden-Fowler type. *Commun. Nonlinear Sci. Numer. Simulat.* **2009**, *14*, 1121–1131. [[CrossRef](#)]
7. Chandrasekhar, S. *An Introduction to the Study of Stellar Structure*; Dover Publications: New York, NY, USA, 1957.
8. Ramos, J.I. Series approach to the Lane–Emden equation and comparison with the homotopy perturbation method. *Chaos Solitons Fractals* **2008**, *38*, 400–408. [[CrossRef](#)]
9. Liao, S.J. The Proposed Homotopy Analysis Technique for the Solution of Nonlinear Problems. Ph.D. Thesis, Shanghai Jiao Tong University, Shanghai, China, 1992.
10. Liao, S.J. *Beyond Perturbation: Introduction to Homotopy Analysis Method*; Chapman and Hall/CRC Press: Boca Raton, FL, USA, 2003; p. 321.
11. Liao, S.J. On the homotopy analysis method for nonlinear problems. *Appl. Math. Comput.* **2004**, *147*, 499–513. [[CrossRef](#)]
12. Liao, S.J. Comparison between the homotopy analysis method and homotopy perturbation method. *Appl. Math. Comput.* **2005**, *169*, 1186–1194. [[CrossRef](#)]
13. He, J.H. A new approach to nonlinear partial differential equation. *Commun. Nonlinear Sci. Numer. Simul.* **1997**, *2*, 230–235. [[CrossRef](#)]
14. He, J.H. Approximate analytical solution for seepage flow with functional derivatives in porous media. *Comput. Methods Appl. Mech. Eng.* **1998**, *167*, 57–68. [[CrossRef](#)]
15. He, J.H. Approximate solution of nonlinear differential equations with convolution product nonlinearities. *Comput. Methods Appl. Mech. Eng.* **1998**, *167*, 69–73. [[CrossRef](#)]
16. He, J.H. Variational iteration method—a kind of nonlinear analytical technique: Some examples. *Int. J. Non-Linear Mech.* **1999**, *34*, 699–708. [[CrossRef](#)]

17. He, J.H. Variational iteration method for autonomous ordinary differential systems. *Appl. Math. Comput.* **2000**, *114*, 115–123. [[CrossRef](#)]
18. He, J.H. Homotopy perturbation method: A new nonlinear analytical technique. *Appl. Math. Comput.* **2003**, *135*, 73–79. [[CrossRef](#)]
19. He, J.H. Asymptotology by homotopy perturbation method. *Appl. Math. Comput.* **2004**, *156*, 591–596. [[CrossRef](#)]
20. He, J.H. The homotopy perturbation method for nonlinear oscillators with discontinuities. *Appl. Math. Comput.* **2004**, *151*, 287–292. [[CrossRef](#)]
21. He, J.H. Application of homotopy perturbation method to nonlinear wave equations. *Chaos Solitons Fractals* **2005**, *26*, 695–700. [[CrossRef](#)]
22. He, J.H. Homotopy perturbation method for bifurcation of nonlinear problems. *Int. J. Nonlinear Sci. Numer. Simul.* **2005**, *6*, 207–208. [[CrossRef](#)]
23. Batiha, B.; Noorani, M.S.N.; Hashim, I. Application of variational iteration method to heat and wave-like equations. *Phys. Letts. A* **2007**, *369*, 55–61. [[CrossRef](#)]
24. Yousefi, S.A.; Behroozifar, M. Operational matrices of Bernstein polynomials and their applications. *Int. J. Syst. Sci.* **2010**, *41*, 709–716. [[CrossRef](#)]
25. Gupta, V.G.; Sharma, P. Solving singular initial value problems of Emden-Fowler and Lane–Emden type. *Int. J. Appl. Math. Comput. Sci.* **2009**, *1*, 206–212.
26. Wazwaz, A.M. Solving two Emden-Fowler type equations of third order by the variational iteration method. *Appl. Math. Inf. Sci.* **2015**, *9*, 2429–2436.
27. Wazwaz, A.-M. A new algorithm for solving differential equations of Lane–Emden type. *Appl. Math. Comput.* **2001**, *118*, 287–310. [[CrossRef](#)]
28. Rach, R.; Wazwaz, A.M.; Duan, J.S. The Volterra integral form of the Lane–Emden equation: New derivations and solution by the Adomian decomposition method. *J. Appl. Math. Comput.* **2015**, *47*, 365–379. [[CrossRef](#)]
29. Dehghan, M.; Shakeri, F. Approximate solution of a differential equation arising in astrophysics using the variational iteration method. *New Astron.* **2008**, *13*, 53–59. [[CrossRef](#)]
30. Yildirim, A.; Ozis, T. Solutions of singular IVPs of Lane–Emden type by the variational iteration method, Nonlinear Anal. *Theory Methods Appl.* **2009**, *70*, 2480–2484. [[CrossRef](#)]
31. Zomot, N.H.; Ababneh, O.Y. Solution of differential equations of Lane–Emden type by combining integral transform and variational iteration method. *Nonlinear Anal. Diff. Equ.* **2016**, *4*, 143–150. [[CrossRef](#)]
32. Ghorbani, A.; Bakherad, M. A variational iteration method for solving nonlinear Lane–Emden problems. *New Astron.* **2017**, *54*, 1–6. [[CrossRef](#)]
33. Rafiq, A.; Hussain, S.; Ahmed, M. General homotopy method for Lane–Emden type differential equations. *Int. J. Appl. Math. Mech.* **2009**, *5*, 75–83.
34. Kumar, N.; Pandey, R. Solution of the Lane–Emden equation using the Bernstein operational matrix of integration. *ISRN Astron. Astrophys.* **2011**, *2011*, 351747. [[CrossRef](#)]
35. Isik, O.R.; Sezer, M. Bernstein series solution of a class of Lane–Emden type equations. *Math. Probl. Eng.* **2013**, *2013*, 423797. [[CrossRef](#)]
36. Bhrawy, A.H.; Alofi, A.S. The operational matrix of fractional integration for shifted Chebyshev polynomials. *Appl. Math. Lett.* **2013**, *26*, 25–31. [[CrossRef](#)]
37. Kashkari, B.S.; Syam, M.I. Fractional-order Legendre operational matrix of fractional integration for solving the Riccati equation with fractional order. *Appl. Math. Comput.* **2016**, *290*, 281–291. [[CrossRef](#)]
38. Pandey, R.K.; Kumar, N. Solution of Lane–Emden type equations using Bernstein operational matrix of differentiation. *New Astron.* **2012**, *17*, 303–308. [[CrossRef](#)]
39. Maleknejad, K.; Basirat, B.; Hashemizadeh, E. A Bernstein operational matrix approach for solving a system of high order linear Volterra-Fredholm integro-differential equations. *Math. Comput. Model.* **2012**, *55*, 1363–1372. [[CrossRef](#)]
40. Yousefi, S.A.; Behroozifar, M.; Dehghan, M. The operational matrices of Bernstein polynomials for solving the parabolic equation subject to specification of the mass. *J. Comput. Appl. Math.* **2011**, *335*, 5272–5283. [[CrossRef](#)]
41. Yousefi, S.; Behroozifar, M.; Dehghan, M. Numerical solution of the nonlinear age-structured population models by using the operational matrices of Bernstein polynomials. *Appl. Math. Model.* **2012**, *36*, 945–963. [[CrossRef](#)]

42. Maleknejad, K.; Hashemizadeh, E.; Basirat, B. Computational method based on Bernstein operational matrices for nonlinear Volterra-Fredholm-Hammerstein integral equations. *Commun. Nonlinear Sci. Numer. Simulat.* **2012**, *17*, 52–61. [[CrossRef](#)]
43. Parand, K.; Hossayni, S.A.; Rad, J.A. Operation matrix method based on Bernstein polynomials for the Riccati differential equation and Volterra population model. *Appl. Math. Model.* **2016**, *40*, 993–1011. [[CrossRef](#)]
44. Bataineh, A.S.; Isik, O.R.; Hashim, I. Bernstein method for the MHD flow and heat transfer of a second grade fluid in a channel with porous wall. *Alex. Eng. J.* **2016**, *55*, 2149–2156. [[CrossRef](#)]
45. Nemati, A. Numerical solution of 2D fractional optimal control problems by the spectral method along with Bernstein operational matrix. *Int. J. Control* **2018**, *91*, 2632–2645. [[CrossRef](#)]
46. Bhatti, M.I.; Bracken, P. Solutions of differential equations in a Bernstein polynomial basis. *J. Comput. Appl. Math.* **2007**, *205*, 272–280. [[CrossRef](#)]
47. Hosseini Shekarabi, F.; Maleknejad, K.; Ezzati, R. Application of two-dimensional Bernstein polynomials for solving mixed Volterra-Fredholm integral equations. *Afr. Mat.* **2015**, *26*, 1237–1251. [[CrossRef](#)]
48. Asgari, M.; Ezzati, R. Using operational matrix of two-dimensional Bernstein polynomials for solving two-dimensional integral equations of fractional order. *Appl. Math. Comput.* **2017**, *307*, 290–298. [[CrossRef](#)]



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).