World Applied Sciences Journal 13 (Special Issue of Applied Math): 48-54, 2011
ISSN 1818-4952

A New Spectral Variational Iteration Method for Solving Nonlinear Two-Point Boundary Value Problems

A. Ghorbani, J. Saberi-Nadjafi and M. Gachpazan

Department of Applied Mathematics, School of Mathematical Sciences,
Ferdowsi University of Mashhad, Mashhad, Iran

Abstract: In this paper we develop an efficient method based on the Spectral Collocation Method (SCM) and the Variational Iteration Method (VIM) that can be used for the efficient numerical solution of nonlinear stiff/nonstiff two-point Boundary Value Problems (BVPs). The method derived here has the advantage that it does not require the solution of nonlinear systems of equations. We derive the method which requires one evaluation of the Jacobian and one LU decomposition per step. Some numerical experiments on nonlinear stiff/nonstiff problems and on well-known nonlinear BVPs like the Van der Pol and convective-radiative conduction problems which have been extensively studied show the efficiency and accuracy of the method.

Key words: Spectral variational iteration method . variational iteration method . spectral collocation method . two-point boundary value problems . nonlinear boundary conditions

INTRODUCTION

In this paper, we first investigate the approximate solution of the nonlinear second-order BVPs with linear boundary conditions of the type (we assumed that the problem has the unique solution on [a,b])

$$\begin{aligned} &F(x,y,y',y'')=0, \quad a\le x\le b \\ &y(a)=\alpha,\ y(b)=\beta, \end{aligned} \tag{1}$$

by a new spectral method proposed in this work, which is a combination of the Variational Iteration Method (VIM) [5, 12] and the (pseudo) Spectral Collocation Method (SCM) [2, 13]. This novel hybrid spectral VIM is then developed for solving the nonlinear kth-order BVPs with nonlinear two-point boundary conditions. Here a, b, α and β are the real constants and F is a nonlinear continuous function with respect to their arguments. These BVPs arise in engineering, applied mathematics and several branches of physics and have attracted much attention. However, it is difficult to obtain closed-form solutions for BVPs, especially for nonlinear problems. In most cases, only approximate solutions (either numerical solutions or analytical solutions) can be expected. Some numerical methods such as finite difference method [4], finite element method [3] and shooting method [11] have been developed for obtaining approximate solutions to BVPs.

It is well known that, if (1) is stiff, explicit methods generally may provide a good approximation of the solution only if the number of nodes is chosen very large and this choice is usually unfeasible from the computational point of view. For this reason, implicit methods are generally used to face such problems, but they require to solve a nonlinear system of equations. Obviously this represents a serious drawback if the number of nodes is large.

The strategy that will be pursued in this work rests mainly on establishing an effective algorithm, requiring no tedious computational work, based on the VIM and the SCM for solving (1). Unlike implicit methods, the method extracted in this paper does not require the solution of nonlinear systems of equations. This can be considered as a merit of this method over implicit methods. To demonstrate the utility of the proposed method, in this study, some nonlinear examples are given and are solved using the new method and compare the obtained results with the numerical results. In all cases, the present technique performed excellently.

VARIATIONAL ITERATION METHOD

In this section, we first describe the new version of the VIM [5] for solving (1). Then the local convergence is discussed in details.

Corresponding Author: Dr. A. Ghorbani, Department of Applied Mathematics, School of Mathematical Sciences, Ferdowsi University of Mashhad, Mashhad, Iran

Analysis of the VIM: The VIM provides the solution of (1) as a sequence of approximations. The method gives rapidly convergent successive approximations of the exact solution if such a solution exists, otherwise approximations can be used for numerical purposes. The idea of the VIM is very simple and straightforward. To explain the basic idea of the VIM, we first consider (1) as

$$L[y(x)] + N[y(x)] = g(x) \tag{2}$$

where L with the property $L[u] \equiv 0$ when $u \equiv 0$ denotes the so-called auxiliary linear operator with respect to y, N a nonlinear continuous operator with respect to y and g(x) is the source term. Next we construct a explicit iterative process for (2) as [5]:

$$L[y_{n+1}(x) - y_n(x)] = -A[y_n(x)] \tag{3}$$

with the boundary conditions

$$y_{n+1}(a) = \alpha \text{ and } y_{n+1}(b) = \beta \tag{4}$$

where

$$\begin{aligned} A[y_n(x)] &= L[y_n(x)] + N[y_n(x)] - g(x) \\ &\equiv F(x, y_n(x), y_n'(x), y_n''(x)) \end{aligned} \tag{5}$$

and the subscript n denotes the nth iteration and $y_0(x)$ is the initial guess, which can be freely chosen with possible unknown constants, or it can also be solved from its corresponding linear nonhomogeneous equation i.e.,

$$L[y_0(x)] = g(x)$$

It should be emphasized that we could construct a family of implicit iterative processes for (2) as follows:

$$L[y_{n+1}(x) - y_n(x)] = -(L[y_n(x)] + N[y_{n+1}(x)] - g(x)) \tag{6}$$

Accordingly, the successive approximations $y_n(x)$, $(n \geq 1)$ of the VIM equations (3) will be readily obtained. Consequently, the exact solution may be obtained by using

$$y(x) = \lim_{n\to\infty} y_n(x) \tag{7}$$

Therefore, the (n+1) th-order VIM equations (3) form a set of linear ordinary differential equations and can be easily solved, especially by means of symbolic computation software such as Maple, Mathematica, Matlab and others.

Convergence theorem: The variational iteration formula, (3), makes a recurrence sequence $\{y_n\{x\}\}$. Obviously, the limit of the sequence will be the solution of (1) if the sequence is convergent. In the following, we give a proof of convergence of the VIM, which details can be found in [5]. Here we suppose that for every n, $y_n \in C^2[a,b]$ and $\{y_n^{(m)}\},(m=1,2)$ is uniformly convergent.

Theorem: If the sequence (7) converges, where $y_n(x)$ is produced by the variational iteration formulation of (3), then it must be the exact solution of the problem (1).

Proof: If the sequence $y_n(x)$ converges, we can write

$$Y(x) = \lim_{n\to\infty} y_n(x) \tag{8}$$

and it holds

$$Y(x) = \lim_{n\to\infty} y_{n+1}(x) \tag{9}$$

Using (8), (9) and the definition of L, we can easily gain

$$\lim_{n\to\infty} L[y_{n+1}(x) - y_n(x)] = L \lim_{n\to\infty} [y_{n+1}(x) - y_n(x)] = 0 \tag{10}$$

From (10) and according to (3), we obtain

$$L \lim_{n\to\infty} [y_{n+1}(x) - y_n(x)] = -\lim_{n\to\infty} A[y_n(x)] = 0 \tag{11}$$

The relation (11) gives us

$$\lim_{n\to\infty} A[y_n(x)] = 0 \tag{12}$$

From (12) and continuity of the operator F, it follows that

$$\begin{aligned} \lim_{n\to\infty} A[y_n(x)] &= \lim_{n\to\infty} F(x, y_n(x), y_n'(x), y_n''(x)) \\ &= F(x, \lim_{n\to\infty} y_n(x), (\lim_{n\to\infty} y_n(x))', (\lim_{n\to\infty} y_n(x))'') \\ &= F(x, Y(x), Y'(x), Y''(x)) \end{aligned} \tag{13}$$

From equations (12) and (13), we get

$$F(x, Y(x), Y'(x), Y''(x)) = 0, \quad a \leq x \leq b \tag{14}$$

On the other hand, in view of (1), (4) and (9), it holds that

$$Y(a) = \lim_{n\to\infty} y_{n+1}(a) = \alpha$$

and

$$Y(b) = \lim_{n\to\infty} y_{n+1}(b) = \beta \tag{15}$$

Therefore, according to the above expressions, (14) and (15), Y(x) must be the exact solution of the problem (1) and this ends the proof.

It is clear that the convergence of the sequence (7) depends upon the initial guess $y_0(x)$ and the auxiliary linear operator L. Fortunately, the VIM provides us with great freedom of choosing them. Thus, as long as $y_0(x)$ and L are so properly chosen that the sequence (7) converges in a region $a \le x \le b$, it must converge to the exact solution in this region.

A SPECTRAL VARIATIONAL ITERATION METHOD

Before describing the new method, we point out that in order to solve a nonlinear two-point BVP using the (pseudo)spectral collocation method, because of the nonlinearity, it is no longer enough simply to invert the corresponding differentiation matrix. Instead, we can iteratively solve the problem via an explicit or implicit scheme with a stopping criterion [13]. In the implicit scheme, one obtains a nonlinear system of equations to be solved; this can be accomplished by a Newton-Krylov iteration (with a suitable preconditioner). The computational cost of this approach may be very high if the number of nodes is large. Note that this implementation may lead to divergent results.

On the other hand, since the VIM method presented above, the variational iteration formula (3) or (6), provides the solution as a sequence of iterates, its successive iterations may be very complex, so that the resulting integrations in its iterative relation may be impossible to perform analytically. In this section, we will overcome this shortcoming of the original VIM for solving (1) by proposing a new spectral VIM. As will be shown in this paper later, the new method will be very simple to implement and save time and calculations.

Consider basis functions φ_k that are polynomials of degree N-1 satisfying $\varphi_k(x_j) = \delta_{k,j}$ for the Chebyshev nodes (note that $x_1 = 1$ and $x_N = -1$)

$$x_j = \cos\left(\frac{(j-1)\pi}{N-1}\right), \qquad j = 1, \dots, N \tag{16}$$

The polynomial (the unknown function y(x) is approximated as a truncated series of polynomials)

$$p(x) \approx y(x) = \sum_{j=1}^{N} \phi_{j+1}(x) y_i \tag{17}$$

interpolates the points (x_j, y_j), that is, $p(x) = y$. The values of the interpolating polynomial's rth derivative at the nodes are

$$p^{(r)}(\mathbf{x}) = D^{(r)}\mathbf{y} \tag{18}$$

where the i, jth element of the differentiation matrix $D^{(r)}$ is $\phi_k^{(r)}(x_j)$. Note that $D^{(r)} \neq (D^{(1)})^r$.

Generally, in order to solve the problem (1) using a (pseudo)spectral collocation scheme, the interpolating polynomial is required to satisfy the differential equation at the interior nodes. The values of the interpolating polynomial at the interior nodes (m = 2: N-1 ≡ {2,3,…,N-2, N-1}) are $p(x_m) = y_m = I_{m,:}y$ and the derivative values are $p^{(r)}(\mathbf{x}_m) = D^{(r)}_{m,:}\mathbf{y}$. Boundary conditions that involve the values of the interpolating polynomial can be handled by using the formulas

$$p(\mathbf{x}_N) = \mathbf{y}_N = I_{N,:}\mathbf{y} \quad \text{and} \quad p(\mathbf{x}_1) = \mathbf{y}_1 = I_{1,:}\mathbf{y} \tag{19}$$

The references [1, 14] provide two useful Matlab functions for spectral collocation, chebdift and chebintt. The function call [x, DM] = chebdift (N, M, a, b) directly computes the transformed differentiation matrices for r = 1,2,…,M, where $0<M\le N-1$ and nodes on the arbitrary interval [a,b], where the subarray DM(:,:,r) contains the N×N matrix $DM^{(r)}$. The column vector x contains the transformed Chebyshev nodes with $x_1 = b$ and $x_N = a$. The function call p = chebintt (y,x) directly evaluates the polynomial that interpolates the data vector y at the transformed Chebyshev nodes on the arbitrary interval [a,b].

As a result, requiring no change of the variables, we can straightforwardly solve the problem (1). Here we present a simple iterative procedure for solving the problem (1).

Now, by using Eqs. (16)-(19), the VIM equation (6) and the boundary conditions (4) are transformed into the following matrix equations:

$$\begin{aligned} L[\mathbf{y}_m^{n+1} - \mathbf{y}_m^n] &= -(L[\mathbf{y}_m^n] + N[\mathbf{y}_m^{n+1}] - g(\mathbf{x}_m)) \\ \mathbf{y}_N &= \alpha \\ \mathbf{y}_1 &= \beta \end{aligned} \tag{20}$$

where

$$L[\mathbf{y}_m^n] \equiv L(\mathbf{x}_m, \mathbf{y}_m^n)$$

and

$$N[\mathbf{y}_m^{n+1}] \equiv N(\mathbf{x}_m, \mathbf{y}_m^{n+1})$$

The iterative formula (20) can be readily written in terms of the single unknown y^{n+1} as follows. This makes it particularly cheap for the solution of BVPs since the procedure now will be explicit. Now, by expanding $N(\mathbf{x}_m, \mathbf{y}_m^{n+1})$ in Taylor series about $\mathbf{y}_m^n$ (note that $\mathbf{y}_m^{n+1} = \mathbf{y}_m^n + (\mathbf{y}_m^{n+1} - \mathbf{y}_m^n)$), we can obtain

$$N[\mathbf{y}_m^{n+1}] \equiv N(\mathbf{x}_m, \mathbf{y}_m^{n+1}) \approx N(\mathbf{x}_m, \mathbf{y}_m^n) + (\mathbf{y}_m^{n+1} - \mathbf{y}_m^n) N_{y_m^n}(\mathbf{x}_m, \mathbf{y}_m^n) \quad (21)$$

where $N_{y_m^n}$ denotes the partial derivative of $N(\mathbf{x}_m, \mathbf{y}_m^n)$ with respect to $\mathbf{y}_m^n$.

Let I, 0 and diag denote the identity matrix of order N, the zero matrix of order N and a diagonal matrix of size N×N, respectively. By substituting the above matrix relations, the equations (20) can be collected into the single linear matrix equation, which is called the spectral VIM (SVIM)

$$\left(\begin{bmatrix} L \\ I_{N,:} \\ I_{1,:} \end{bmatrix} + \begin{bmatrix} I_{m,:}\operatorname{diag}(N_y) \\ 0_{N,:} \\ 0_{1,:} \end{bmatrix}\right)\mathbf{y}^{n+1} = \left(\begin{bmatrix} L \\ I_{N,:} \\ I_{1,:} \end{bmatrix} + \begin{bmatrix} I_{m,:}\operatorname{diag}(N_y) \\ 0_{N,:} \\ 0_{1,:} \end{bmatrix}\right)\mathbf{y}^{n} - \begin{bmatrix} L[\mathbf{y}_m^n] + N[\mathbf{y}_m^n] - g(\mathbf{x}_m) \\ \mathbf{y}_N - \alpha \\ \mathbf{y}_1 - \beta \end{bmatrix} \quad (22)$$

or compactly

$$\mathbf{E}\mathbf{y}^{n+1} = \mathbf{E}\mathbf{y}^{n} - A[\mathbf{y}^n] \quad (23)$$

where E = L+J and according to (5), the matrix L, the matrix J and the vector $A[y^n]$ are respectively as

$$\mathbf{L} = \begin{bmatrix} L \\ I_{N,:} \\ I_{1,:} \end{bmatrix} \quad \text{and} \quad \mathbf{J} = \begin{bmatrix} I_{m,:}\operatorname{diag}(N_y) \\ 0_{N,:} \\ 0_{1,:} \end{bmatrix} \quad (24)$$

and

$$A[\mathbf{y}^n] = \begin{bmatrix} I_{m,:}F(\mathbf{x}, \mathbf{y}, D^{(1)}\mathbf{y}, D^{(2)}\mathbf{y}) \\ \mathbf{y}_N - \alpha \\ \mathbf{y}_1 - \beta \end{bmatrix} \quad (25)$$

The linear system (23) can be solved by means of LU decomposition per step. Here the vector y^{n+1} is defined as:

$$\mathbf{y}^{n+1} = \{y^{n+1}(x_1), y^{n+1}(x_2), \ldots, y^{n+1}(x_N)\} \quad (26)$$

Using the SVIM algorithm above, we begin by choosing the best possible initial approximation that satisfies the boundary conditions (4). To this end, we may determine the initial approximation by solving the linear system $Ly_0 = g$ where $g = [g(x_m), \alpha, \beta]^T$ (the superscript T denotes the transpose). Thus, starting from the initial approximation y^0, we can use the recurrence formula (23) to successively obtain directly y^{n+1} for $n \geq 0$.

It should be emphasized that the main advantage of the SVIM algorithm presented here is its simplicity and its accuracy in solving the nonlinear two-point BVPs subject to general boundary conditions. Assume that we want to solve a nonlinear second-order BVP as $F(x,y,y',y'') = 0$ subject to the general two-point boundary condition $G(y(a), y(b)) = 0$ (with two nonlinear equations) using the SVIM algorithm (23). In this case, the matrix L, the matrix J and the vector $A[y^n]$ of the iterative relation (23) to solve the above problem become:

$$\mathbf{L} = \begin{bmatrix} L_F \\ L_G \end{bmatrix} \quad \text{and} \quad \mathbf{J} = \begin{bmatrix} I_{m,:}\operatorname{diag}\{(N_F)_y\} \\ I_{\bar{m},:}\operatorname{diag}\{(N_G)_y\} \end{bmatrix} \quad (27)$$

and

$$A[\mathbf{y}^n] = \begin{bmatrix} I_{m,:}F(\mathbf{x}, \mathbf{y}, D^{(1)}\mathbf{y}, D^{(2)}\mathbf{y}) \\ G(\mathbf{y}_N, \mathbf{y}_1) \end{bmatrix} \quad (28)$$

with

$$\begin{aligned} F(x,y,y',y'') &:\equiv L_F[y(x)] + N_F[y(x)] = g(x) \\ G(y(a),y(b)) &:\equiv L_G[y(a),y(b)] + N_G[y(a),y(b)] = bc \end{aligned} \quad (29)$$

where $m = 2: N-1$, $\bar{m} = \{1, N\}$, $y_1 = y(b)$ and $y_N = y(a)$, L_F and L_G are the auxiliary linear operators of the differential equation F and boundary condition G and N_F and N_G the nonlinear operators of F and G, respectively.

Now suppose that we want to solve a nonlinear kth-order BVP as $F(x,y,y',\ldots,y^{(k)}) = 0$ subject to the general two-point boundary condition $G(y(a), y(b)) = 0$ (with $k \geq 3$ nonlinear equations) using the SVIM algorithm (23). In this case, the matrixes L and J and the vector $A[y^n]$ of the iterative relation (23) to solve the above problem become:

$$\mathbf{L} = \begin{bmatrix} L_F \\ L_G \end{bmatrix} \quad \text{and} \quad \mathbf{J} = \begin{bmatrix} I_{m,:}\operatorname{diag}\{(N_F)_y\} \\ I_{\bar{m},:}\operatorname{diag}\{(N_G)_y\} \end{bmatrix} \quad (30)$$

and

$$A[\mathbf{y}^n] = \begin{bmatrix} I_{m,:}F(\mathbf{x}, \mathbf{y}, D^{(1)}\mathbf{y}, \cdots, D^{(k)}\mathbf{y}) \\ G(\mathbf{y}_N, \mathbf{y}_1) \end{bmatrix} \quad (31)$$

where y_1, y_N, L_F, L_G, N_F and N_G are as noted above and

$$m = \begin{cases} i+1 : N-i \equiv \{i+1, i+2, \ldots, N-(i+1), N-i\}, & k = 2i \\ i+1 : N-(i+1) \equiv \{i+1, i+2, \ldots, N-(i+2), N-(i+1)\}, & k = 2i+1 \\ 2i+1 : N-i \equiv \{2i+1, 2i+2, \ldots, N-(i+1), N-i\}, & k = 2i+1 \end{cases} \quad (32)$$

and

$$\overline{m} = \begin{cases} \{1,\ldots,i\}\cup\{N,\ldots,N-(i-1)\}, & k=2i \\ \{1,\ldots,i\}\cup\{N,\ldots,N-i\}, & k=2i+1 \\ \{1,\ldots,2i\}\cup\{N,\ldots,N-(i-1)\}, & k=2i+1 \end{cases} \quad (33)$$

(note that $m\cup\overline{m}=\{1,\ldots,N\}$). Here bc is a k-by-1 vector, that is, the values of right hand side of boundary conditions.

NUMERICAL EXPERIMENTS

In this section, to give a clear overview of the content of this study, several nonlinear two-point BVPs will be tested by the proposed SVIM algorithm, which will ultimately show the efficiency of this method. We mention that all tests here are started from $y^0 = L^{-1}g$ as noted before, performed in Matlab with machine precision 10^{-16} and terminated when the current iterate satisfies $\|y^{n+1}-y^n\|_\infty \le 10^{-15}$, where y^n is the solution vector of the nth SVIM iteration.

Before considering our numerical results, we indicate that the nonlinear problems analyzed here become increasingly stiff as the corresponding parameters within the frame of equations are increased.

Example 1 (Van der Pol equation) In this example we consider the following nonlinear second-order boundary value problem

$$y'' + \mu(y^2-1)y' + y = 0\ ,\ 0\le x\le 1 \quad (34)$$

with the boundary conditions y(0) = 2 and y(1) = 0.

In order to solve the equation (34) using the SVIM algorithm (23), according to (2) and (29), we choose

$$L_F[y(x)] = y'' + y,\ N_F[y(x)] = \mu(y^2-1)y' \text{ and } g(x)\equiv 0 \quad (35)$$

and

$$L_G[y(a),y(b)] = \begin{bmatrix} y(0) \\ y(1) \end{bmatrix}$$

$$N_G[y(a),y(b)] = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \text{ and } bc = \begin{bmatrix} 2 \\ 0 \end{bmatrix} \quad (36)$$

with

$$(N_F)_y = 2\mu y D^{(1)} y + (y^2-1)D^{(1)} \text{ and } (N_G)_y = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

To show the efficiency of the SVIM algorithm, here we plotted the solution of (34) for different values of μ, as shown in Fig. 1.

Example 2 (Convective-radiative conduction equation) In this example we consider the following nonlinear second-order boundary value problem [8]

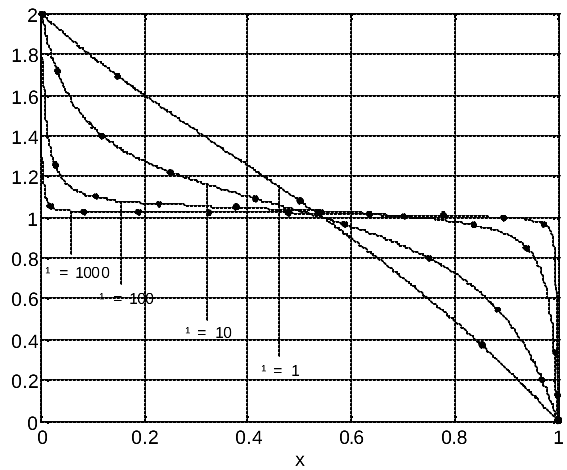

Fig. 1: The numerical comparisons of the SVIM solution (symbols) when N = 5(μ = 1), N = 10 (μ = 10), N = 20 (μ = 100) and N = 110 (μ = 1000) with the numerical finite-difference solution (lines) for different values of μ in Example 1

$$y'' - M^2 y + \varepsilon_1 y'^2 + \varepsilon_1 y y'' - \varepsilon_2 y^4 = 0,\quad 0\le x\le 1 \quad (37)$$

with the linear boundary conditions $y'(0) = 0$ and $y(1) = 1$.

In order to solve the equation (37) using the SVIM algorithm (23), according to (2) and (29), we choose

$$L_F[y(x)] = y'' - M^2 y$$

$$N_F[y(x)] = \varepsilon_1 y'^2 + \varepsilon_1 y y'' - \varepsilon_2 y^4 \text{ and } g(x)\equiv 0 \quad (38)$$

and

$$L_G[y(a),y(b)] = \begin{bmatrix} y'(0) \\ y(1) \end{bmatrix}$$

$$N_G[y(a),y(b)] = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \text{ and } bc = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad (39)$$

with

$$(N_F)_y = 2\varepsilon_1 D^{(1)}(D^{(1)} y) + \varepsilon_1 D^{(2)} y + \varepsilon_1 y D^{(2)} - 4\varepsilon_2 y^3$$

and

$$(N_G)_y = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

In Fig. 2 we have plotted the solution obtained using the SVIM algorithm when N = 25 versus the Matlab bvp4c solver solution for different values of M for Example 2 with $\varepsilon_1 = 1$ and $\varepsilon_2 = 1$.

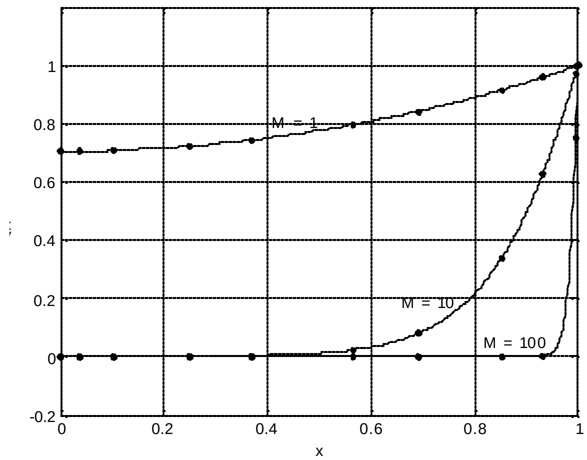

Fig. 2: The numerical comparison of the SVIM solution (symbols) when N = 25 with the Matlab bvp4c solver solution (lines) for different values of M for Example 2 with $\varepsilon_1 = 1$ and $\varepsilon_2 = 1$

Example 3 In this example we consider the following nonlinear second-order boundary value problem [6]

$$y'' - xy - \cos(y') = 0, \quad 3 \le x \le 5 \tag{40}$$

with the nonlinear boundary conditions $y(3)+y'(5) = 7$ and $y'^2(3)y(5) = 10$.

In order to solve the equation (40) using the SVIM algorithm (23), according to (2) and (29), we choose

$$L_F[y(x)] = y'' - xy, \quad N_F[y(x)] = -\cos(y') \text{ and } g(x) \equiv 0 \tag{41}$$

and

$$L_G[y(a), y(b)] = \begin{bmatrix} y(3) + y'(5) \\ y'(3) \end{bmatrix}$$

$$N_G[y(a), y(b)] = \begin{bmatrix} 0 \\ -\sqrt{\dfrac{10}{y(5)}} \end{bmatrix} \text{ and } bc = \begin{bmatrix} 7 \\ 0 \end{bmatrix} \tag{42}$$

with

$$\left(N_F\right)_y = D^{(1)} \sin(D^{(1)} y) \text{ and } \left(N_G\right)_y = \begin{bmatrix} 0 \\ \dfrac{\sqrt{10}}{2y(5)^{3/2}} \end{bmatrix}$$

Here, we have plotted the solution ($y(x)$ and $y'(x)$) obtained using the SVIM algorithm when $J \equiv 0$ and N = 10 versus the Matlab bvp4c solver solution in Fig. 3.

Remark: It is interesting to point out that the user can consider $J \equiv 0$ within the frame of the SVIM algorithm

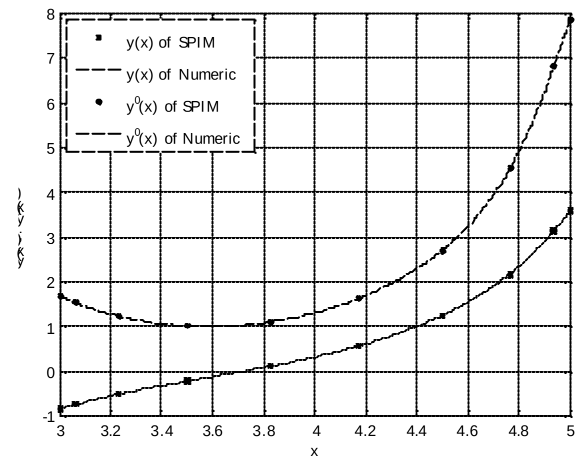

Fig. 3: The numerical comparison of the SVIM solution when $J \equiv 0$ and N = 10 with the Matlab bvp4c solver solution for Example 3

in solving the non-stiff BVPs (for instance, see Example 3).

CONCLUSION

In this study we have proposed a new hybrid spectral variational iteration method for the solution of nonlinear non-stiff/stiff boundary value problems with general two-point boundary conditions. The obtained results demonstrate that the SVIM algorithm has the following advantages over the spectral-homotopy analysis method (SHAM) proposed in [9, 10]. The SVIM is more efficient than the SHAM as it does not depend on the calculation of the so-called homotopy polynomials. This new method is easy to implement, accurate when applied to the nonlinear two-point BVPs and avoids additional computational work. The SVIM works successfully in handling the two-point BVPs. This confirms our belief that the SVIM is a promising scheme in solving the nonlinear two-point BVPs and more promising because it can be utilized for a wider class of BVPs with high accuracy.

REFERENCES

1. Ghorbani, A. and J. Saberi-Nadjafi, 2011. A piecewise-spectral parametric iteration method for solving the nonlinear chaotic Genesio system. Mathematical and Computer Modelling, 54: 131-139.
2. Canuto, C., M.Y. Hussaini, A. Quarteroni and T.A. Zang, 1988. Spectral Methods in Fluid Dynamics, Springer-Verlag, Berlin.
3. Deacon, A.G. and S. Osher, 1979. Finite-element

method for a boundary-value problem of mixed type. SIAM J. Numer. Anal., 16: 756-778.

4. Doedel, E., 1979. Finite difference methods for nonlinear two-point boundary-value problems. SIAM J. Numer. Anal., 16: 173-185.
5. Ghorbani, A. and J. Saberi-Nadjafi, 2010. Convergence of He’s Variational Iteration Method for Nonlinear Oscillators. Nonlinear Sci. Lett. A, 1: 379-384.
6. Kincaid, D.R. and E.W. Cheny, 2001. Numerical Analysis Mathematics of Scientific Computing, Brooke Cole, California.
7. Liao, Sh.J., 2003. Beyond Perturbation: Introduction to the Homotopy Analysis Method, Boca Raton, Chapman & Hall/CRC Press.
8. Miansari, Mo., D.D. Ganji and Me. Miansari, 2007. Application of He's variational iteration method to nonlinear heat transfer equations. Phys. Lett. A, doi:10.1016/j.physleta.2007.08.065.
9. Motsa, S.S., P. Sibanda, F.G. Awad and S. Shateyi, 2010. A new spectral-homotopy analysis method for the MHD Jeffery-Hamel problem. Computers & Fluids, doi:10.1016/j.compfluid.2010.03.004.
10. Motsa, S.S., P. Sibanda and S. Shateyi, 2009. A new spectral-homotopy analysis method for solving a nonlinear second order BVP. Commun. Nonlinear Sci. Numer. Simulat., doi:10.1016/j.cnsns.2009.09.019.
11. Roberts, S.M. and J.S. Shipman, 1972. Two Point Boundary Value Problems: Shooting Methods. American Elsevier, New York.
12. He, J.H., 2007. Variational iteration method-some recent results and new interpretations. J. Comput. Appl. Math., 207: 3-17.
13. Trefethen, L.N., 2000. Spectral Methods in MATLAB, SIAM.
14. Weideman, J.A.C. and S.C. Reddy, 2000. A MATLAB Differentiation Matrix Suite. ACM Transactions on Mathematical Software, 26: 465-519.