Cubic Non-Polynomial Solution for Solving Two-Point Boundary Value Problems Using SOR Iterative Method

Hynichearry Justine*, Jumat Sulaiman

Faculty of Science and Natural Resources, Universiti Malaysia Sabah, Jalan UMS, 88400 Kota Kinabalu, Sabah, MALAYSIA.
*Corresponding author. E-Mail: cherryjust20@gmail.com

ABSTRACT: Two-point boundary values problems for certain order of derivatives are cases where only the initial and final values are known. These problems were normally solved using systems of ordinary or partial differential equations, and have wide applications in modelling of most physical phenomena as well as in economics. Previous investigations have shown the implementation of solutions for two-point boundary value problems by using polynomial spline approximation scheme. In this paper, non-polynomial spline approximation scheme is used where the general functions of cubic non-polynomial spline was employed to discretize the two-point boundary value problems to generate approximation equations which yield to its corresponding linear system in matrix form. Successive Over-Relaxation (SOR) iterative method was then used to solve the problem together with Gauss-Seidel (GS) as reference to assess the performance result of non-polynomial spline approximation scheme in respect of its number of iteration, execution time and maximum absolute error when solving the two-point boundary value problems. It was found that SOR iterative method has performed better compared to GS for all different grid sizes as shown through the improvement of its respective number of iteration, execution time and maximum absolute error. Therefore, SOR iterative method is an efficient approach for solving the two-point boundary value problems.

KEYWORDS: Cubic non-polynomial solution; Successive Over Relaxation; Gauss-Seidel; Two-point boundary value problems

Received 6 April 2016 **Revised** 28 May 2016 **Accepted** 6 June 2016 **Inpress** 4 July 2016 **Online** 20 December 2016

TRANSACTIONS ON SCIENCE AND TECHNOLOGY

INTRODUCTION

Obtaining approximate solutions for two-point boundary value problems play a vital role in solving many problems in the field of sciences, economics and engineering. This is due to its various applications which including modelling of chemical reactions, modelling of heat transfer (Ozisik, 1989) such as in rocket thrust chamber liners and modelling of heat transfer in the fuel elements for nuclear reactors. Similar applications can be found in economic including modelling of growth theory, capital theory, investment theory, resource economics and labor economics (Goffe, 1993). As a consequence, many techniques have been initiated by researchers such as EADM method (Jang, 2006), shooting method (Sung, 2001) and PTI method by Chen *et al*. (2006). In addition to that, Fang *et al.* (2002), stated that some of the solutions are finite difference, finite element and finite volume. In 2001, Sung introduced a solution known as nonlinear shooting method. Other than that, spline approach also one of the solutions which has captured countless researchers' field of exploration in order to solve the two-point boundary problems (Albasiny & Hoskin, 1969; Ramadan *et al*., 2007).

In this paper, cubic non-polynomial spline functions were used to discretize two-point boundary value problems based on spline approximation equations in order to derive cubic non-polynomial spline approximation equations. Then, these approximation equations yield their own corresponding large and sparse linear system. To solve the linear system, various iterative methods have been proposed and discussed by Young (1954, 1971, 1972, 1976), Hackbusch (1995) and Saad (1996). From the previous studies of iterations, there exist several families of iterative methods. In addition to these iterative methods, the concept of block iteration has also been introduced by Evans (1985), and furthermore, explanation of this block iterative concept has been extended by Ibrahim and Abdullah (1995), Yousif and Evans (1995), in which these block iterative methods can be one of the efficient iterative methods.

Assuming the advantages of Successive Over-Relaxation (SOR) method (Young, 1954; 1971; 1972; 1976), this scheme is examined in solving the two-point boundary value problems of the cubic non-polynomial spline. Then, for assessment, Gauss-Seidel (GS) iterative method is used as reference.

The two-point boundary problems can be expressed as follows;

$$y''+f(x)y'+q(x)y = g(x), \; x \in [a,b] \tag{1}$$

subject to boundary conditions

$$y(a) = A_1, \; y(b) = A_2 \tag{2}$$

where A_i, $i=1,2$ is constant and functions $f(x)$ and $g(x)$ are known function with boundary $[a,b]$. Then, analitical solution for problem (1) is depending on baundary conditions (2) and cannot be created with random selection of functions, $f(x)$ and $g(x)$.

To facilitate us in discretizing problem (1), let the solution domain of the problem be divided uniformly. To do this, we can consider any positive integer $m = 2^p$, where $p \geq 2$ and then let the solution domain, $[a,b]$ be divided uniformly into m subinterval (Figure 1) in which the length of uniformly subintervals, Δx is defined as

$$\Delta x = \frac{b-a}{m} = h, \quad n = m-1. \tag{3}$$

Therefore, the grid points in the solution domain $[a,b]$ are labeled as the numbers $x_i = a + ih, i = 0,1,2,\cdots,m$. Then, the values of the function $y(x)$ at the grid points are denoted as $y_i = y(x_i)$. Formulation and implementation of GS and SOR iterative methods have been conducted by using interior grid points until the test of convergence rate can be satisfied.

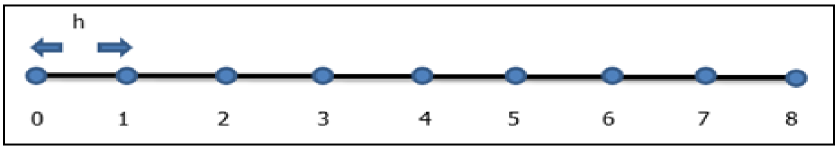

Figure 1. Distribution of node point for domain solution m=8

CUBIC NON-POLYNOMIAL SPLINE APPROXIMATION EQUATION

To construct a cubic non-polynomial spline approximation equation, firstly, problem (1) needs to be discretized by using the cubic non-polynomial spline scheme. To do this, let $y(x)$ be the exact solution of problem (1) and S_i be non-polynomial spline approximation to $y_i = y(x_i)$ acquire by the segments of $Q_i(x)$ are passing through to the points (x_i, S_i) and (x_{i+1}, S_{i+1}). Then the non-polynomial spline approximation in general form can be considered as

$$S(x) = Q_i(x), \quad x \in [x_i, x_{i+1}], \quad i = 0, 1, 2, \cdots, n \tag{4}$$

TRANSACTIONS ON SCIENCE AND TECHNOLOGY

Let us consider the cubic non-polynomial spline from the equation (4) in $Q_i(x)$ general form be defined as

$$Q_i(x) = a_i \cos k(x - x_i) + b_i \sin k(x - x_i) + c_i(x - x_i) + d_i \quad (5)$$

for $i = 0, 1, 2, \cdots, n$ where a_i, b_i, c_i and d_i are constant, and k is the frequency for the trigonometric function and the equation (5) is known as the general form of cubic non-polynomial spline when $k \to 0$ (Saudi & Sulaiman, 2009).

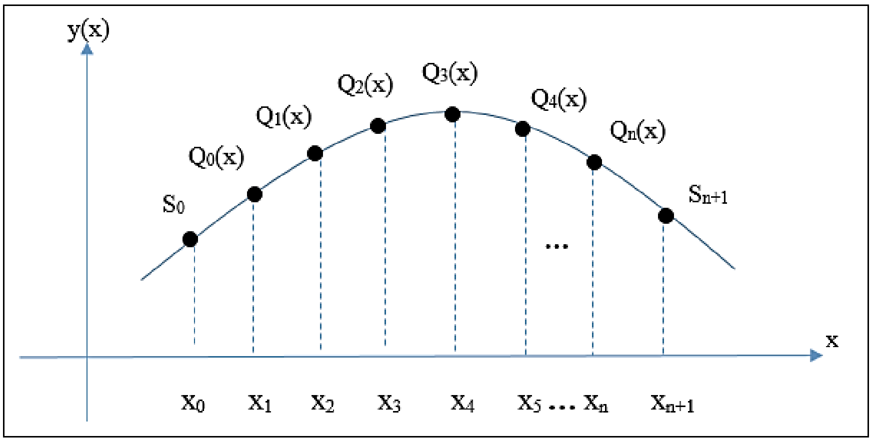

Figure 2. Illustration of cubic non-polynomial spline function for the domain solution $m = 8$

By referring to Figure 2, in order to formulate the approximation of cubic non-polynomial spline equation, discretization process is very important to be done first as previously discussed. This paper presents the approximation of problem (1) through the discretization of cubic non-polynomial spline. Assume that y_i is an accurate solution obtained from many segment of spline function that passing through point $(x_i y_i)$ and $(x_{i+1,}\ y_{i+1})$, then, in order to obtain the expression of constant variables for equation (5) in form of $y_{i,} y_{i+1,}\ D_{i,} D_{i+1,} S_{i,,}\ S_{i+1}$, the functions have to be defined as $Q_i(x_i) = y_i$, $Q_i(x_{i+1}) = y_{i+1}$, $Q_i{}'(x_i) = D_i$, $Q_i{}'(x_{i+1}) = D_{i+1}$, $Q_i{}''(x_i) = S_i$, $Q_i{}''(x_{i+1}) = S_{i+1}$. By doing several algebraic manipulation via the substitution technique, the expressions of constants, a_i, b_i, c_i and d_i were obtained as $a_i = h^2 \dfrac{-S_{i+1} + S_i \cos(\theta)}{\theta^2 \sin(\theta)}$, $b_i = -h^2 \dfrac{S_i}{\theta^2}$, $c_i = \dfrac{y_{i+1} - y_i}{h} + h\dfrac{(S_{i+1} + S_i)}{\theta^2}$, $d_i = y_i + h^2 \dfrac{S_i}{\theta^2}$ where $\theta = kh$ and $i = 0,1,2,..,N.$

Then, after all points a_i, b_i, c_i and d_i which passing through point $(x_i y_i)$ were obtained, let us consider the following condition $Q_{i-1}^m(x) = Q_i^m(x)$ where $m = 0,1$ by solving this part simultaneously, we can get the following cubic non-polynomial spline approximation equation as

$$y_{i-1} - h^2 S_{i-1}\left(\frac{1}{\theta \sin\theta} - \frac{1}{\theta^2}\right) - 2y_i - 2h^2 S_i\left(\frac{1}{\theta^2} - \frac{\cos\theta}{\theta \sin\theta}\right) + y_{i+1} - 2h^2 S_i\left(\frac{1}{\theta \sin\theta} - \frac{1}{\theta^2}\right) = 0 \quad (6)$$

Again, equation (6) can be simplified as

$$-y_{i-1} + 2y_i - y_{i+1} + h^2[\alpha S_{i-1} + \beta S_i + \alpha S_{i+1}] = 0 \quad (7)$$

where $\alpha = [\dfrac{1}{\theta \sin\theta} - \dfrac{1}{\theta^2}]$, $\beta = [\dfrac{1}{\theta^2} - \dfrac{\cos\theta}{\theta \sin\theta}]$ and $i = 1,..,N.$

The equation of central finite difference, backward finite difference and forward finite difference as

$$S_{i-1} = -f_{i-1}y'_i - q_{i-1}y_{i-1} + g_{i-1},\ S_i = -f_i y'_i - q_i y_i + g_i,\ S_{i+1} = -f_{i+1}y'_{i+1} - q_{i+1}y_{i+1} + g_{i+1}, \tag{8}$$

where $y'_i = \dfrac{y_{i+1} - y_{i-1}}{2h}$, $y'_{i-1} = \dfrac{-y_{i+1} + 4y_i - 3y_{i-1}}{2h}$, $y'_{i+1} = \dfrac{3y_{i+1} - 4y_i + y_{i-1}}{2h}$

By substituting equation (8) into equation (7), the cubic non-polynomial spline approximation can be expressed as

$$a_i y_{i-1} + b_i y_i + c_i y_{i+1} = F_i,\ i = 1,2,\ldots,n \tag{9}$$

where

$$a_i = -\mu_0 + \gamma\left(\frac{3p_{i-1}}{2h} - q_{i-1}\right) + \theta\frac{p_i}{2h} - \gamma\frac{p_{i+1}}{2h},\ b_i = 2\mu_0 - \gamma\frac{2p_{i-1}}{h} - \theta q_i + \gamma\frac{2p_{i+1}}{h}$$

$$c_i = -\mu_0 + \gamma\frac{p_{i+1}}{2h} - \theta\frac{p_i}{2h} + \gamma\left(\frac{3p_{i+1}}{2h} - q_{i+1}\right),\ F_i = -\gamma f_{i-1} - \theta f_i \frac{}{2h} + \gamma f_{i+1}$$

$$\gamma = \mu_1 h^2,\ \phi = 2\mu_2 h^2,\ \mu_1 = \theta - \sin\theta,\quad \mu_2 = \sin\theta - \theta\cos\theta.$$

Then equation (9) can be used to construct a system of linear equations in matrix form as

$$A\underline{y} = \underline{F} \tag{10}$$

where

$$A = \begin{bmatrix} b_1 & c_1 & & & & \\ a_2 & b_2 & c_2 & & & \\ & a_3 & b_3 & c_3 & & \\ & & \ddots & \ddots & \ddots & \\ & & & a_{n-1} & b_{n-1} & c_{n-1} \\ & & & & a_n & bn_1 \end{bmatrix}_{(n\times n)},$$

$$\underline{y} = [y_1\quad y_2\quad y_3\quad \cdots\quad y_{n-1}\quad y_n]^T,\ \underline{F} = [F_1 - a_1 y_0\quad F_2\quad F_3\quad \cdots\quad F_{n-1}\quad F_n - c_n y_{n+1}].$$

DERIVATION OF SOR METHOD

As mentioned that the SOR iterative method is categorized as a family of point iterative methods. This iterative method was introduced by Young (1954, 1971, 1972, 1976) to accelerate the convergence rate of Gauss-Seidel (GS) iterative method. To derive the formulation for SOR iterative method, let the coefficient matrix, A in equation (10) be decomposed as

$$A = D + L + U \tag{11}$$

where L, D and T are lower triangular, diagonal and upper triangular matrices respectively. By imposing the decomposition in equation (11) into equation (10), the formulation of SOR iterative methods can be stated as

$$\underset{\sim}{y}^{(k+1)} = (1-\omega)\underset{\sim}{y}^{(k)} + \omega(D+L)^{-1}\left(-U\underset{\sim}{y}^{(k)} + \underset{\sim}{F}\right) \tag{12}$$

whereas, GS method is formulated as follows

$$\underset{\sim}{y}^{(k+1)} = -(D+L)^{-1}U\underset{\sim}{y} + (L+D)^{-1}F \tag{13}$$

To accelerate the convergence rate of this method, a good choice for the value of the parameter ω must be determined. In practice, the optimal value of ω in range $1 \le \omega < 2$ can 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. As taking, $\omega = 1$, the SOR iterative method can be reduced to GS iterative method. In this study, the GS iterative method is assigned to be used as control methods. Therefore, the general algorithm for the SOR iterative method in equation (12) would be described in Algorithm 1.

Algorithm 1 : SOR scheme

i. Initialize $U_i^{(0)} \leftarrow 0,\ \varepsilon \leftarrow 10^{-10}$
ii. Assign the optimal value of ω
iii. Calculate $U_i^{(k+1)}$ using
$\underset{\sim}{S}^{(k+1)} = (1-\omega)\underset{\sim}{S}^{(k)} + \omega(D+L)^{-1}\left(-U\underset{\sim}{S}^{(k)} + \underset{\sim}{F}\right)$
iv. Check the convergence test, $\left|U_i^{(k+1)} - U_i^{(k)}\right| \le \varepsilon = 10^{-10}$. If yes, go to step (v). Otherwise go back to step (iii).
v. Display approximate solutions.

NUMERICAL PERFORMANCE ANALYSIS

In order to investigate the performance analysis of the cubic non-polynomial spline approximation equation by using the two proposed iterative methods, there are three criteria can be used to compare with GS method. The following three criteria are the number of iterations, time of iterations and maximum absolute error. Numerical test for following equation

$$y'' - 4y = 4\cosh(1), \quad x \in [0,1] \tag{14}$$

which is the exact solution for problem (14) is given as $y(x) = \cosh(2x-1) - \cosh(1)$. Then, the results for the performance analysis have been tabulated in Table 1.

Table 1. Numerical results of the performance assessment

Number of Iterations					
m	**128**	**256**	**512**	**1024**	**2048**
GS	18173	66139	238353	848604	2975185
SOR	382	723	1438	4097	5367
Improvement (%)	97.90	99.18	99.40	99.52	99.82
Time of Iterations (Second)					
GS	16.4300	47.1800	169.3099	881.0900	3747.6400
SOR	0.8700	1.6500	2.400	5.6800	7.7200
Improvement (%)	94.70	96.50	98.58	99.36	99.79
Maximum Absolute Error					
GS	9.5665e-06	1.9487e-06	1.2847e-06	7.4088e-06	3.0203e-05
SOR	9.6849e-06	2.4225e-06	6.0855e-07	1.5644e-07	2.6459e-08
Improvement (%)	1.24	24.31	52.63	97.89	99.91

CONCLUSION

In this paper, the cubic non-polynomial spline approximation has been derived based on the cubic non-polynomial spline general function and its performance in term of number of iterations, time of iterations and maximum absolute error has been obtained by considering the two-point boundary value problems together with its exact solution and solved iteratively by using SOR and GS iterative methods. Then, the numerical performance analysis result has shown that SOR iterative method is superior compared to GS iterative method for different grid sizes (128, 256, 512, 1024, 2048) and can be seen through the improvement of its respective number of iteration (97.90%, 99.18%, 99.40%, 99.52%, 99.82%), execution time (94.70%, 96.50%, 98.58%, 99.36%, 99.79%) and maximum absolute error (1.24%, 24.31%, 52.63%, 97.89%, 99.91%). Thus, it can be concluded that the cubic non-polynomial spline approximation approach is best solved by using SOR iterative method compared to GS iterative method.

REFERENCES

[1] Albasiny, E. L & Hoskin, W. D. (1969). Cubic Spline Solution to Two-Point Boundary Value Problems. *The Computer Journal*, **12**(2), 151-153.

[2] Chen, B., Tong, L. & Gu, Y. (2006). Precise time integration for linear two-point boundary value problems. *Applied Mathematics and Computation*, **175**, 182–211.

[3] Evans, D. J. (1985). Group Explicit iterative methods for solving large linear systems. *International Journal of Computer Mathematics*, **17**, 81-108.

[4] Fang, Q., Tsuchiya, T. & Yamamoto, T. (2002). Finite difference, finite element and finite volume methods applied to two-point boundary value problems. *Journal of Computational and Applied Mathematics,* **139**, 9-19.

[5] Goffe, W. L. (1993). A user's guide to the numerical solution of two-point boundary value problems arising in continuous time dynamic economic models. *Computational Economics*, **6**(3-4), 249-255.

[6] Hackbusch, W. (1995). Iterative solution of large sparse systems of equations. New York: Springer-Verlag.

[7] Jang, B. (2008). Two-point boundary value problems by the extended adomian decomposition method. *Journal of Computational and Applied Mathematics*, **219**, 253–262.

[8] Mohsena, A. & Gamel, M. E. (2008). On the Galerkin and Collocation methods for two-point boundary value problems using sinc bases. *Computer and Mathematics with Applications*, **56**, 930-941.

[9] Ozisik, M. N. (1989). *Boundary value problems of heat conduction*. United States: Courier Corporation.

[10] Ramadan, M. A., Lashien, I. F. & Zahra, W. K. (2007). Polynomial and nonpolynomial spline approaches to the numerical solution of second order boundary value problems. *Applied Mathematics and Computation*, **184**, 476-484.

[11] Saad, Y. (1996). *Iterative method for sparse linear systems.* Boston: International Thomas Publishing.

[12] Ibrahim, A. & Abdullah, A. R. (1995). Solving the two-dimensional diffusion equation by the four point explicit decoupled group (EDG) iterative method. *International Journal of Computer Mathematics*, **58**, 253-256.

[13] Saudi, A. & Sulaiman, J. (2009). Path Planning for Mobile with Half-Sweep Successive Over-Relaxation (HSSOR) Iterative Method. *Proceedings Symposium on Progress in Information and*

Communication Technology (SPICT'09). 7-8 December, 2009, Kuala Lumpur, Malaysia. pp 57-62.

[14] Sung, N. H. (2001). A nonlinear shooting method for two point boundary value problems. *Computers and Mathematics with Applications*, **42**, 1411-1420.

[15] Young, D. M. (1954). Iterative methods for solving partial difference equations of elliptic Type. *Transactions of the American Mathematical Society*, **76**, 92-111.

[16] Young, D. M. (1971). Iterative solution of large linear systems. London: Academic Press.

[17] Young, D. M. (1972). Second-degree iterative methods for the solution of large linear systems. *Journal of Approximation Theory*, **5**, 37-148.

[18] Young, D. M. (1976). Iterative solution of linear systems arising from finite element techniques. In: Whiteman, J. R. (eds.). *The Mathematics of Finite Elements and Applications* (Vol. 2). London: Academic Press.

[19] Yousif, W. S. & Evans, D. J. (1995). Explicit De-coupled group iterative methods and their implementations. *Parallel Algorithms and Applications* (Vol. 7). New York: Springer-Verlag.

TRANSACTIONS ON SCIENCE AND TECHNOLOGY