Punjab University Journal of Mathematics
ISSN: 1016-2526
Vol. 51(11)(2019) pp. 101-113

An improved shooting technique for solving boundary value problems using higher order initial approximation algorithms

Shumaila Javeed
Department of Mathematics,
COMSATS University, Islamabad, Pakistan,
Email: shumaila_javeed@comsats.edu.pk

Assiya Shabnam
Department of Mathematics,
COMSATS University, Islamabad, Pakistan,
Email: assiya.shabnam@gmail.com

Dumitru Baleanu
Department of Mathematics,
Cankaya University, Ankara, Turkey & Institute of Space Sciences, Magurele-Bucharest, Romania,
Email: dumitru@cankaya.edu.tr

Received: 13 March , 2019 / Accepted: 06 September, 2019 / Published online: 01 October, 2019

Abstract. This paper introduces the better algorithms to obtain refined initial guesses with shooting method for solving boundary value problems (BVPs). Each boundary value problem (BVP) is reformulated as a system of equations i.e. initial value problems (IVPs) with one unknown initial conditions. Afterwards, the system of equations is solved using newly developed shooting method [2]. This article proposes efficient initial guess algorithms rather than conventional Newton method to approach the adjoint terminal conditions rapidly. We enhanced the efficiency and accuracy of shooting method by first improving our initial guess and then solving the problem iteratively. The suggested technique is applied to solve different nonlinear higher order boundary value problems. The results indicate that the proposed method is more efficient and accurate as compared to build-in-functions which is being used in MATLAB.

1. Introduction

BVPs are very important owing to their diverse use in different fields e.g. applied mathematics includes the boundary layer theory in fluid mechanics, theoretical physics, engineering, control and optimization theory. Since the analytical solution of a BVP is not always possible, efficient and accurate numerical schemes are required to solve a BVP. One of the most widely used techniques for the aforementioned purpose is the shooting method. The shooting method is based on converting a BVP into an equivalent system of IVPs. Thus, the benefits of accuracy and efficiency of an IVP's solver can be utilized for solving a BVP. Shooting methods have been widely studied by numerous researchers to get reliable solutions of non linear BVPs [13, 10].

The shooting method to obtain eigen-values of fourth-order BVPs was investigated by D. J. Jones [9]. Wang et al studied a second order multi-point integral BVPs [21]. Kwong and Wong [11] have discussed the shooting method for solving non-homogeneous multipoint BVPs of second-order ordinary differential equations (ODEs). Granas et al presented the shooting method for a class of nonlinear BVPs [7]. Russell and Shampine studied various numerical techniques for solving singular BVPs [18].

The shooting method is based on converting a BVP into an equivalent system of IVPs. In order to obtain accurate results of the reformulated system of IVPs, an appropriate initial guess should be made to initiate a recursive procedure. Initial guesses are usually acquired by using iterative methods for finding roots of algebraic equations, such as the Newton method, the secant method and interpolation formulae, etc. This paper proposes new ways to obtain efficient initial guesses of higher order methods which work far better than conventional methods and yield faster convergence.

We proposed and proved the idea of using a family of iterative initial approximation algorithms to refine initial guesses for approaching the adjoint terminal condition rapidly. The fast convergence can be achieved by utilizing these initial approximation algorithms. A BVP is reformulated into an associated system of IVPs and, then, the IVPs are solved by refining an initial guess. Shooting technique with Newton-Raphson formula is mostly used in different softwares to solve a BVP, but Newton-Raphson formula fails to predict results when the first derivative of a function is zero or undefined. To avoid this issue, we utilize a family of higher order iterative methods to approximate the initial guess for solving nonlinear BVPs in an efficient manner. The novelty of this work is to introduce and implement a family of iterative initial approximation algorithms with shooting method instead of using conventional Newton method. The rest of the paper is organized as follows: Section 2 presents different algorithms to refine initial guesses, Section 3 describes the proposed shooting method, Section 4 contains the test problems with results of some nonlinear BVPs, and Section 5 summarizes the findings of the paper.

2. IMPROVED SHOOTING METHOD

Consider a nonlinear second-order BVP,

$$y'' = f(x, y, y'), \quad \text{for } a \leq x \leq b \text{ where } y(a) = \alpha \text{ and } y(b) = \beta,\ \alpha, \beta \in R. \tag{2. 1}$$

This BVP can be rewritten as

$$y' = z, \qquad z' = f(x, y, z), \qquad y(a) = \alpha \text{ and } y(b) = \beta. \tag{2. 2}$$

Then, this system of BVP is transformed into a system of IVPs by replacing the boundary condition (BC) at $x = b$ as

$$y' = z, \qquad z' = f(x, y, z), \qquad y(a) = \alpha \text{ and } y'(a) = t. \tag{2. 3}$$

Afterwards, the parameter $t = t_k$ is selected in such a way that it satisfies

$$\lim_{k\to\infty} y(b, t_k) = y(b) = \beta,$$

where $y(x, t_k)$ represents the solution of IVP (c.f. Eq. 2. 3) with $t = t_k$, and the corresponding $y(x)$ represents the solution of the BVP (c.f. Eq. 2. 1).
We begin the shooting method process with an initially selection of the parameter $t = t_0$. If $y(b, t_0)$ is not sufficiently close to β, we choose another elevation, i.e. t_1, until $y(b, t_0)$ is sufficiently close to hitting the target β. Generally, Newton's method or a secant method are being used to choose an initial guess for solving the BVP.
The IVP (c.f. Eq. 2. 3), emphasizes that the solution depends on both x and t as

$$y''(x, t) = f(x, y(x, t), y'(x, t)), \quad \text{for } a \le x \le b, \tag{2. 4}$$

where $y(a, t) = \alpha$ and $y'(a, t) = t$.
Some higher order algorithms are used to hit the target, $y(b) = \beta$, efficiently. To begin this process, we have to calculate $\frac{\partial y}{\partial t}(b, t_k)$.
Differentiating Eq. 2. 4 with respect to t yields,

$$z''(x, t) = f_y(x, y, y')z(x, t) + f_{y'}(x, y, y')z'(x, t), \text{ for } a \le x \le b, \tag{2. 5}$$

where $z(a, t) = 0$ and $z'(a, t) = 1$.
This shows that for each iteration solving two IVPs, given in Eqs. 2. 3 and 2. 5 , are required [5]. From Eq. 2. 5 , the value of $z(b, t_k)$ will be obtained and used in the subsequent subsection to find the sequence of t_k used for rapidly approaching to the right BC, i.e. $y(b) = \beta$.

2.1. **Iterative algorithms to refine initial guess.**

In recent years several higher order iterative methods have been developed for solving nonlinear equations. These methods are based on several techniques, i.e. Taylor series, decomposition method, variational iteration technique, homotopy perturbation method, and quadrature formula. Researchers used these techniques to develop several one and two step iterative methods for achieving higher order convergence [1, 15, 6].
This work proposes to use two or three steps methods instead of applying Newton and secant methods. These suggested algorithms proved to be faster in terms of convergence than conventional methods. In the subsequent section, the applied algorithms are presented and Newton's method is considered for comparison.

2.1.1. *Algorithm 1: Newton's method.* For a given t_0, we calculate t_1, t_2 ,... such that

$$t_{k+1} = t_k - \frac{y(b, t_k) - \beta}{z(b, t_k)}. \tag{2. 6}$$

2.1.2. *Algorithm 2.* For a given t_0, we calculate t_1, t_2 ,... such that

$$t_{k+1} = t_k - \frac{[y(b, s_k) - \beta][y(b, r_k) - \beta]}{z(b, r_k)[y(b, r_k) - \beta] - 2\gamma[y(b, s_k) - \beta]}, \tag{2. 7}$$

where $$r_k = t_k - \frac{y(b, t_k) - \beta}{z(b, t_k)} \quad \text{and} \quad s_k = r_k - \frac{y(b, r_k) - \beta}{z(b, r_k)}.$$

Here, $\gamma \neq 0$ is the controlling parameter and should be chosen as to make denominator largest in magnitude. The order of Algorithm 2 is at least 6 and the convergence analysis is presented in [19].

2.1.3. *Algorithm 3.* For a given t_0, we calculate t_1, t_2 ,... such that

$$t_{k+1} = t_k - \frac{2[y(b,t_k) - \beta]}{z(b,t_k) \pm \sqrt{[z(b,t_k)]^2 + 4p^3[y(b,t_k) - \beta]^3}}, \tag{2. 8}$$

where $p \, \varepsilon \, \Re$, should be chosen as to make the denominator largest in magnitude and both $[y(t_k) - \beta]$ and p have the same sign. The Algorithm 3 converges quadratically [14].

2.1.4. *Algorithm 4.* For a given t_0, we calculate t_1, t_2 ,... such that

$$t_{k+1} = s_k - \frac{[y(b,s_k) - \beta][y(b,r_k) - \beta]}{z(b,r_k)[y(b,r_k) - \beta] + \gamma[[y(b,r_k) - \beta] - 2[y(b,s_k) - \beta]]}, \tag{2. 9}$$

$$\text{where} \qquad r_k = t_k - \frac{y(b,t_k) - \beta}{z(b,t_k)} \quad \text{and} \quad s_k = r_k - \frac{y(b,r_k) - \beta}{z(b,r_k)}.$$

where the controlling parameter $\gamma \neq 0$ should be chosen in such a way to make denominator largest in magnitude. The order of Algorithm 4 is at least 6 and the convergence analysis is presented in [19].
In order to obtain the desired accuracy, the following criteria is used.
1. For a given $\varepsilon > 0$, if $|t_{k+1} - t_k| < \varepsilon$, then stop.
2. Otherwise, set $k = k + 1$ and repeat the recursive process until the target is approached.
Finally, the implemented iterative formula for solving the IVP is presented in the subsequent section.

2.2. **Iterative formula to solve IVP.** Various iterative formulas, e.g. Euler's method and a family of Runge-Kutta methods, can be used to solve IVPs. In this study, the IVPs given in Eqs. 2. 3 and 2. 5 , are solved by the iterative formula proposed in [2] using a Taylor series approach.
For $n = 0, 1, 2, ...$ we define our iterative method as:

$$\begin{aligned} y_{n+1} &= y_n + \frac{h}{3}(k_0 + k_1 + k_2), \qquad (2.\ 10)\\ x_{n+1} &= x_n + h,\\ \text{where } k_0 &= f(x_n, y_n),\\ k_1 &= f(x_n + \frac{h}{2}, y_n + \frac{h}{2}k_0),\\ k_2 &= f(x_n + h, y_n + hk_1). \end{aligned}$$

The proposed method is applied to solve linear and non-linear BVPs in the subsequent section.

3. NUMERICAL TEST PROBLEMS

This section comprises of three non-linear BVPs which mainly occur in fluid dynamics. In test problems, the suggested iterative formula to solve IVPs with the proposed algorithms for updating the initial guess is applied.

Problem 1: Blasius equation

Blasius equation is derived from the famous Navier-Stokes equations for boundary layer flows using similarity transformation [20].

The equation states

$$f''' + f(\eta)f'' = 0, \quad \text{with BCs} \quad f(0) = 0, \quad f'(0) = 0 \text{ and } f'(\infty) = 1. \tag{3. 11}$$

After converting Eq. 3. 11 into the system IVP (c.f. Section 2), we consider $f''(0) = t\ (some\ initial\ guess)$ and try to satisfy the BC $f'(\infty) = 1$. If this solution does not satisfy the BC, we will continue to update initial guesses using the algorithms described in Section 2.

To find the solution of Blasius equation, the domain, i.e. $f'(\infty) = 1$, is restricted arbitrarily to $f'(10) = 1$.

The first initial guess is taken as $t_0 = 0.332$, which is revealed by the numerical solutions in literature [17]. After the completion of first iteration of the shooting method, we implemented efficient and accurate Algorithms 1 to 4 for obtaining initial guesses (c.f. Section 2.1). The initial guesses yield t_1 options given below.

Algorithm 1 gives $t_1 = 0.4617497568$.

Algorithm 2 gives $t_1 = 0.4695996403$.

Algorithm 3 gives $t_1 = 0.4613020362$.

Algorithm 4 gives $t_1 = 0.4696004256$. Using these aforementioned initial guesses, the techniques presented in Section 2.2 are applied to solve the problem. The results obtained from the proposed algorithms are presented in Table 1. The relative percentage error was calculated using the formula given below:

$$\text{Relative percent error} = \frac{\text{Absolute Error}}{\text{Exact solution}} \times 100\%. \tag{3. 12}$$

It can be seen that the proposed Algorithm 2 and Algorithm 4 approach the target (right boundary) accurately as compared to Algorithm 1 (Newton's Method) and Algorithm 3. As Algorithm 2 and Algorithm 4 reached the right boundary rapidly, than Algorithms 1 and 3, are proven to be more accurate and efficient. This is also quantitatively depicted from relative percentage error in Table 1. The solution of Blasius equation is presented in Fig. 1 using step size $h = 0.005$. It can be seen from Fig. 1 that the results produced by using Algorithm 2 and Algorithm 4 are more accurate. As Algorithm 1 and Algorithm 3 did not hit the target in first iteration, we repeated the procedure and acquired the next initial guess. The values of new initial guesses obtained from Algorithm 1 and Algorithm 3 are $t_2 = 0.4696015437$ and $t_2 = 0.4696002883$, respectively. After using the updated initial guesses (called t_2), the shooting method approaches the target $f'(10) = 1$ as shown in Fig. 2. It is worthwhile to mention that Algorithms 2 and 4 hit the target in one iteration while Algorithms 1 and 3 took two iterations. Consequently, the computational time of using Algorithms 1 and 3 is twice as compared to Algorithms 2 and 4. Therefore, Algorithms 2 and 4 are preferable algorithms to acquire initial guess in shooting method for solving

BVP. On the basis of these results, one can infer the inclusion of better initial approximation algorithms with shooting method produce more accurate results and converges fast.

Table 1: Comparison of proposed initial guess Algorithms and relative percentage error in hitting condition $f'(10) = 1$.

Algorithm	t	f	f'	f''	$CPUtime/sec$	Relative err (%) f'
1	0.4617497568	8.67828	0.98882	0.00000	1.279381	1.118
2	0.4695996403	8.78322	1.00000	0.00000	1.134184	0.000
3	0.4613020362	8.67228	0.98819	0.00000	1.23462	1.181
4	0.4696004256	8.78323	1.00000	0.00000	1.142960	0.000

Problem 2: Falkner-Skan equation
Consider the famous Falkner-Skan equation governed by

$$f''' + ff'' - f'^2 + 1 = 0, \text{with BCs } f(0) = 0, \quad f'(0) = 0 \text{ and } f'(\infty) = 1. \tag{3. 13}$$

After converting Eq. 3. 13 to IVP, we consider $f''(0) = t\ (some\ initial\ guess)$ and try to satisfy the BC $f'(\infty) = 1$. If this solution does not satisfy the BC, we will continue to update initial guesses using the algorithms described in Section 2.
To find the solution of Falkner-Skan equation, the domain of the problem is restricted arbitrarily to $f'(10) = 1$.
The first initial guess is taken as $t_0 = 1.224744871391$, obtained from the numerical calculations done in literature [16]. After the completion of first iteration of the shooting method, we applied Algorithms 1 to 4 to obtain the next initial guesses, t_1, given below.
Algorithm 1 gives $t_1 = 1.23350588$.
Algorithm 2 gives $t_1 = 1.23258786$.
Algorithm 3 gives $t_1 = 1.23350543$.
Algorithm 4 gives $t_1 = 1.232587955$.
Using these aforementioned initial guesses, the technique presented in Section 2.2 is applied to solve the problem. The solution of the Falkner-Skan equation is presented in Fig. 3 with step size $h = 0.001$. Using the proposed Algorithm 2 and Algorithm 4, results approached the target (right boundary) more accurately than Algorithm 1 (Newton's method) and Algorithm 3, as depicted in Fig. 3. Therefore, Algorithm 2 and Algorithm 4 are proven to be accurate and efficient. As Algorithm 1 and Algorithm 3 did not hit the target in first iteration, we repeated the procedure and acquired the next initial guesses t_2 and t_3. The values of new initial guesses obtained from Algorithm 1 and Algorithm 3 are $t_3 = 1.232587638$ and $t_3 = 1.232587601$, respectively. After using the updated initial guesses (called t_3), the shooting method approached the target i.e. $f'(10) = 1$ rapidly, as depicted in Fig. 4. Algorithms 2 and 4 hit the target in one iteration while Algorithm 1 and 3 took two iterations again. This means that Algorithm 1 and 3 requires more computational time as compared to Algorithms 2 and 4. Therefore, Algorithms 2 and 4 are proven to be more suitable algorithms to acquire an initial guess in the shooting method for solving BVPs. On the basis of these results, one can conclude that using better initial

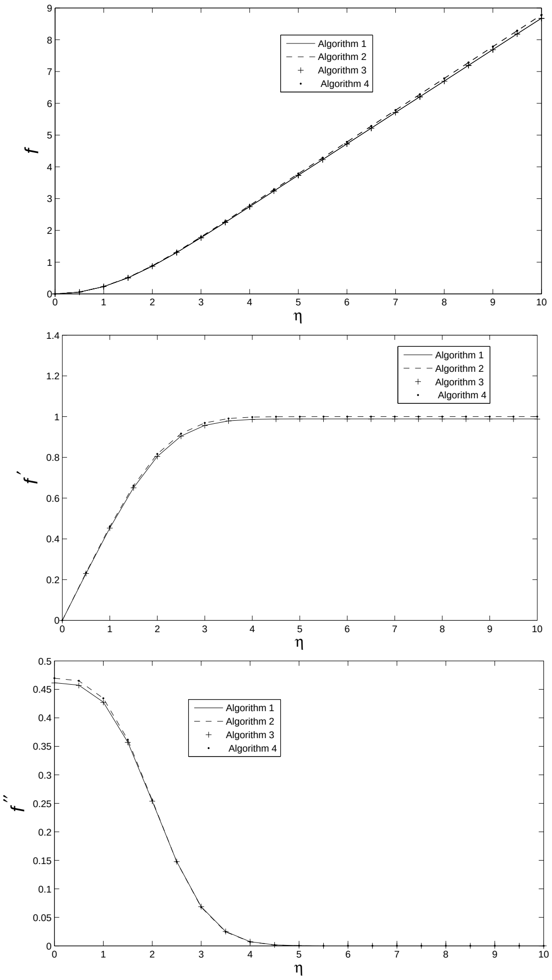

Figure 1: Problem 1: Comparison of f', f, f'' after one time recursive iterations with $h = 0.005$.

approximation algorithms produce more accurate results and converges faster.

Problem 3: Hydromagnetic fluid problem
Consider a Hydromagnetic Fluid problem governed by the equation

$$f''' + ff'' - f' - f'^2 = 0, \text{ with BCs } f(0) = 0, \; f'(0) = 0 \text{ and } f'(\infty) = 0. \qquad (3.\ 14)$$

After converting Eq. 3. 14 to IVP, we consider $f''(0) = t \; (some\ initial\ guess)$ and try to satisfy the BC $f'(\infty) = 0$. If this solution does not satisfy the BC, we will continue to update initial guesses using the algorithms described in Section 2.

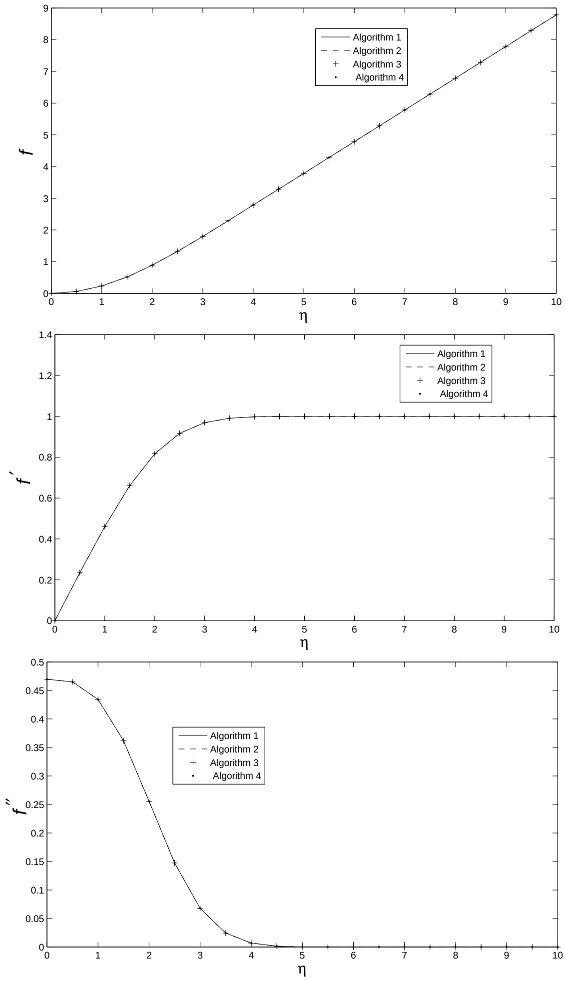

Figure 2: Problem 1: Comparison of f', f, f'' after two times recursive iterations with $h = 0.005$.

In order to solve the problem, the domain is truncated arbitrarily to $f'(10) = 0$.
The first initial guess is taken as $t_0 = -1.42$ [16]. After the completion of the first iteration of the shooting method, we applied Algorithms 1 to 4 to obtain the next initial guesses, t_1. The solution did not converge accurately. Thus, we used Algorithms 1 to 4 again to further refine the initial guesses, t_2. After the recursive procedure is completed two times, we obtained the following new initial guesses.
Algorithm 1 gives $t_2 = -1.414200017$.
Algorithm 2 gives $t_2 = -1.414213560$.
Algorithm 3 gives $t_2 = -1.414200061$.

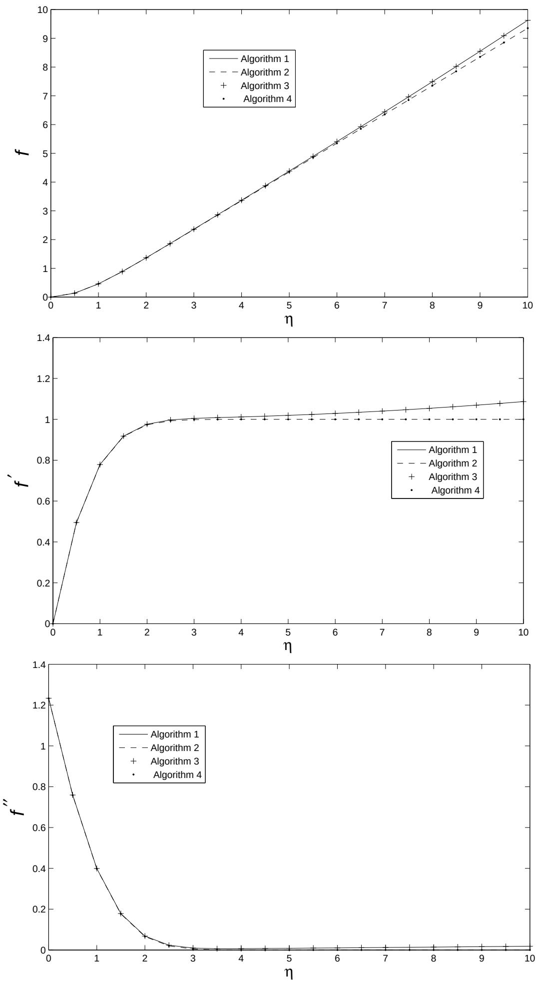

Figure 3: Problem 2: Comparison of f', f, f'' after 1 time recursive iterations with $h = 0.001$.

Algorithm 4 gives $t_2 = -1.414213566$.
The solution of the Hydromagnetic fluid problem is presented in Fig. 5 with step size $h = 0.0025$.. It can be seen from Fig. 5 that the results produced by using Algorithm 2 and Algorithm 4 are more accurate. As Algorithm 1 and Algorithm 3 did not hit the target in two iterations, we repeated the procedure and acquired the next initial guesses, t_3 and t_4. The values of new initial guesses obtained from Algorithm 1 and Algorithm 3 are $t_4 = -1.414213559$ and $t_4 = -1.414213563$, respectively. After using the updated initial guesses (called t_4), the shooting method reached the target $f'(10) = 0$, as shown in Fig. 6 . It is important to conclude that Algorithms 2 and 4 hit the target in two iterations while

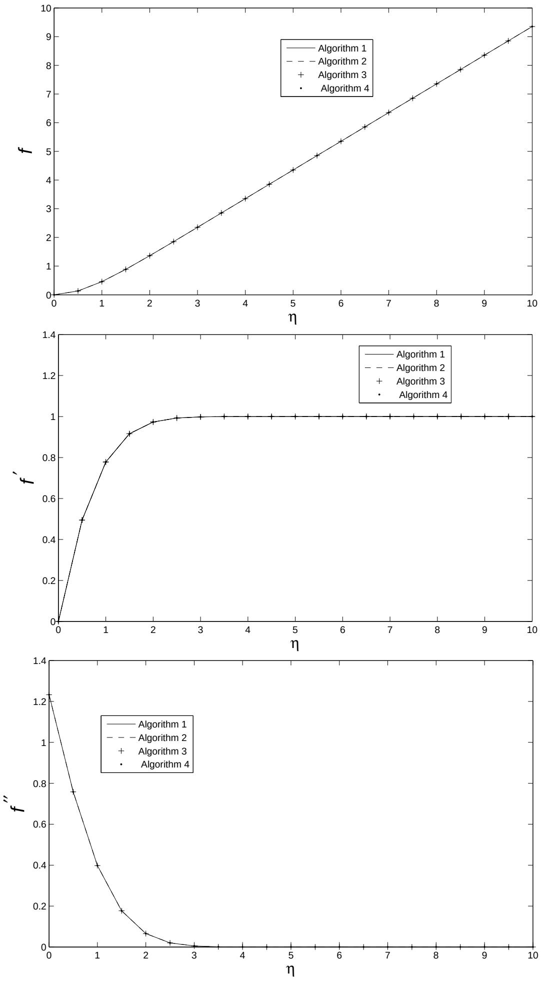

Figure 4: Problem 2: Comparison of f', f, f'' after 2nd time recursive iterations with $h = 0.001$.

Algorithms 1 and 3 took four iterations. This means that Algorithm 1 and 3 requires twice computational time as compared to Algorithms 2 and 4. Therefore, Algorithms 2 and 4 are proven optimal algorithms to obtain initial guesses in the shooting method for solving the BVP.

4. Conclusion

This paper proposes the idea of using a family of higher order initial approximation algorithms to refine initial guesses for a shooting method to solve BVPs. Faster convergence was achieved and solutions approached the right end boundary rapidly by implementing the

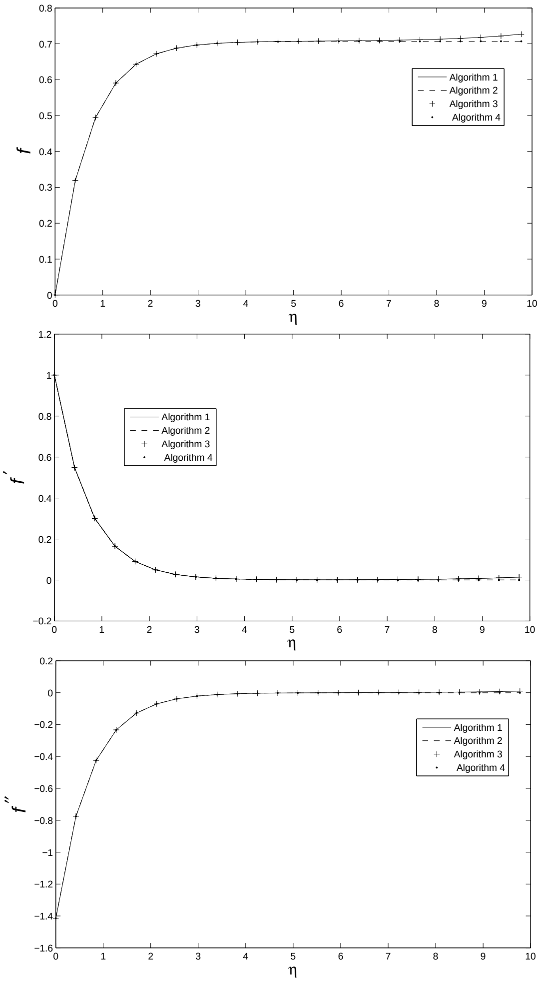

Figure 5: Problem 3: Comparison of f', f, f'' after two times recursive iterations with $h = 0.0025$.

aforementioned algorithms. In each case, the given BVP was reformulated as an IVP with one unknown initial condition. We applied three different algorithms to better approximate initial guesses instead of using conventional methods, such as Newton's method. Afterwards, the system of equations was solved using the shooting method. The suggested technique was applied to solve different nonlinear higher order BVPs. On the basis of results, we concluded that the suggested Algorithms 2 and 4 accurately reached the target in almost half number of iterations as compared to Algorithms 1 and 3. Thus, the proposed shooting technique with Algorithms 2 and 4 was proven more efficient and accurate than Algorithms 1 and 3. Furthermore, the results of the proposed method are accurate as compared to the

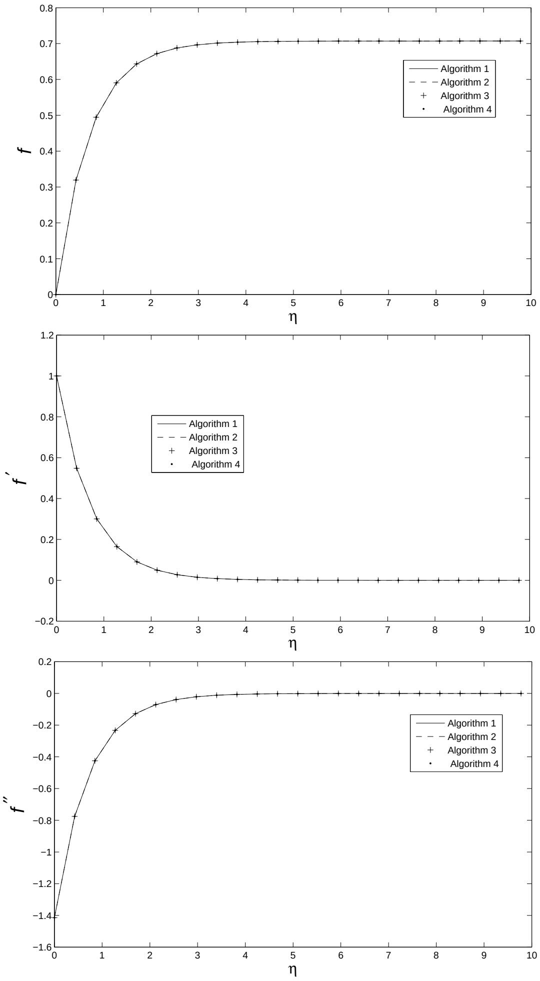

Figure 6: Problem 3: Comparison of f', f, f'' after four times recursive iterations with $h = 0.0025$.

results obtained from build-in-functions such as, ode45, ode113, ode15s, ode23s, ode23t, ode23tb and ode15i which are being used in MATLAB. The current study recommends the inclusion of using better initial approximation algorithms when using shooting methods for solving BVPs. Future research extends the implementation of the suggested method for solving coupled systems of equations.

5. Author Contributions:

Conceptualization, S.J; Methodology, S.J. and A.S; Software, A.S; Validation, S.J, A.S. and D. B; Writingoriginal draft preparation, S.J and A.S; Writingreview and editing, D.B.; Supervision, D.B.

6. Acknowledgments

The authors would like to express profound gratitude to referees for deeper review of this paper and their useful suggestions.

References

[1] Abbasbandy, S. *Improving Newton−Raphson method for nonlinear equations by modified Adomian decomposition method*, Appl. Math. Comput., **145**(2), 887-893 (2003).

[2] Ahsan, M. and Farrukh, S. A new type of shooting method for nonlinear boundary value problems, Alexandria Engineering Journal, **52**(4), 801-805 (2013).

[3] S. Banach, *Sur les oprations dans les ensembles abstraits et leur application aux quations intgrales*, Fundamental Mathematics **3**, (1922) 133-181.

[4] S. Bashir, M. U. G. Khan, A. A. Shah and A. Nasir, *Towards Computational Model of Human Brain Memory*, Punjab Univ. J. Math. Vol. **46**, No. 2 (2014) 35-45.

[5] Burden, R.L., Faires, J.D. *Numerical Analysis*, 9th Edition, Brooks/Cole Cengage Publishing, USA(2010).

[6] Chun, C. *A new iterative method for solving nonlinear equations.* Appl. Math. Comput., **178**(2), 415-422 (2006).

[7] Granas, A., Guenther, R.B. and Lee, J.W. *The shooting method for the numerical solution of a class of nonlinear boundary value problems*, SIAM J. Numer. Anal., **16**(5), 828-836 (1979).

[8] Jones, D.J. *Use of a shooting method to compute eigenvalues of fourth-order two-point boundary value problems*, J. Comput. Appl. Math, **47**(3) 395-400, (1993).

[9] Jones, D.J. *Use of a shooting method to compute eigenvalues of fourth-order two-point boundary value problems*, J. Comput. Appl. Math., **47**(3), 395-400 (1993).

[10] Keller, H.B. *Numerical solution of two point boundary value problems*, 3rd ed., SIAM, Philadelphia (1976).

[11] Kwong, M.K. and Wong, J.S. *The shooting method and nonhomogeneous multipoint BVPs of second-order ODE*, Bound. Value Probl., **1**, 1-16 (2007)

[12] Mahmoud, M.A., Mahmoud, M.A.E. and Waheed, S.E. *Hydromagnetic boundary layer micropolar fluid flow over a stretching surface embedded in a non-Darcian porous medium with radiation*, Math. Probl. Eng., Article ID 39392, 10 Pages (2006).

[13] Morrison, D.D., Riley, J.D. and Zancanaro, J.F. *Multiple shooting method for two-point boundary value problems*, Commun. ACM,**5**(12), 613-614 (1962).

[14] Noor, M.A., Ahmad, F. and Javeed, S. Two-step iterative methods for nonlinear equations. *Appl. Math. Comput.*, **181**(2), 1068–1075 (2006).

[15] Noor, K.I. and Noor, M.A. Iterative methods with fourth-order convergence for nonlinear equations. *Appl. Math. Comput.*, **189**(1), 221-227 (2007).

[16] Oderinu, R.A. and Aregbesola, Y.A.S. *Shooting Method via Taylor Series for Solving Two Point Boundary Value Problem on an Infinite Interval*, Gen. Math. Notes, **24**(1), 74-83 (2014).

[17] Pozrikidis, C. *Fluid dynamics: theory, computation, and numerical simulation*, 2nd ed., Springer Science + Business Media, USA,489-490 (2009).

[18] Russell, R.D. and Shampine, L.F. *Numerical methods for singular boundary value problems*, SIAM J. Numer. Anal., **12**(1), 13-36 (1975).

[19] Shah, F.A., Noor, M.A. and Waseem, M. *Some second-derivative-free sixth-order convergent iterative methods for non-linear equations*, Maejo Int. J. Sci. Technol., **10**(1), 79-87 (2016).

[20] Schlichting, H. and Gersten, K. *Boundary-layer theory*, 8th ed., Springer Science & Business Media, Germany (2003).

[21] Wang, H., Ouyang, Z. and Wang, L. *Application of the shooting method to second-order multi-point integral boundary-value problems*, Boundary Value Problems. Bound. Value Probl., **1**, 1-10 (2013).