Available online at http://ijim.srbiau.ac.ir/
Int. J. Industrial Mathematics (ISSN 2008-5621)
Vol. 14, No. 3, 2022 Article ID IJIM-0873, 11 pages
DOI: http://dx.doi.org/10.30495/ijim.2022.17794.873
Research Article

An Improvement on Collocation Algorithm to Solve Initial Value Problems

M. Nikarya *†, S. Sarabadana ‡

Received Date: 2019-05-16 Revised Date: 2019-08-10 Accepted Date: 2021-01-08

Abstract

In this paper an improved version of the collocation method is proposed to solve ordinary differential equations with initial conditions. Our proposed algorithm is described by applying it to some well-known IVPs. The results are compared with basic collocation algorithms to show the advantages, applicability and efficiency of the proposed method. Based on numerical results, the proposed algorithm has better accuracy and execution time.

Keywords : IVPs; Spectral methods; Collocation algorithm; CPU time; Accuracy.

1 Introduction

Many problems arising in science and engineering are defined as IVPs; thus, many researcher in the field of mathematics, numerical analysis and computer science have tried to solve such issues by different methods and algorithms, for example: Adomian decomposition method [8, 27, 30], Homotopy analysis method [1, 9, 25], VIM [3, 14], Tau [12, 6, 7], Pseudo-spectral [5], Hybrid functions [2, 16], etc. The collocation method is one of the high accuracy numerical tools in computational mathematics while it is popular in engineering. [18, 10, 17, 13].

There are some drawbacks such as enough accuracy and computational time in applying the conventional collocation algorithm to some specific problems.

The origin of these issues is the condition number of obtained system of equations from Newton's method, because the condition number of obtained system of equations via spectral methods grows exponentially by increasing the collocation points or number of basis functions. As a result, sometimes, the mathematic software (Matlab, Maple etc.) can not find the solution (Spectral coefficients) of obtained linear system (or obtained linear system derived from nonlinear system of Newton's method).

In order to address these concerns, this paper aims to propose an algorithm through improving the collocation method to solve initial value problems. Guo and Yan [11, 29] introduced same method for Legender-Gauss and performed the

*Corresponding author. m.nikarya@irost.ir, Tel:+98(21)57416314.
†Department of Electrical Engineering and Information Technology, Iranian Research Organization for Science and Technology, Tehran, Iran.
‡Department of Mathematics, Faculty of Science, Imam Hossein Comprehensive University, Tehran, Iran.

error analysis. But they did not apply the method to difficult nonlinear IVPs. In this paper we improve the collocation algorithm utilizing the first kind of Bessel functions [19, 18, 20]. Then, the proposed method is applied to some well-known IVPs in order to discuss its advantages and applicability.

This paper is organized as follows: the basic definition and properties of Bessel function and are presented in Section *1.1*. In Section 2, we have described the spectral collocation methods. Section 3 proposes a new algorithm based on improvement of collocation method to solve IVPs. In Section 4, the applicability, accuracy and reliability of proposed method is investigated by applying it to solve IVPs. the results are compared with common collocation algorithm. The paper concludes in Section 5.

1.1 Introduction of Bessel functions

In this section, the first kind of Bessel function and its properties which will be used to construct the Bessel functions collocation (BFC) method will be described.

The Sturm - Liouville equation of order n of Bessel function, is [26, 4]:

$$x^2y''(x)+xy'(x)+(x^2-n^2)y(x)=0,\quad \text{for}\ \ x\in(-\infty,\infty),\quad (n\in\mathbb{R}). \tag{1.1}$$

This is also called the Bessel equation. An obtained solution of this equation is [4]:

$$\sum_{r=0}^{\infty} a_0\frac{(-1)^r\Gamma(n+1)}{2^{2r}r!\,\Gamma(n+r+1)}(\frac{x}{2})^{2r+n},$$

for each value of a_0; where $\Gamma(\lambda)$ is the gamma function which is defined as follows:

$$\Gamma(\lambda)=\int_0^{\infty} e^{-t}t^{\lambda-1}dt.$$

if we choose $a_0=\frac{1}{2^n\Gamma(n+1)}$, the solution of eq. (1.1) be as follows, which we shall denote it by $J_n(x)$ and call it the Bessel function of the first kind of order n:

$$J_n(x)=\sum_{r=0}^{\infty}\frac{(-1)^r}{r!\,\Gamma(n+r+1)}(\frac{x}{2})^{2r+n}, \tag{1.2}$$

the series (1.2) is convergent for all $-\infty<x<\infty$. Some relationships between $J_n(x)$ and its derivative are as follows [4]:

$$\frac{d}{dx}(x^nJ_n(x))=x^nJ_{n-1}(x),$$
$$J_n'(x)=J_{n-1}(x)-\frac{n}{x}J_n(x),$$
$$J_n'(x)=\frac{n}{x}J_n(x)-J_{n+1}(x).$$

2 Spectral methods and WRMs

Spectral methods, in the of numerical and industrial mathematics to solve differential equations, generally belong to the big family of weighted residual methods (WRMs) [24]. WRMs present the particular classes of approximation techniques, that in all of them, the aim is the residual functions (or errors) be minimized in a certain method and thereby leading to specific methods like collocation, Galerkin, Petrov-Galerkin and tau etc. WRMs are considered as the base and cornerstone of many famous methods such as spectral methods, finite element, boundary element, finite volume etc. In this section, WRMs, spectral methods and one of the best and simplest sub-methods of it that named collocation method are described [5, 24]. Before everything, first, we introduce the WRM, briefly. Consider the following problem:

$$\mathcal{L}u(x)+\mathcal{N}u(x)=f(x)\quad ,\quad x\in\Omega, \tag{2.3}$$

where $\mathcal{L}$ is the integral or/and differential operators, $\mathcal{N}$ is a lower-order linear and/or nonlinear operators involving u and/or its derivatives (if exist) and $f(x)$ is a linear or nonlinear function. There are also enough initial conditions.
The first step of the WRM is approximating the unknown solution u by following finite sum:

$$u(x)\approx u_N(x)=\sum_{i=0}^{N}a_i\phi_i(x)\ ,\ x\in\Omega, \tag{2.4}$$

where $\phi_i(x)$ is the *basis functions*, and the unknown expansion coefficients must be determined. This finding the unknown coefficients is main duty of WRMs. Substituting u with u_N in (2.3) the residual function is obtained as follows:

$$\mathcal{R}_N(x)=\mathcal{L}u_N(x)+\mathcal{N}u_N(x)-f(x). \tag{2.5}$$

The notion of the WRM is to force the residual to zero by requiring:

$$< \mathcal{R}_N, \psi >_\omega = 0 \quad \Rightarrow \int_\Omega \mathcal{R}_N(x)\psi_j(x)\omega(x)dx = 0 \ , \quad 0 \le j \le N, \tag{2.6}$$

where $\{\psi_j(x)\}$ are *test functions*, and ω is positive weight function. The choice of test functions results to a kind of the spectral methods, for example:

- Collocation method: the test functions in (2.6) are Lagrange basis polynomials, such that $\psi_j(x_k) = \delta_{jk}$, where $\{x_k\}$ are preassigned collocation points. Hence the residual is forced to zero at x_j,i.e., $R_N(x_j) = 0$, [24].

2.1 Function approximation

We now explain how to approximate the unknown solution of a problem based on a finite expansion of known functions. Based on features of Bessel function, $\{J_n(x)\}_{n=0}^{\infty}$, is a basis for Hilbert space $\mathcal{H} = L^2(\Gamma)$, where $\Gamma = (-\infty, +\infty)$. This means $\{J_0(x), J_1(x), ..., J_n(x)\} \subset \mathcal{H}$. Now, we define the generated space by these functions as follows

$$\mathbf{J} = \text{span}\{J_0(x), J_1(x), ..., J_n(x)\}, \tag{2.7}$$

$\mathbf{J}$ is the finite-dimensional subspace of $\mathcal{H} = L^2(\Gamma)$, dim $\mathbf{J} = n+1$, so $\mathbf{J}$ is a closed subspace of $\mathcal{H}$, therefore, $\mathbf{J}$ is a complete subspace of $\mathcal{H}$. For each arbitrary element f in $\mathcal{H}$, we have an unique best approximation $\widehat{j} \in \mathbf{J}$, that:

$$\exists\, \widehat{j} \ \in \ \mathbf{J};\ \forall\ j \ \in \ \mathbf{J},\ \|f - \widehat{j}\| \le \|f - j\| \ , \tag{2.8}$$

where $\|f\| = < f, f >^{1/2}$ and $< f, g >= \int_{-\infty}^{\infty} f(t)g(t)\ dt$.

Definition 2.1. *(Direct sum ($\oplus$)): suppose that Y and Z are two subspaces of $\mathcal{H}$, we can write $\mathcal{H} = Y \oplus Z$, if for each $x \in \mathcal{H}$ there exist a unique representation $x = y + z$ that $y \in Y$ and $z \in Z$. Then, Z is called complement of Y in $\mathcal{H}$ and viceversa.*

Definition 2.2. *Let $\mathcal{H}$ be an Hilbert space and Y be any closed subspace of $\mathcal{H}$. $Y^{\perp}$ is defined the* ***orthogonal complement****, as:*

$$Y^{\perp} = \{z \in \mathcal{H} |\ z \perp Y\} \ . \tag{2.9}$$

Now, by using (2.7) and (2.8), we can say $\mathcal{H} = \mathbf{J} \oplus Z$,
where $Z = \mathbf{J}^{\perp}$, so that for each $x \in \mathcal{H}$, $x = j + z$.
where $z = x - j \perp j$, hence, $< x - j, j >= 0$. We have $j \in \mathbf{J}$, therefore,

$$j = \sum_{k=0}^{n} a_k\ J_k(x) \ , \tag{2.10}$$

and $x - j \perp j$ gives the n conditions

$$< J_m(x), x - j >=< J_m(x), x - \sum_{k=0}^{n} a_k\ J_k(x) >= 0, \tag{2.11}$$

that is

$$< J_m(x), x >= \sum_{k=0}^{n} \bar{a_k}\ \ < J_m(x), J_k(x) >, \quad m = 0, 1, ..., n \ . \tag{2.12}$$

This is a nonhomogeneous system of $n+1$ linear equations in $n+1$ unknown coefficients $\{\bar{a_k}\}_{k=0}^{n}$ (spectral coefficients). The determinant of the coefficients is

$$G(J_0(x), J_1(x), ..., J_n(x)) = \begin{vmatrix} < J_0(x), J_0(x) > & \dots & < J_0(x), J_n(x) > \\ < J_1(x), J_0(x) > & \dots & < J_1(x), J_n(x) > \\ \vdots & \ddots & \vdots \\ < J_n(x), J_0(x) > & \dots & < J_n(x), J_n(x) > \end{vmatrix}$$

Since $\mathbf{J}$ exists and is unique, that system has a unique solution. Hence, $G(J_0(x), J_1(x)$ $, ..., J_n(x))$ must be different from 0. The determinant $G(J_0(x), J_1(x), ..., J_n(x))$ is called the Gram determinant of $J_0(x), J_1(x), ..., J_n(x)$.

Theorem 2.1. *Suppose that $\mathcal{H}$ is a Hilbert space and Y a closed subspace of $\mathcal{H}$ such that* dim$Y < \infty$ *and $\{y_1, y_2, ..., y_n\}$ is any basis for Y. Let x be an arbitrary element in $\mathcal{H}$ and y_0 be the unique best approximation to x from Y. Then,*

$$\|x - y_0\|^2 = \frac{G(x, y_1, y_2, ..., y_n)}{G(y_1, y_2, ..., y_n)} , \qquad (2.13)$$

Proof. . See [15]. □

2.2 Collocation algorithm

A method for forcing the residual function (2.5) to zero, is the collocation algorithm. In this method, by substituting the finite series (2.4) into residual function (2.5) and collocating it on $\{x_k\}$, we have $N+1$ equations and $N+1$ unknown coefficients (spectral coefficients). In all of spectral methods, the purpose is, finding these coefficients.
In the following algorithm, we aim to solve equation (2.3):

BEGIN

1. Set N.
2. Insert the constructed series (2.4), into equation (2.3).
3. Make the Residual function as follows:

$$\mathcal{R}es(x; a_0, a_1, ..., a_n) = \mathcal{L}u_N(x) + \mathcal{N}u_N(x) - f(x).$$

4. Substitute the conditions of the problem into set of equations (or change the basis function to satisfy the conditions). Now, we have $N+1$ unknown coefficients $\{a_n\}_{n=0}^{N}$. Therefor, we need $N+1$.
5. By choosing $N+1$ points $\{x_i\}$, $i = 0, 1, ..., N$, in the domain of the equation (2.3) as collocation points and collocate $Res(x; a_0, a_1, ..., a_n) = \mathcal{L}u_N(x) - f(x)$ in these points, we will have a nonlinear system containing $N+1$ equation.
6. By solving this obtained system of nonlinear equations, we gain the a_n, $n = 0, 1, ..., N$.
7. By substituting the obtained values of these coefficients in (2.4), we obtain approximated solution $u_N(x)$ of $u(x)$.

END.

In step 7, a computer software (Maple, Matlab, etc.) can solve a linear or nonlinear system of equations, but if the nonlinear order of the system of equations, or, the number of equations be large, then the computer encounters a problem and can't solve this system of equations in a reasonable time. While, in Spectral methods, specifically, collocation method, to increase the accuracy, we have to increase the collocation points and degree of series (2.4) (in other words: by increasing N). But, the larger value of N results to higher order of nonlinearity. Therefore, the Maple or Matlab can't solve the obtained system of equations. This problem occurs more often in nonlinear fractional equations, nonlinear ODEs, PDEs and IDEs.

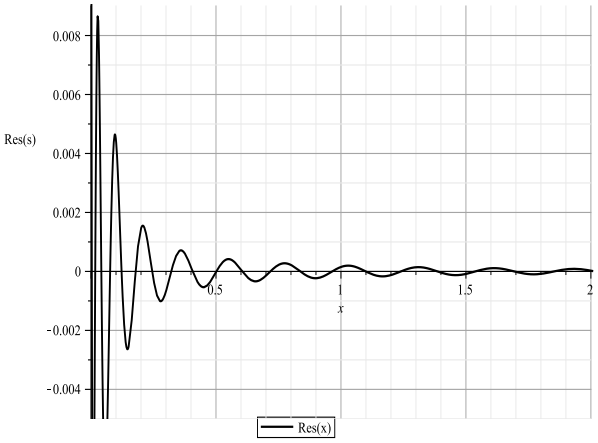

Figure 1: The obtained graph of residuals function for Volterra's population by common collocation method for $\kappa = 0.02$ and $N = 48$.

3 Improved collocation method to solve IVPs

In Section 2.2, the collocation method was discussed. In the spectral methods, in order to increase the accuracy of the approximation, N must be increased. Now, we want introduce a strategy to surmount this problem of IVPs. The following algorithm is presented accordingly: :

BEGIN

Table 1: The obtained results of common collocation algorithm for solving Volterra's population with $\kappa = 0.5$.

N	CPU time$_{\text{sec}}$	Max (Res(x))	Error in $u_{max}(x)$
20	1.716	$4.4588500e-5$	$1.0107763e-6$
30	5.975	$1.1234182e-7$	$7.388164e-10$
40	14.820	$9.649933e-10$	$8.326697e-13$
50	32.573	$1.054115e-12$	$2.032415e-15$

Table 2: The obtained results of presented algorithm for solving Volterra's population with $\kappa = 0.5$, $b_0 = 2$

N_0	N_1	CPU time$_{\text{sec}}$	Max (Res(x))	Error in $u_{max}(x)$
13	11	0.749	$2.794123e-7$	$6.026950e-8$
20	16	2.371	$3.554891e-10$	$4.894438e-12$
28	24	8.736	$1.851814e-15$	$3.703856e-17$
35	30	14.305	$7.623663e-18$	$4.105005e-20$

Table 3: The obtained results of common collocation algorithm for solving Volterra's population with $\kappa = 0.02$.

N	CPU time$_{\text{sec}}$	Max (Res(x))	Error in $u_{max}(x)$
30	24.325	$1.1234182e-1$	$7.388164e-2$
40	39.453	$7.1536839e-2$	$3.691622e-3$
45	can't solve	———	———

Table 4: The obtained results of presented algorithm for solving Volterra's population with $\kappa = 0.02$, $b_0 = 0.15$

N_0	N_1	CPU time$_{\text{sec}}$	Max (Res(x))	Error in $u_{max}(x)$
18	30	18.749	$4.244761e-6$	$4.736303e-6$
21	35	22.086	$9.882021e-7$	$1.498205e-8$
25	38	24.674	$3.056482e-8$	$1.8259794e-10$

Table 5: The obtained results of presented algorithm for solving Volterra's population with $\kappa = 0.02$. and $b_0 = 0.15$, $b_1 = 1$

N_0	N_1	N_3	CPU time$_{\text{sec}}$	Max (Res(x))	Error in $u_{max}(x)$
30	35	15	18.129	$2.4687e-11$	$3.0453e-13$
33	38	18	21.235	$5.3654e-13$	$2.3694e-15$

Table 6: The obtained results of common collocation algorithm for solving Lane-Emden standard equation with $m = 4$

N	CPU time$_{\text{sec}}$	Max (Res(x))
40	823.919	$2.43409e-3$
50	can't solve	———

1. Set $m = 0$.
2. Input b_m (end point of subinterval) and N_m.
3. Set the interval $[a, b_m)$. As a is an initial point of initial value problem, and the amount of the solution is given in a (the initial condition).

Table 7: The obtained results of presented algorithm for solving Lane-Emden standard equation with $m = 4.$ and $b_0 = 3,\ b_1 = 9$

N_0	N_1	N_3	CPU time$_{\text{sec}}$	Max (Res(x))
20	20	15	46.672	$1.079851e-10$

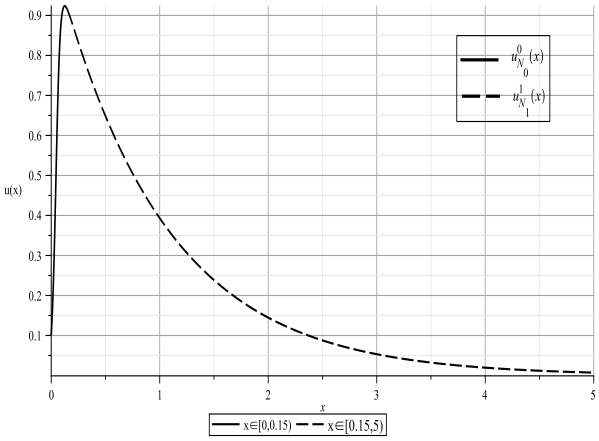

Figure 2: The graph of volterra's population for $\kappa = 0.02$ in two subintervals and and $N_0 = 20,\ N_1 = 30.$

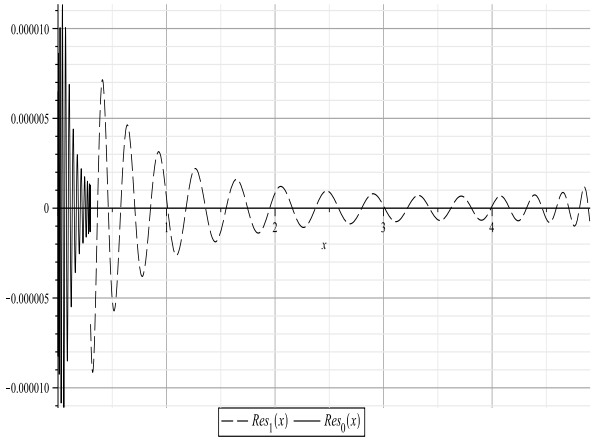

Figure 3: The graph of residual functions via present algorithm of volterra's population for $\kappa = 0.02$ in two subintervals and and $N_0 = 18,\ N_1 = 30.$

4. Construct the following series from (2.4):

$$u^m_{N_m} = \sum_{i=0}^{N_m} a_i J_i(x), \qquad (3.14)$$

where the $\{J_i(x)\}_{i=0}^{N_m}$ are the Bessel functions of the first kind.

5. By multiplying and adding adequate terms to (3.14), satisfy the initial conditions in point a.

6. Insert the constructed series of step 4, into equation (2.3).

7. Construct the Residual function as follows:

$$\mathcal{R}es_m(x; a_0, a_1, ..., a_n) = \mathcal{L}u^m_{N_m}(x) + \mathcal{N}u^m_{N_m}(x) - f(x).$$

Now, we have N_m+1 unknowns $\{a_n\}_{n=0}^{N_m}$. To obtain these unknown coefficients, we need $N_m + 1$ equations, thus:

8. By choosing $N_m + 1$ points $\{x_i\},\ i = 0, 1, ..., N_m,$ in the subinterval $[a, b_m)$, as collocation points and substituting them in $\mathcal{R}es_m(x; a_0, a_1, ..., a_{N_m}) = \mathcal{L}u^m_{N_m} + \mathcal{N}u^m_{N_m} - f(x)$, we construct a system containing N_m+1 equations.

9. By solving the obtained system of equations we gain the $a_n,\ n = 0, 1, ..., N_m.$

10. Substitute the obtained values of these coefficients in (3.14), we shall approach $u(x)$ by $u^m_{N_m}(x)$ in subinterval $[a, b_m)$.

11. Obtain the required value as conditions of problem in point b_m by $u^m_{N_m}(x)$.

12. $a \leftarrow a + b_m$.

13. Now, if b_m or m is large enough or termination condition is met, go to next step; *Else*, $m = m + 1$ and *Go* to step 2.

14. $u_N(x) = \begin{cases} u^0_{N_0} & a \le x < b_0 \\ u^1_{N_1} & b_0 \le x < b_1 \\ u^2_{N_2} & b_1 \le x < b_2 \\ \vdots & \vdots \\ u^m_{N_m} & b_{m-1} \le x < b_m \end{cases}$

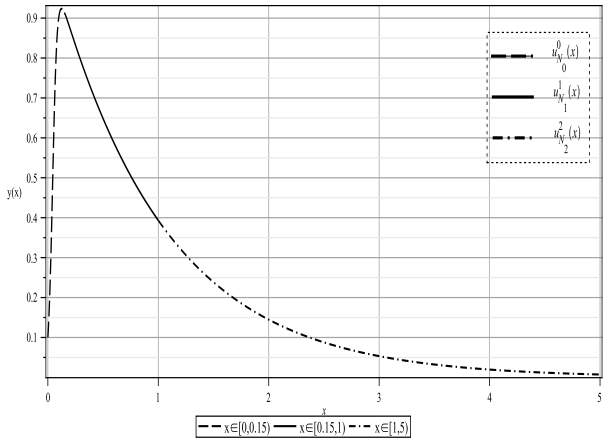

Figure 4: The graph of piecewise solution of volterra's population for $\kappa = 0.02$ in 3 subintervals and $N_0 = 30,\ N_1 = 35,\ N_2 = 15$

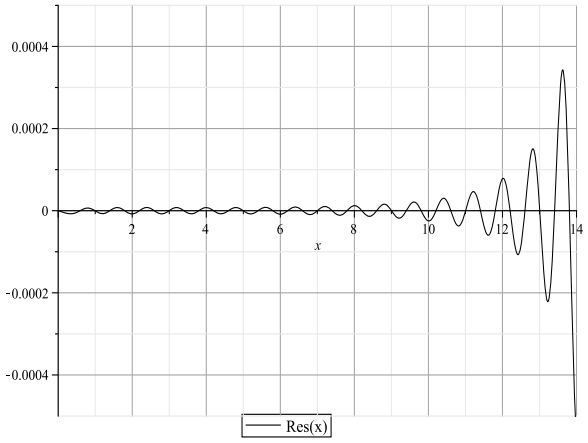

Figure 6: The graph of obtained residuals for standard Lane-Emden by common collocation method for $m = 4$ and $N = 40$.

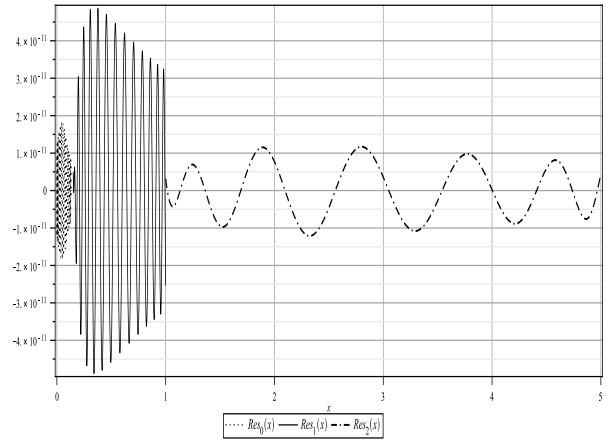

Figure 5: The graph of obtained residuals for volterra's population for $\kappa = 0.02$ in three subintervals and and $N_0 = 30,\ N_1 = 35,\ N_2 = 15$.

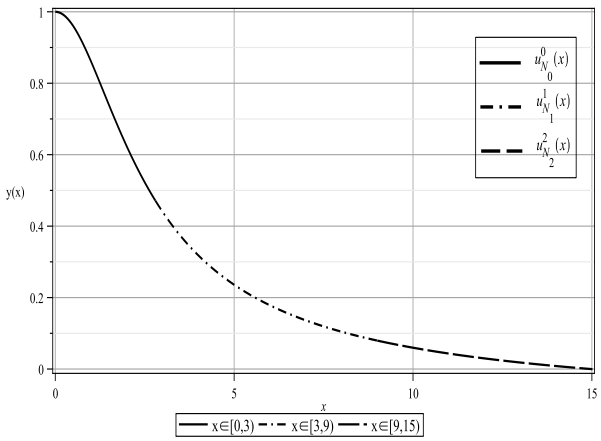

Figure 7: The graph of standard Lane-Emden for $m = 4$ in three subintervals and and $N_0 = 18,\ N_1 = 17,\ N_2 = 14$.

End.

It is clear that in the last interval $[b_m, \infty)$ the problem can be solved by collocation algorithm or other spectral methods.
The description of the mentioned algorithm:

In step 1 and 2, we input the N_m (number of basis functions and collocation points in subinterval m) and b_m (is greater than a and endpoint of subinterval). In step 3, we want to solve problem (2.3), in the subinterval $[a, b_m]$; in step 4, we construct series (3.14), and, in step 5, by multiplying and adding operations, we aim to satisfy the initial conditions. In step 6 and 7, the Residual function is constructed and, in step 8, by choosing $N_m + 1$ collocation points in the subinterval $[a, b_m]$, we construct a set of N_m+1 equations and $N_m + 1$ unknowns. In step 9 and 10, we solve the constructed system of equations, and obtain the unknown coefficients and substitute these coefficients in series of step 4. In step 11, we have an approximate solution $u^m_{N_m}$ of equation (2.3) in the subinterval $[a, b_m]$ for any m. Therefore, by using this solution, we obtain the required values in point b_m, as initial conditions of equation (2.3) for subinterval $[b_m, b_{m+1}]$. Now we repeat previous tasks for next subinterval $[b_m, b_{m+1}]$, so that, in step 12, by substituting b_m to a, the initial point of problem becomes b_m. Step 13 contains a con-

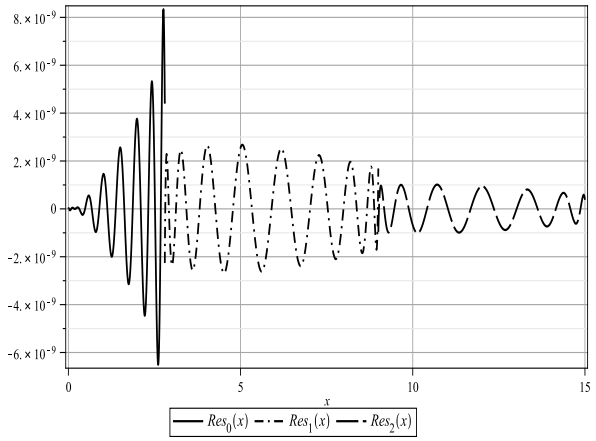

Figure 8: The graph of obtained residuals for standard Lane-Emden for $m = 4$ in three subintervals and and $N_0 = 18$, $N_1 = 17$, $N_2 = 14$.

dition for ending or continuing the loop for next subintervals. In step 14, we construct a piecewise function by bringing together the obtained approximate solutions of all subintervals.

3.1 The advantages of this algorithm

As mentioned before, one of the drawbacks of collocation method is the need to increase N for improving the accuracy. Increasing the N, results in a large computational time. But, by utilizing the proposed algorithm in Sec. 3 we can solve the problem in separated subintervals. Since any interval m have a separate N_m, by increasing N_m, arbitrary level of accuracy can be achieved. The key point is that by using proposed algorithm, m systems of equations with lower order are solved, instead of solving a system of equations with large order. Therefore, the required computational time is reduced.

4 Solving some examples in order to compare these two algorithms

Now, in this section we shall solve some initial value equations in order to show the advantages of the proposed algorithm in terms of accuracy and execution time compared to conventional collocation methods.

```
> restart : Digits := 80 :
  with(Optimization) : with(orthopoly) : with(plots) :
  k := 0 : A := 0 : B := 0.1 : t := 0 : b_t := 0 : Φ := 0 :
  Residual := 0 : Ctime := 0 :
> n_k := scanf('%d')[1] :
  E := A + B·(x − b_t) :
  a := array(0..n_k) :
  for i from 0 to n_k − 1 do
      E := E + a_i·(x − b_t)^2·BesselJ(i, x) :
  od:
  dE := diff(E, x) : ddE := diff(dE, x) :
  Eq := 0.02·ddE − dE + dE^2 + E·dE :
  y := scanf('%f')[1] :
  C := fsolve(T(n_k, x/y − 1), x) :
  for i from 0 to n_k − 1 do
  BZ[i + 1] := C[i + 1] + b_t :
  od:
  for i from 0 to n_k − 1 do
      Eqq[i] := subs(x = BZ[i + 1], Eq);
  od:
  T1 := time( ) :
  X := fsolve({seq(Eqq[m], m = 0..n_k − 1)}, {seq(a[i] = 0, i = 0..n_k − 1)}) :
  T2 := time( ) :
  Ctime := (T2 − T1) + Ctime :
  for i from 0 to n_k − 1 do
  a[i] := subs(X, a[i]);
  od:
  RES_k := Eq : U_k := E :
  t := t + 1 :
  b_t := scanf('%f')[1] : k := k + 1 :
  A := evalf(subs(x = b_t, E)) : B := evalf(subs(x = b_t, dE)) :
  E := "" : dE := "" : ddE := "" : a := "" : Eq := "" : Eqq := "" : X := "" :
  Φ := piecewise(b_t−1 ≤ x ≤ b_t, diff(U_k−1, x), Φ) :
  Residual := piecewise(b_t−1 ≤ x ≤ b_t, RES_k−1, Residual) :
  plot(diff(Φ, x), x = b_0..b_t);
  plot(Residual, x = b_0..b_t);
```

Figure 9: Code of maple for solving example 4.1 for $\kappa = 0.02$.

Example 4.1. *Volterra's Population Model is*

$$\kappa u'(x) = u(x) - u(x)^2 - u(x)\int_0^x u(t)dt,$$
$$u(0) = 0.1 \ .$$

Let,

$$y(x) = \int_0^x u(t)dt,$$

then:

$$y'(x) = u(x) \ ,$$
$$y''(x) = u'(x) \ .$$

The ODE model of population growth becomes [20, 21, 22]:

$$\kappa y''(x) = y'(x) - (y'(x))^2 - y(x)y'(x), \quad (4.15)$$

with initial conditions:

$$y(0) = 0\ , \quad y'(0) = 0.1. \tag{4.16}$$

The u_{max} is presented in [20, 21, 22]:

$$u_{max} = 1 + \kappa \ln(\frac{\kappa}{1 + \kappa + 0.1}) \tag{4.17}$$

We solve equation (4.15) by applying the common collocation method and proposed algorithm in $[0, 5]$ for $\kappa = 0.5$ and 0.02; then, we compare their results. Table. 1 shows the results of the common collocation method for solving equation (4.15) with $\kappa = 0.02$ and Table 2 shows the results of the proposed algorithm for solving this equation, in two sub intervals with $b_0 = 2$. In Tables **??**, 4 and 5, we have shown the obtained results of the common collocation algorithm and the proposed algorithm for two and three sub intervals, respectively, and we have shown the CPU times$_{sec}$, maximum residual in all intervals and the error in u_{max}. Also, in Fig. 1 the Residual function of common collocation method for $\kappa = 0.02$ and $N = 48$ is shown, in Fig. 2 and 4, we have shown the solutions of this example in two and three subintervals, respectively. In Fig. 3 and 5, the residual functions of proposed method are shown, in two and three subintervals, respectively.

Example 4.2. *Lane-Emden standard equation[17, 28]*

$$y'' + \frac{2}{x}y' + y^m = 0, \tag{4.18}$$

with initial conditions:

$$y(0) = 1,$$

$$y'(0) = 0.$$

Now, we apply the common collocation algorithm, and the proposed algorithm for solving this example in three subintervals, and compare their results. Table 6 and 7 show the results of solving this example by using collocation algorithm and the proposed algorithm, respectively. Fig. 6 shows obtained residual function of common collocation method. Also, in Fig. 7 and 8, we have shown the solutions and residual functions obtained from the proposed algorithm for three subintervals.

5 Conclusions

To increase the accuracy in the collocation method (in general: Spectral methods), the N must be increased. But this causes to increase the CPU time, rapidly. To resolve these challenge, a novel algorithm was proposed. In our algorithm, we fragmented the interval of a problem, and, in each interval, we forced an equation with new initial conditions. This strategy showed several advantages. Firstly, in each interval, we have a distinct N (number of collocation points and the degree of basis functions). Each interval has a variable length that gives the advantage to utilize the different basis functions and collocation points in each interval. Therefore, we have some degrees of freedom to solve the problems easily, without being concerned about solving set of equations. Rate of accuracy in tables 2, 4, 5 and 7 show the rate of convergency of proposed method. The open problems of this algorithm are selection of optimum m and b, and knowledge about type of basis functions. In order to evaluate the proposed algorithm, some well-known IVPs were discussed. Finally, we propose four questions to be answered in future studies: 1) is this algorithm capable to solve the equation with boundary conditions? 2) Is this algorithm able to solve the equation with boundary conditions in the infinite? 3) Can this algorithm solve the partial differential equations? And 4) is this algorithm capable to solve the fractional equations?

Acknowledgments

The corresponding author would like to thank Iranian Research Organization for Science and Technology (IROST) for the awarded grant.

References

[1] S. Abbasbandy, T. Hayat, Solution of the MHD Falkner-Skan flow by Homotopy analysis method, *Communications in Nonlinear Science and Numerical Simulation* 14 (2009) 3591-3598.

[2] B. Abdollahi, E. Babolian, Theory of block-pulse functions in numerical solution of Fred-

holm integral equations of the second kind, *International Journal of Industrial Mathematics* 8 (2016) 157- 163.

[3] V. K. Baranwal, R. K. Pandey, M. P. Tripathi, O. Singh, An analytic algorithm for time fractional nonlinear reaction-diffusion equation based on anew iterative method, *Communications in Nonlinear Science and Numerical Simulation* 17 (2012) 3906-3921.

[4] W. W. Bell, Special Functions For Scientists And Engineers, Published simultaneously, D. Van Nostrand Company, *Canada, Ltd.* 1967.

[5] J. P. Boyd, Chebyshev and Fourier spectral methods. 2nd ed, *New York Dover*, 2000.

[6] M. Dehghan, M. Abbaszadeh, A. Mohebbi, Legendre spectral element method for solving time fractional modified anomalous subdiffusion equation, *Applied Mathematical Modelling* 40 (2015) 3635-36541.

[7] M. Dehghan, M. Abbaszadeh, Two meshless procedures: moving kriging interpolation and element-free Galerkin for fractional PDEs, *Applicable Analysis* 1 (2016) 31-4.

[8] M. Dehghan, J. Manafian, A. Saadatmandi, Application of semi-analytical methods for solving the Rosenau-Hyman equation arising in the pattern formation in liquid drops, *International Journal of Numerical Methods for Heat and Fluid Flow* 22 (2012) 777-790.

[9] M. Dehghan, R. Salehi, The use of Homotopy analysis method to solve the time-dependent nonlinear Eikonal partial differential equation, *Zeitschrift fur Naturforschung A* 66 (2011) 259-271.

[10] F. Fattahzadeh, Numerical solution of general nonlinear Fredholm-Volterra integral equations using Chebyshev approximation, *International Journal of Industrial Mathematics* 8 (2016) 81-86.

[11] B. Y. Guo, J. P. Yan, Legendre Gauss collocation method for initial value problems of second order ordinary differential equations, *Applied Numerical Mathematics* 56 (2009) 13861408.

[12] S. A. Hossayni, J. A. Rad, K. Parand, S. Abbasbandy, Application of the exact operational matrices for solving the emden-fowler equations, arising in astrophysics, *International Journal of Industrial Mathematics* 7(2015) 351-374.

[13] S. G. Hosseini, E. Babolian, S. Abbasbandy, A new algorithm for solving Van Der Pol equation based on piecewise spectral Adomian decomposition method, *International Journal of Industrial Mathematics* 8 (2016) 177-184.

[14] S. A. Khuri, A. Sayfy, A Laplace variational iteration strategy for the solution of differential equations, *Applied Mathematics Letters* 25 (2012) 2298-2305.

[15] E. Kreyszig, Introductory functional analysis with applications, *John Wiley, New York*, 2000.

[16] H. R. Marzban, S. Hoseini, M. Razzaghi, Solution of Volterras population model via block-pulse functions and Lagrange-interpolating polynomials, *Mathematical Methods in the Applied Sciences* 32 (2009) 127- 134.

[17] K. Parand, M. Nikarya, J. A. Rad, Solving non-linear Lane-Emden type equations using Bessel orthogonal functions collocation method, *Celestial Mechanics and Dynamical Astronomy* 116 (2013) 97-107.

[18] K. Parand, M. Nikarya, New numerical method based on generalized Bessel function to solve nonlinear Abel fractional differential equation of the first kind. Journal, *Nonlinear Engineering* 8 (2019) 438-448

[19] K. Parand, M. Nikarya, Application of Bessel functions for solving differential and integro-differential equations of the fractional order, *Applied Mathematical Modelling* 38 (2014) 4137-4147.

[20] K. Parand, J. A. Rad, M. Nikarya, A new numerical algorithm based on the first kind

of modified Bessel function to solve population growth in a closed system, *International Journal of Computer Mathematics* 91 (2014) 1239-1254.

[21] K. Parand, M. Nikarya, Application of Bessel functions and jacobian free Newton method to solve time-fractional burger equation, *Nonlinear Engineering* 8 (2019) 688-694.

[22] K. Parand, S. A. Hossayni, J. A. Rad, Operation matrix method based on Bernstein polynomials for the Riccati differential equation and Volterra population model, *Applied Mathematical Modelling* 15 (2016) 993-1011.

[23] K. Parand, M. Nikarya, A novel method to solve nonlinear Klein-Gordon equation arising in quantum field theory based on Bessel functions and Jacobian free Newton-Krylov subspace methods, *Communications in Theoretical Physics* 69 (2018) 637.

[24] J. Shen, T. Tang, L. L. Wang, Spectral Methods Algorithms, Analysics and Applicattions, first edition, *Springer, Berlin*, 2001.

[25] H. Vosughi, E. Shivanian, S. Abbasbandy, A new analytical technique to solve Volterra's integral equations, *Mathematical Methods in The Applied Sciences* 34 (2011) 1243-1253.

[26] G. Watson, A treatise on the theory of Bessel Functions, 2nd edition, Cambridge University Press, Cambridge (England), 1967.

[27] M. Wazwaz, A reliable study for extensions of the Bratu problem with boundary conditions, *Mathematical Methods in The Applied Sciences* 35 (2012) 845-856.

[28] A. M. Wazwaz, A new algorithm for solving differential equations of Lane-Emden type, *Applied Mathematics and Computation* 118 (2001) 287-310.

[29] J. P. Yan, B. Y. Guo, A collocation method for initial value problems of second-order odes by using Laguerre functions, *Numerical Mathematics: Theory, Methods and Applications* 4 (2011) 283-295.

[30] Y. Zhang, K. T. Chong, N. Kazantzis, A. G. Parlos, Discretization of nonlinear input-driven dynamical systems using the Adomian Decomposition Method, *Applied Mathematical Modelling* 36 (2012) 5856-5875.

Mehran Nikarya received the M.Sc. Computer Science from Shahid Beheshti University, Tehran, Iran in 2014 and the Ph. D in Computer Science from Shahid Beheshti University, Tehran, Iran in 2020. Since 2021. He has been a faculty member of Electrical Engineering and Information Technology department of Iranian Research Organization for Science and Technology (IROST), Tehran, Iran. His research interests are Data sciences, Machin learning, Non-linear Optimization, Numerical algorithms, Spectral Methods, Differential equations, Matrix computations and digital signal processing.

Saeed Sarabadan received his B.Sc. degree in Mathematics from Kashan University, in 1997, Kerman, Iran, the M.Sc. Mathematics in Kerman Shahid Bahonar University, Kerman, Iran in 2009 and the Ph. D in Mathematics in Analysis field from Payame Noor University, Tehran, Iran in 2012. He has been a faculty member of Department of Research Institute for Mathematics, Imam Hossein University, Tehran, Iran. His research interests are Numerical Analysis, Non-linear Optimization, Numerical algorithms, Spectral Methods, Finite Elements, Differential equations and Matrix computations.