Jurnal KALAM Jurnal Karya Asli Lorekan Ahli Matematik Vol. 9 No.2 (Dec 2016) Page 18-23

Jurnal
Karya Asli Lorekan
Ahli Matematik

COMPARATIVE ANALYSIS OF THE FAMILY OF NEWTON-HSSOR ITERATIVE METHODS FOR THE SOLUTION OF NONLINEAR SECOND-ORDER TWO-POINT BOUNDARY VALUE PROBLEMS

Jumat Sulaiman[1], Andang Sunarto[1], Mohd Khatim Hasan[2], Mohamed Othman[3] and Samsul Ariffin Abdul Karim[4]

[1] *Mathematics with Economics Programme, Faculty of Science and Natural Resources, Universiti Malaysia Sabah, 88400 Kota Kinabalu, Sabah, Malaysia*
[2] *Centre of Artificial Intelligence Technology , Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia, 43600 UKM Bangi, Selangor, Malaysia*
[3] *Dept. of Communication Technology and Network, Faculty of Computer Science and Information Technology, Universiti Putra Malaysia, 43400 UPM Serdang, Selangor, Malaysia*
[4]*Dept. of Fundamental and Applied Sciences, Faculty of Science and Information Technology, Universiti Teknologi Petronas, 31750 Tronoh, Perak, Malaysia.*
[1]jumat@ums.edu.my, [2]khatim@ftsm.ukm.my, [3]mothman@fsktm.upm.edu.my [4]samsul_ariffin@petronas.com.my

Abstract: The purpose of this paper is to investigate the comparative analysis of HSSOR iteration family together with Newton scheme in solving the nonlinear systems generated from the discretization process of second order two-point nonlinear boundary value problems via The half-sweep finite difference scheme. In order to get the numerical solution of the generated nonlinear systems, firstly the Newton scheme is used to linearize the nonlinear system into the form of linear system. In addition to that, the basic formulation and implementation of the HSSOR iteration family are also shown. Consequently numerical results of these iterative methods based on three examples have been compared to demonstrate the validity and applicability of tested methods. Clearly, The findings show that the Red-Black-Newton-HSSOR method indicates the superiority over other tested methods.

Keywords: Two-point nonlinear boundary value problems, Finite Difference, HSSOR iteration, Newton scheme

Introduction

Boundary value problems play an important role in mathematical models used to simulate many branches of applied mathematics and physics such as gas dynamics, quantum mechanics, fluid dynamics, aerodynamics, chemical reactions, atomic structures, atomic calculations etc. Most phenomena in these problems, however, have been modeled by nonlinear differential equations. To get the numerical solutions of these problems, several purpose methods have been proposed to solve the nonlinear problems. For instance, in solving nonlinear two-point boundary value problems, among these methods are numerical analytic, finite difference, finite element, finite volume and boundary element methods. In this paper, however, we deal with the application of the finite difference method in order to develop a reliable algorithm in solving nonlinear two-point boundary value problems. By using the finite difference methods to derive a nonlinear approximation equation, a nonlinear system can be generated and need to be linearized through the Newton method in order to form the corresponding linear system. Since the characteristics of linear systems are large and sparse, iterative methods are the natural options for efficient solutions.

To get numerical solutions of linear systems, various iterative methods also have been initiated to solve linear systems (see Young [1,2,3]; Hackbusch [4]; Saad [5]). Apart from those iterative methods, the concept of the half-sweep iterative method has been introduced by Abdullah [6] via Explicit Decoupled Group (EDG) iterative method in solving two-dimensional Poisson equations. The advantage of this iteration concept is to reduce the computational complexity of full-sweep linear systems generated from corresponding approximation equations. Due to the low computational complexity, this concept has been widely used to demonstrate its capability in constructing a fast reliable algorithm [7,8,9,10,11,12]. Besides

Published by Pustaka Aman Press Sdn. Bhd.

these one-stage iteration concepts, several studies have been conducted to combine between half-sweep iteration concept with two-stage iterative methods such as HSIADE [13], HSAM [14] and HSGM [15] in order to solve linear systems. Therefore, they pointed out that their proposed two-stage iterative methods are one of most efficient iterative methods in solving any system of linear equations. In addition to one- and two stage iterative methods, the standard multigrid methods have been modified by introducing a family of half-sweep multigrid methods [16,17]. Consequently, Hassan et al. [18,19] have also established a family of FDTD methods using this concept in solving wave propagation problems. Then Saudi and Sulaiman [20,21] applied this half-sweep iteration to solve the robotic path planning.

In this paper, we conduct the performance analysis of Red-Black HSSOR iteration together with Newton scheme, which is identified as Newton-RBHSSOR in solving a nonlinear second-order two-point boundary value problem. For comparison purpose, this method will be compared with Full-Sweep Gauss-Seidel (FSGS), Full-Sweep SOR (FSSOR) and Half-Sweep SOR (HSSOR) iterative methods with Newton scheme namely Newton-FSGS, Newton-FSSOR, and Newton-HSSOR respectively. To examine the performance of these proposed iterative methods, let us consider a nonlinear second-order two-point boundary value problem defined as

$$-\frac{d^2U}{dx^2} = g(x,U,U'),\ a \le x \le b \qquad 1)$$

subject to the boundary conditions

$$U(a) = \beta_0,\ \ U(b) = \beta_1$$

where β_0 and β_1 are constants and $g(x,U)$ is a nonlinear continuous function. For the sake of simplicity, we shall restrict our discussion onto uniform node points only as shown in Figure 1. Let the solution domain (1) be uniformly divided into $m = 2^p,\ p \ge 2$ subinterval in which its distance, Δx is defined in Eq. (2).

$$\Delta x = \frac{(b-a)}{m} = h,\ n = m - 1 \qquad 2)$$

a).

b).

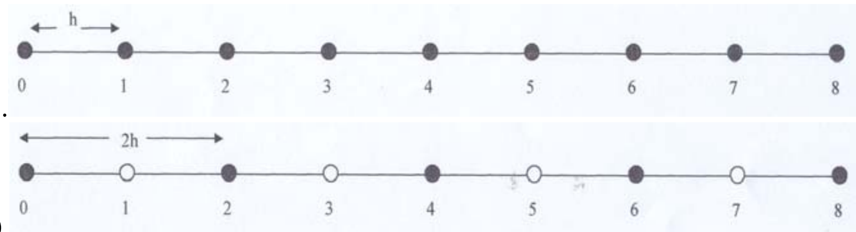

Figure 1. a) and b) show the distribution of uniform node points for the full- and half-sweep cases respectively at m = 8.

Based on Figure 1, it can be shown that the finite grid networks can facilitate us for the implementation of these proposed iteration algorithms. In case of half-sweep iteration, the half-sweep finite grid network involve points of type ● and ○. In fact, the implementation of the half-sweep point iteration will consider the interior node points of type ● until the iterative convergence fixed is achieved. Due to the advantage of half-sweep approach, we examine the efficiency of the Newton-RBHSSOR iterative method for solving nonlinear second-order two-point boundary value problems by using the corresponding second-order finite difference approximation equation. The further explanation of the finite difference approximation equation will be enlightened in the next section.

Second-Order Half-Sweep Nonlinear Finite Difference Approximation Equation

To get the numerical solution of problem (1) iteratively, first we need to discretize the proposed problem. Therefore, let us impose the second-order central difference discretization scheme over problem (1) to derive the second-order nonlinear finite difference approximation equation

$$U_{i-1} - 2U_i + U_{i+1} - h^2 g(x_i, U_i, \frac{U_{i+1} - U_{i-1}}{2h}) = 0, \qquad a)$$

for $i = 1, 2, 3, \cdots, n$. Actually, Eq. (3) is called as the second-order full-sweep nonlinear finite difference approximation. Similarly to derive Eq. (3) and using the second-order half-sweep central difference discretization scheme, the half-sweep nonlinear finite difference approximation equations [22] can be shown as

$$U_{i-2} - 2U_i + U_{i+2} - 4h^2 g(x_i, U_i, \frac{U_{i+2} - U_{i-2}}{4h}) = 0, \quad \text{b)}$$

for $i = 2, 4, 6, \cdots, n-2$. Based on Eqs. (3) and (4), let us define the nonlinear function, f in general form as

$$f_i\left(U_{1p}, U_{2p}, \cdots, U_{n-p}\right) = U_{i-p} - 2U_i + U_{i+p} - p^2h^2 g(x_i, U_i, \frac{U_{i+p} - U_{i-p}}{2ph}) \quad \text{c)}$$

for $i = 1p, 2p, 3p, \cdots, ((n+1)/p) - 1$, where the value of p in Eq. (5), which equals to 1 and 2 represents the full- and half-sweep cases respectively. By considering all interior node points in the solution domain (1), it can be shown that Eq. (5) leads a nonlinear system, which can be expressed as

$$\left.\begin{array}{l} f_{1p}\left(U_{1p}^{(k)}, U_{2p}^{(k)}, \cdots, U_{((n+1)/p)-1}^{(k)}\right) = 0, \\ f_{2p}\left(U_{1p}^{(k)}, U_{2p}^{(k)}, \cdots, U_{((n+1)/p)-1}^{(k)}\right) = 0, \\ \vdots \\ f_{((n+1)/p)-1}\left(U_{1p}^{(k)}, U_{2p}^{(k)}, \cdots, U_{((n+1)/p)-1}^{(k)}\right) = 0, \end{array}\right\} \quad \text{d)}$$

where, $U_i^{(k)}, i = 1p, 2p, \cdots, ((n+1)/p) - 1$ indicate as the k[th] estimation for corresponding exact solutions. Before solving the nonlinear system (6) by using any linear solvers, we impose the Newton method over the original nonlinear system in order to form the corresponding linear system which can be easily shown in the matrix form as

$$J\left(\underset{\sim}{U}^{(k)}\right) \underset{\sim}{\Delta h}_j = -f_j\left(\underset{\sim}{U}^{(k)}\right) \quad \text{e)}$$

where,

$$J\left(\underset{\sim}{U}^{(k)}\right) = \begin{bmatrix} \dfrac{\partial f_{1p}}{\partial u_{1p}} & \dfrac{\partial f_{1p}}{\partial u_{2p}} & \cdots & \dfrac{\partial f_{1p}}{\partial u_{((n+1)/p)-1}} \\ \dfrac{\partial f_{2p}}{\partial u_{1p}} & \dfrac{\partial f_{2p}}{\partial u_{2p}} & \cdots & \dfrac{\partial f_{2p}}{\partial u_{((n+1)/p)-1}} \\ \vdots & \vdots & \ddots & \vdots \\ \dfrac{\partial f_{((n+1)/p)-1n}}{\partial u_{1p}} & \dfrac{f_{((n+1)/p)-1}}{\partial u_{2p}} & \cdots & \dfrac{\partial f_{((n+1)/p)-1}}{\partial u_{((n+1)/p)-1}} \end{bmatrix},$$

$$\underset{\sim}{\Delta h}_j = \left[\Delta h_{1p}, \Delta h_{2p}, \Delta h_{3p}, \ \ldots \ , \Delta h_{((n+1)/p)-1}\right]^T.$$

From Eq. (7), the Jacobian matrix, $J\left(\underset{\sim}{U}^{(k)}\right)$ is a coefficient matrix of the linear system. Therefore, the value of the vector, $\underset{\sim}{\Delta h}_j$ needs to be calculated by solving the linear system. Then estimate solutions of $\underset{\sim}{U}^{(k)}$ can be determined by using the following expression

$$U_i^{(k+1)} = U_i^{(k)} + \Delta h_i, \quad i = 1p, 2p, \cdots, ((n+1)/p) - 1 \quad \text{f)}$$

FORMULATION OF Newton-RBHSSOR Method

For simplicity, consider the linear system (7) be rewritten in general form as

$$A\underset{\sim}{U} = \underset{\sim}{F} \tag{a}$$

Clearly it can be observed that since the number of subintervals, $m = 2^p$, $p \geq 2$ is relatively a large positive integer number, this coefficient matrix of this linear system (9) is also large and sparse. According to Young [2] and other works [15,16], they pointed out that the iterative methods are the natural options for efficient solutions of sparse linear system. For the reason, we begin by considering four proposed iterative methods such as FSGS, FSSOR HSSOR and RBHSSOR as linear solvers. To apply these methods, firstly, we need to discuss on how to construct the formulation of the standard SOR (FSSOR) method, which is known as the Gauss-Seidel method with a weighted parameter, ω. In fact, this method is used to speed up the convergence rate of the standard Gauss-Seidel (GS) method for solving any linear system. Essentially, Young [1,2,3] has initiated the standard Successive Over-Relaxation (SOR) method, which is also called as Full-Sweep SOR (FSSOR). This iterative method is one of the most known and widely used iterative techniques to solve any linear systems. Based on this method, the formulation of the RBHSSOR method is derived from a combination between the HSSOR method with the Red-Black ordering strategy. Now we begin on how to derive the formulation for FSSOR and HSSOR iterative methods. To do this, let the coefficient matrix, A in Eq. (9) be decomposed as

$$A = D + L + V \tag{b}$$

where L, D and V are lower triangular, diagonal and upper triangular matrices respectively. By using the decomposition in Eq. (10) and determining values of matrices D, L and V, therefore, the general scheme for FSSOR and HSSOR methods can be stated as [1,2,3,20,22]

$$\underset{\sim}{U}^{(k+1)} = (1-\omega)\underset{\sim}{U}^{(k)} + \omega(D+L)^{-1}\left(-V\underset{\sim}{S}^{(k)} + \underset{\sim}{F}\right) \tag{c}$$

where ω and $\underset{\sim}{U}^{(k)}$ represent as a relaxation factor and an unknown vector at the k[th] iteration respectively. The choice of relaxation factor depends upon the properties of the coefficient matrix, A. In addition, a good choice for the value of the parameter ω can be used to accelerate the convergence rate of iteration process. In practice, the optimal value of ω in range $1 \leq \omega < 2$ will be obtained by implementing several computer programs and then the best approximate value of ω is chosen in which its number of iterations is the smallest.

Since the concept of the RBHSSOR method is said as HSSOR method with red-black ordering strategy, the general scheme for the RBHSSOR method can be derived by using Eq.(11). To implement this method, the general algorithm for the RBHSSOR iterative methods to solve problem (9) would be generally described in Algorithm 1.

Algorithm 1: RBHSSOR scheme

i. Initialize

$$U_i^{(0)} \leftarrow 0, \quad \varepsilon \leftarrow 10^{-10}$$

ii. Assign the optimal value of ω and p=2

iii. Calculate $U_i^{(k+1)}$ from

$$\underset{\sim}{U}^{(k+1)} \leftarrow (1-\omega)\underset{\sim}{U}^{(k)} + \omega(D+L)^{-1}\left(-V\underset{\sim}{S}^{(k)} + \underset{\sim}{F}\right), \quad i = 1p, 3p, \cdots, (((n+1)/p)-1)$$

$$\underset{\sim}{U}^{(k+1)} \leftarrow (1-\omega')\underset{\sim}{U}^{(k)} + \omega'(D+L)^{-1}\left(-V\,\underset{\sim}{S}^{(k)} + \underset{\sim}{F}\right), \quad i = 2p, 4p, \cdots, (((n+1)/p))$$

iv. Check the convergence test, $\left|U_i^{(k+1)} - U_i^{(k)}\right| \le \varepsilon$. If yes, go to step (iv). Otherwise go back to step (iii)

v. Display approximate solutions.

.

NUMERICAL RESULTS

To examine the effectiveness of the the FSGS, FSSOR, HSSOR and RBHSSOR methods together with Newton scheme namely Newton-FSGS, Newton-FSSOR, Newton-HSSOR and Newton-RBHSSOR respectively, three nonlinear examples of the problems were tested. For comparison purpose, we take into account three criteria such as number of iterations, execution time and maximum absolute error. These criteria will be recorded during the numerical experiments of the following three examples. In the implementation of the iterative methods, the convergence test considered the tolerance error, $\varepsilon = 10^{-10}$.

Example 1 (Sung [24])
For comparison purpose, we consider the following nonlinear two-point boundary value problem

$$\frac{d^2U}{dx^2} = \frac{3}{4}U^2, \qquad x \in [0,1] \qquad \text{g)}$$

Subject to the boundary conditions

$$U(0) = 4, \qquad U(1) = 1.$$

Then boundary conditions and the exact solution of the problem (12) were defined by

$$U(x) = \frac{4}{(1+x)^2}, \qquad 0 \le x \le 1 \qquad \text{h)}$$

Example 2 (Sung [24])
Let consider the following problem

$$\frac{d^2U}{dx^2} = U^3 - UU', \qquad x \in [1,2] \qquad \text{i)}$$

with the boundary conditions

$$U(1) = \frac{1}{2}, \qquad U(2) = \frac{1}{3}.$$

Then boundary conditions and the exact solution of the problem (14) were defined by

$$U(x) = \frac{1}{x+1}, \qquad 1 \le x \le 2 \qquad \text{j)}$$

Example 3 (Sung [24])
The third nonlinear boundary value problem, we consider as follows

$$\frac{d^2U}{dx^2} = \frac{32 + 2x^3 - UU'}{8}, \qquad x \in [1,3] \qquad \text{k)}$$

Subject to the boundary conditions

$$U(1) = 17, \qquad U(3) = \frac{14}{3}.$$

Then boundary conditions and the exact solution of the problem (16) were defined by

$$U(x) = x^2 + \frac{16}{x}, \qquad 1 \le x \le 3 \qquad \text{l)}$$

For all above examples, results of numerical experiments obtained from implementation of the Newton-FSGS, Newton-FSSOR, Newton-HSSOR and Newton-RBHSSOR iterative methods, have been recorded in Table 1.

Conclusion

In this paper, we present the formulation of the the FSGS, FSSOR, HSSOR and RBHSSOR methods together with Newton scheme namely Newton-FSGS, Newton-FSSOR, Newton-HSSOR and Newton-RBHSSOR respectively for solving the corresponding nonlinear systems, which are generated from the corresponding second-order nonlinear finite difference approximation equations of the proposed problem (1). To demonstrate the efficiency of the four proposed iterative methods, three nonlinear examples are presented. Throughout the numerical results obtained in Table 1, it can be concluded that the Newton-RBHSSOR method is superior in terms of the number of iterations and the execution time for five different mesh sizes, $m = 256, 512, 1024, 2048, 4096$ as compared with Newton-FSSOR and Newton-HSSOR methods. This is because of the computational complexity of Newton-RBHSSOR method is approximately 50% compared to the full-sweep case. Since Newton-RBHSSOR method involve two accelerated parameters in the iteration process, its convergence rate is better than the Newton-HSSOR method. Overall, the approximate solutions for all four proposed methods are in good agreement. It means that the Newton-RBHSSOR method with second-order nonlinear finite difference approximation equations is a promising approach for solving the proposed nonlinear problems. For future works, the Newton-RBHSSOR method can be extended to solve multi-dimensional nonlinear problems.

References

1. D.M. Young, *Trans. Amer. Math. Soc.* 76, 92-111 (1954).
2. D.M. Young, *Iterative solution of large linear systems.* London: Academic Press, 1971.
3. D.M. Young, *Journal of Approximation Theory.* 5, 37-148 (1972).
4. W. Hackbusch. *Iterative solution of large sparse systems of equations*, New York: Springer-Verlag, 1995.
5. Y. Saad. *Iterative method for sparse linear systems*. Boston: International Thomas Publishing, 1996.
6. A.R. Abdullah. *Intern. J. of Computer Mathematics*, 38, 61-70 (1991).
7. A.R. Abdullah and N.H.M. Ali. *Parallel Algorithms and Applications*, 10, 93-103 (1996).
8. E. Aruchunan and J. Sulaiman, 2011. *Aust. J. of Bas. Appl. Sci.*, 5, 38-43 (2011).
9. A. Ibrahim and A.R. Abdullah. *Intern. J. of Computer Mathematics*. 58, 253-256 (1995).
10. J. Sulaiman, M.K. Hasan and M. Othman, *Lecture Notes in Computer Science* (LNCS 4707), 298-308 (2007).
11. J. Sulaiman, M.K. Hasan and M. Othman, *Lecture Notes in Computer Science* (LNCS 4487), 326-333 (2007).
12. W.S. Yousif and D,J, Evans, *Parallel Algorithms and Applications*, 7, 53-71 (1995).
13. J. Sulaiman, M.K. Hasan and M. Othman. *Lecture Note on Computer Science* (LNCS 3314), 57-63 (2004).
14. M.S. Muthuvalu and J. Sulaiman. *Applied Mathematics and Computation,* 217(12), 5442-5448 (2011).
15. M.S. Muthuvalu and J. Sulaiman. *Proyecciones Journal of Mathematics*, 31(1), 65-79 (2012).
16. M. Othman, J. Sulaiman and A.R. Abdullah. *Malaysian Jour. of Computer Science*, 13(2), 1-6 (2000).
17. J. Sulaiman, M. Othman and M.K. Hasan, "Half-Sweep Algebraic Multigrid (HSAMG) method applied to diffusion equations" in *Modeling, Simulation and Optimization of Complex Processes,* edited by H.G. Bock et al. Berlin: Springer-Verlag, 2008,pp. 547-556 .
18. M.K. Hasan, M.Othman, Z.Abbas, J. Sulaiman, and F. Ahmad. *Lecture Notes in Computer Science* (LNCS 4706), 13-24 (2007).
19. M.K. Hasan, M. Othman, Z. Abbas, J. Sulaiman and R. Johari, "The HSLO(3)-FDTD With Direct-Domain and Temporary-Domain Approaches On Infinite Space Wave Propagation", Proceedings of the 7th IEEE Malaysia International Conference on Communications and the 13th IEEE International Conference on Networks (MICC-ICON 2005), 2005, pp. 1002-1007.
20. A. Saudi and J. Sulaiman, "Path Planning for Mobile Robot with Half-Sweep Successive Over-Relaxation (HSSOR) Iterative Method", Symposium on Progress in Information and Communication Technology (SPICT'09), 2009, pp.57-62.
21. A. Saudi and J. Sulaiman, *Lecture Notes in Engineering and Computer Science*, 2182(1), 2215-2220 (2010).
22. J. Sulaiman, M.K. Hasan, M. Othman and S.A.A. Karim. *Journal of Concrete and Applicable Mathematics,* 11(1), 112-120 (2013).
23. D.R. Kincaid and D.M. Young, *Mathematics of Computations*, 26(119), 705-717 (1972).
24. N.H. Sung. *An International Journal Computers And Mathematics With Applications*, 42, 1411-1420 (2001).