Global Journal of Pure and Applied Mathematics.
ISSN 0973-1768 Volume 13, Number 11 (2017), pp. 7921-7934

http://www.ripublication.com

Cubic B-spline Solution of Two-point Boundary Value Problem Using HSKSOR Iteration

M. N. Suardi, N. Z. F. M. Radzuan and J. Sulaiman

Department of Mathematics with Economics, Faculty of Science and Natural Resources,

Universiti Malaysia Sabah, 88400 Kota Kinabalu, Sabah, Malaysia.

Abstract

The purpose of this paper deals with two-point boundary value problems (BVPs) which dicretized by using cubic B-spline discretization scheme to generate cubic B-spline approximation equation. Then, the approximate solution is used to construct a linear system, in which this linear system will be solved via using Half-sweep Kaudd Successive Over Relaxation (HSKSOR) iterative method. As comparative, Full-sweep Gauss-Seidel (FSGS) and Full-sweep Kaudd Successive Over Relexation (FSKSOR) iterative methods are considered and three numerical examples are taken to observe the performance of the proposed iterative methods. Throughout implementation of numerical experiments, three parameters are compared such as number of iteration, execution time and maximum error. According to the results, the HSKSOR method is a superior in term number of iteration and execution time compared with FSGS and FSKSOR.

Keywords: cubic B-spline approximation equation; HSKSOR iteration; two-point boundary value problem

Introduction

The two-point BVPs problems arise very frequently since it gives more advantages to solve many phenomena in science, physics and engineering field. Clearly, it is necessary and important to construct an efficient numerical methpd for solving the two-point BVPs. For this reason, many researchers give attention in solving the two-point BVPs [1-3]. Also, there are various methods have been used to these problem such as Sinc-Galerkin method and modifications decomposition [4], Adomain decomposition method [5] and hybrid Galerkin method [6]. Besides these methods, others methods have been used such as shooting method [7], the spline solution based on quadratic and cubic spline schemes [8,9] and B-spline method [10].

Consider the two-point boundary value problem at interval $[x_0, x_n]$ as

$$y'' + f(x)y' + g(x)y = r(x)\ , x \in [x_0, x_n] \tag{1}$$

subject to the boundary conditions

$$y(x_0) = a,\ \ y(x_n) = b \tag{2}$$

where a and b represented as left and right boundary respectively for two-point boundary value problem [8].

To solve problem (1), this paper aims to examine the application of HSKSOR iteration together with B-spline discretization scheme for solving the linear system which are generated from the discretization process. Before that, we need to know the basic concept of the B-spline function.

Actually in early 1960s, Pierre Bezier has upgraded the B-spline method which was proposed by P. De Calteljau [11]. He is a mathematician and engineer who fixed the disadvantage that exist in the early theories of B-spline. Basically, B-spline approach from the theories of spline methods was introduced by a mathematician, Schoenberg [12]. The general formula of B-spline curve is defined by [11,13]

$$y(x) = \sum_{i=0}^{n} P_i \cdot \beta_{i,d}(x), 0 \leq \mathrm{x} \leq 1 \tag{3}$$

where the control point is P_i and $\beta_{i,d}(x)$ represent as B-spline basis functions and the function $\beta_{i,d}(x)$ can be written in term of B-spline of degree d as [14]

$$\beta_{i,d}(x) = \frac{x-x_i}{x_{i+d-1}-x_i}\beta_{i,d-1}(x)\,\frac{x_{i+d}-x_i}{x_{i+d}-x_{i+2}}\beta_{i+2,d-1}(x) \tag{4}$$

with condition,

$$\beta_{i,0}(x) = \begin{cases} 1 & , x \in [x_{\mathrm{i}}, x_{\mathrm{i+1}}] \\ 0 & , otherwise \end{cases} \tag{5}$$

To discretize the two-point BVPs (1), cubic B-spline discretization scheme need to be imposed to get the approximation equation then it is used to construct a linear system. Since having the linear system, there are various iterative methods can be used to solve numerically the linear system which it has been discussed by Young [15], Hackbush [16] and Saad [17]. After that, the implementations of the iterative methods have been considered to identify which one the superior iterative methods. With attention to improve the convergence rate, the half-sweep concept has been applied. This concept was introduced by Abdullah [18] via the Explicit Decoupled Group (EDG) method to solve two-dimensional poisson equations. Due to advantages of the half-sweep iteration, there are many researchers have discussed about this concept in [19-23]. In this paper, the combination of half-sweep as technique and KSOR iterative methods or called as HSKSOR is considered to solve problem (1). The FSGS plays a important role as control method and FSKSOR was also implemented where it can be used as comparison purpose for HSKSOR iterative method.

The outline of this paper is as follows: In section 2, the formulation of cubic B-spline approximate solution has been derived. In section 3, the derivation of family of KSOR are demonstrated where the HSKSOR is introduced. Then, in section 4, some numerical problems are tested and illustrated the performance of the iterative methods are considered. The conclusion in section 5.

Cubic B-Spline Approximation Equations

As mentioned in section 1, the half-sweep concept is imposed to improve the convergence rate. Here the different value of h from the current point to next point between full- and half-sweep is shown in Figures 1. The Figure 1(a) shows the implementation of the full-sweep iteration which all interior nodes are considered one by one point which its distance for each point is 1h. However, for half-sweep iteration with its distance, 2h can seen in Figure 1(b).

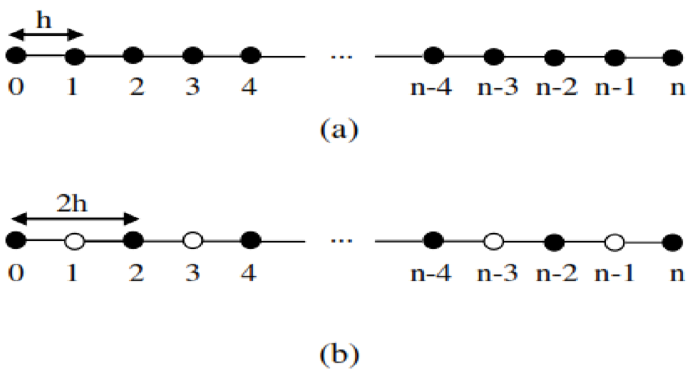

Figure 1. The distribution of uniform points for (a) full-sweep and (b) half-sweep iteration

Before constructing the linear system of problem (1), cubic B-spline approximation equation must be derived completely via cubic B-spline discretization scheme. Based on equation (4) and consider $d = 3$, the cubic B-spline function be given by [24]

$$\beta_{i,3}(x) = \frac{x-x_i}{x_{i+6}-x_i}\left[\begin{array}{l} \frac{x-x_i}{x_{i+4}-x_i}\left[\begin{array}{l}\frac{x-x_i}{x_{i+2}-x_i}\beta_{i,0}(x) \\ +\frac{x_{i+4}-x}{x_{i+4}-x_{i+2}}\beta_{i+2,0}(x)\end{array}\right] \\ +\frac{x_{i+6}-x}{x_{i+6}-x_{i+2}}\left[\begin{array}{l}\frac{x-x_{i+2}}{x_{i+4}-x_{i+2}}\beta_{i+2,0}(x) \\ +\frac{x_{i+6}-x}{x_{i+6}-x_{i+4}}\beta_{i+4,0}(x)\end{array}\right]\end{array}\right]$$

$$+\frac{x_{i+8}-x}{x_{i+8}-x_{i+2}}\left[\begin{array}{l} \frac{x-x_{i+2}}{x_{i+6}-x_{i+2}}\left[\begin{array}{l}\frac{x-x_{i+2}}{x_{i+2}-x_i}\beta_{i+2,0}(x) \\ +\frac{x_{i+6}-x}{x_{i+6}-x_{i+4}}\beta_{i+4,0}(x)\end{array}\right] \\ +\frac{x_{i+8}-x}{x_{i+8}-x_{i+4}}\left[\begin{array}{l}\frac{x-x_{i+4}}{x_{i+6}-x_{i+4}}\beta_{i+4,0}(x) \\ +\frac{x_{i+8}-x}{x_{i+8}-x_{i+6}}\beta_{i+6,0}(x)\end{array}\right]\end{array}\right] \tag{6}$$

Then, the cubic B-spline function in equation (6) is derived and simplified in order to get the piecewise function at several subintervals. The following is the definition of the cubic B-spline function at points, x_i, x_{i-2}, x_{i-4} and x_{i-6}:

$$\beta_{i,3}(x) = \frac{1}{6\mathrm{h}^3}\begin{cases} (x-x_i)^3, & x \in [x_{\mathrm{i}}, x_{\mathrm{i+2}}] \\ k_1, & x \in [x_{\mathrm{i+2}}, x_{\mathrm{i+4}}] \\ k_2, & x \in [x_{\mathrm{i+4}}, x_{\mathrm{i+6}}] \\ (x_{i+8}-x)^3, & x \in [x_{\mathrm{i+6}}, x_{\mathrm{i+8}}] \end{cases} \tag{7}$$

where,

$$k_1 = h^3 + 3h^2(x-x_{i+2}) + 3h(x-x_{i+2})^2 + 3(x-x_{i+2})^3,$$
$$k_2 = h^3 + 3h^2(x_{i+6}-x) + 3h(x_{i+6}-x)^2 + 3(x_{i+6}-x)^3.$$

$$\beta_{i-2,3}(x) = \frac{1}{6\mathrm{h}^3}\begin{cases} (x-x_{i-2})^3, & x \in [x_{\mathrm{i-2}}, x_{\mathrm{i}}] \\ k_3, & x \in [x_{\mathrm{i}}, x_{\mathrm{i+2}}] \\ k_4, & x \in [x_{\mathrm{i+2}}, x_{\mathrm{i+4}}] \\ (x_{i+6}-x)^3, & x \in [x_{\mathrm{i+4}}, x_{\mathrm{i+6}}] \end{cases} \tag{8}$$

where,

$$k_3 = h^3 + 3h^2(x-x_i) + 3h(x-x_i)^2 + 3(x-x_i)^3,$$
$$k_4 = h^3 + 3h^2(x_{i+4}-x) + 3h(x_{i+4}-x)^2 + 3(x_{i+4}-x)^3.$$

$$\beta_{i-4,3}(x) = \frac{1}{6\mathrm{h}^3}\begin{cases}(x - x_{i-4})^3, & x \in [x_{\mathrm{i}-4}, x_{\mathrm{i}-2}]\\ k_5, & x \in [x_{\mathrm{i}-2}, x_{\mathrm{i}}]\\ k_6, & x \in [x_{\mathrm{i}}, x_{\mathrm{i}+2}]\\ (x_{i+4} - x)^3, & x \in [x_{\mathrm{i}+2}, x_{\mathrm{i}+4}]\end{cases} \tag{9}$$

where,

$$k_5 = h^3 + 3h^2(x - x_{i-2}) + 3h(x - x_{i-2})^2 + 3(x - x_{i-2})^3,$$
$$k_6 = h^3 + 3h^2(x_{i+2} - x) + 3h(x_{i+2} - x)^2 + 3(x_{i+2} - x)^3.$$

$$\beta_{i-6,3}(x) = \frac{1}{6\mathrm{h}^3}\begin{cases}(x - x_{i-6})^3, & x \in [x_{\mathrm{i}-6}, x_{\mathrm{i}-4}]\\ k_7, & x \in [x_{\mathrm{i}-4}, x_{\mathrm{i}-2}]\\ k_8, & x \in [x_{\mathrm{i}-2}, x_{\mathrm{i}}]\\ (x_{i+2} - x)^3, & x \in [x_{\mathrm{i}}, x_{\mathrm{i}-2}]\end{cases} \tag{10}$$

where,

$$k_7 = h^3 + 3h^2(x - x_{i-4}) + 3h(x - x_{i-4})^2 + 3(x - x_{i-4})^3,$$
$$k_8 = h^3 + 3h^2(x_i - x) + 3h(x_i - x)^2 + 3(x_i - x)^3.$$

Consider equation (7) to (10) and substituting $x = x_i$ into the formulations, and then we have

$$\left.\begin{aligned}\beta_{i,3}(x_i) &= 0\\ \beta_{i-2,3}(x_i) &= \tfrac{1}{6}\\ \beta_{i-4,3}(x_i) &= \tfrac{4}{6}\\ \beta_{i-6,3}(x_i) &= \tfrac{1}{6}\end{aligned}\right\} \tag{11}$$

Next, by using the first differentiate concept to the formulation in equations (7) to (10), the first derivate is identified and substituting $x = x_i$ into it, we get as follow

$$\left.\begin{aligned}\beta'_{i,3}(x_i) &= 0\\ \beta'_{i-2,3}(x_i) &= \tfrac{1}{2\mathrm{h}}\\ \beta'_{i-4,3}(x_i) &= 0\\ \beta'_{i-6,3}(x_i) &= -\tfrac{1}{2\mathrm{h}}\end{aligned}\right\} \tag{12}$$

By doing with the same steps, to derive equation (12), the second derivative of the functions (7) to (10) can be shown as follows

$$\left.\begin{aligned} \beta''_{i,3}(x_i) &= 0 \\ \beta''_{i-2,3}(x_i) &= \frac{1}{\mathrm{h}^2} \\ \beta''_{i-4,3}(x_i) &= -\frac{2}{\mathrm{h}^2} \\ \beta''_{i-6,3}(x_i) &= \frac{1}{\mathrm{h}^2} \end{aligned}\right\} \tag{13}$$

With the attention of simplicity and by taking $n = 16$, the approximate solution in equation (3) can be expressed as

$$\begin{aligned} y(x) = {} & C_{-6} \cdot \beta_{-6,3}(x) + C_{-4} \cdot \beta_{-4,3}(x) + C_{-2} \cdot \beta_{-2,3}(x) + C_0 \cdot \beta_{0,3}(x) + C_2 \cdot \beta_{2,3}(x) \\ & + C_4 \cdot \beta_{4,3}(x) \\ & + C_6 \cdot \beta_{6,3}(x) + C_8 \cdot \beta_{8,3}(x) + C_{10} \cdot \beta_{10,3}(x) + C_{12} \cdot \beta_{12,3}(x) + C_{14} \cdot \beta_{14,3}(x) \end{aligned} \tag{14}$$

where C_i, $i = -6, -4, -2, \ldots, n-2$ are unknown coefficients. Now, the function in equations (11) to (13) will substitute into the proposed problem (1) in order to derive the cubic B-spline approximation equation of problem (1). For the simplicity, the cubic B-spline approximation equation can be written as

$$\alpha_i \cdot C_{i-6} + \beta_i \cdot C_{i-4} + \gamma_i \cdot C_{i-2} = r_i \tag{15}$$

where

$$\alpha_i = \frac{1}{h^2} - \frac{p_i}{2h} + \frac{q_i}{6},$$

$$\beta_i = -\frac{2}{h^2} + \frac{4q_i}{6},$$

$$\gamma_i = \frac{1}{h^2} + \frac{p_i}{2h} + \frac{q_i}{6}.$$

for $i = 0,2,4, \ldots, n-2$. Furthermore, the approximation equation (15) can be used to construct the tridiagonal linear system which is shown as

$$A\underline{C} = \underline{R} \tag{16}$$

where

$$A=\begin{bmatrix}\alpha_0 & \beta_0 & \gamma_0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & \alpha_2 & \beta_2 & \gamma_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & \alpha_4 & \beta_4 & \gamma_4 & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & \alpha_6 & \beta_6 & \gamma_6 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & \alpha_8 & \beta_8 & \gamma_8 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & \alpha_{10} & \beta_{10} & \gamma_{10} & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 0 & \alpha_{12} & \beta_{12} & \gamma_{12} & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \alpha_{14} & \beta_{14} & \gamma_{14} & 0\\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \alpha_{16} & \beta_{16} & \gamma_{16}\end{bmatrix},$$

$$\underline{C}=[c_{-4} \quad c_{-2} \quad c_0 \quad c_2 \quad c_4 \quad c_6 \quad c_8 \quad c_{10} \quad c_{12}]^T,$$

$$\underline{R}=[r_0-\alpha \quad r_2 \quad r_4 \quad r_6 \quad r_8 \quad r_{10} \quad r_{12} \quad r_{14} \quad r_{16}-\beta]^T.$$

Clearly, A represents as the coefficient matrix, $\underline{C}$ is an unknown vector and $\underline{R}$ is a known vector. By considering the sufficient condition for solving any linear system [15], the coefficients matrix, A in linear system (16) must be the positive definite, $[a_{ii}] \geq \sum_{i\neq j}[a_{ij}]$ to obtain the approximate solution.

Family Of Kaudd Successive Over Relaxation Iterative Methods

According to linear system (16), it can be identified that the coefficient matrix A is large and sparse. It means that the iterative methods are suitable option as linear solver to solve the linear system [15-17]. Therefore, we will consider the family of KSOR iterative methods. In 2012, Youssef [25] introduced the KSOR iterative method which is the improvement method of SOR iterative method. The idea of KSOR is updating process by using current component instead recent calculated value. This paper proposed the HSKSOR iterative method which is the combination of half-sweep iteration and KSOR iterative method. In fact, HSKSOR is the improvement of KSOR iterative method. Let the coefficient matrix A be expressed as the summation of three matrices

$$A=L+D+U \tag{17}$$

where D represents as a diagonal matrix of matrix A and L and U are strictly lower matrix and strictly upper matrix respectively. The linear system can defined by substituting equation (17) into equation (16) as follows

$$(L + D + U)\underline{C} = \underline{R} \tag{18}$$

Referring to equation (18), the HSKSOR can be written in matrix and iterative forms. The HSKSOR iteration scheme in iterative form can expressed as

$$\underline{C}^{(\mathrm{k}+1)} = [(1-\omega)\mathrm{D} - \omega\mathrm{L}]^{-1}(\mathrm{D} + \mathrm{U})\underline{C}^{(\mathrm{k})} + [(1-\omega)\mathrm{D} - \omega\mathrm{L}]^{-1}\underline{R} \tag{19}$$

or the general formula of HSKSOR in matrix form as follow

$$\underline{C_i}^{(k+1)} = \underline{C_i}^{(k)} + \frac{\omega}{A_{ii}}\left(\underline{R_i} - \sum_{j=1}^{i-1} A_{ij}\,\underline{C_j}^{(k+1)} - \sum_{j=i+1}^{n} A_{ij}\,\underline{C_j}^{(k)} - A_{ij}\underline{C_j}^{(k+1)}\right) \tag{20}$$

where $i = 0,2,4,8, \ldots, n-2$. The implementation of HSKSOR iterative method can be seen in Algorithm 1.

ALGORITHM 1. HSKSOR iterative method

---

i. Set initial value $\mathrm{c}^{(0)}$=0.
ii. Calculate the coefficient matrix, A and vector, $\underline{R}$.
iii. For $i = 2,4,6, \ldots, n-2$, calculate
$$\underline{C}^{(\mathrm{k}+1)} = [(1-\omega)\mathrm{D} - \omega\mathrm{L}]^{-1}(\mathrm{D} + \mathrm{U})\underline{C}^{(\mathrm{k})} + [(1-\omega)\mathrm{D} - \omega\mathrm{L}]^{-1}\underline{R}$$
iv. Check the convergence test, $\left|\underline{C_i}^{(k+1)} - \underline{C_i}^{(k)}\right| < \varepsilon = 10^{-10}$. If yes, go to step (vi). Otherwise go back to step (iii).
v. Display numerical solution

Numerical Performance Analysis

Three examples of two-point boundary value problem have been taken to demonstrate the performance of the HSKSOR iterative method. With attention of comparision, the HSKSOR is compared with FSGS and FSKSOR in which the FSGS plays a role as control method for the others iterative methods. There are three parameter that are considered to investigate the performance of the proposed iterative methods such as number of iteration (Iter), execution time in second (Time) and maximum error (Error). In addition to that, the value of tolerance error is constant at different grid sizes as $\varepsilon = 10^{-10}$.

i. Example 1 [26]

We consider the following two-point boundary value problem as

$$y'' - y' = -e^{(x-1)-1},\ x \in [0,1] \tag{21}$$

The exact solution given for problem (21) is

$$y = x\left(1 - e^{(x-1)}\right), x \in [0,1]$$

ii. Example 2 [3]

Let two-point boundary value problem be considered as follows

$$-y''-2y' + 2y = e^{-2x},\ x \in [0,1] \tag{22}$$

with the exact solution for problem (22) given as

$$y(x) = \frac{1}{2}e^{-\left(1+\sqrt{3}\right)} + \frac{1}{2}e^{-2x}, x \in [0,1]$$

iii. Example 3 [7]

Let two-point boundary value problem be considered as follows

$$y'' - 4y = \cosh(1),\ x \in [0,1] \tag{23}$$

where the exact solution given as

$$y(x) = cosh(2x - 1) - cosh(1), x \in [0,1]$$

Results and Discussion

Throughout the numerical experiments for equations (21) to (23), there several grid sizes are used such as 1024, 2048, 4096, 8192 and 16384. The numerical results for FSGS, FSKSOR and HSKSOR can be seen in Table 1, 2 and 3. However, Table 4 shows the reduction percentage for FSKSOR and HSKSOR where FSGS acts as control method.

TABLE 1. Comparison of the number of iterations, execution time (seconds) and the maximum absolute error on iterative methods for example 1.

M	Method	Iter	Time(second)	Error
1024	FSGS	1025490	109.60	1.03e-05
	FSKSOR	2853	0.64	3.78e-08
	HSKSOR	1526	0.36	1.16e-07
2048	FSGS	3527433	501.08	4.14e-05
	FSKSOR	5792	1.22	1.45e-08
	HSKSOR	2853	0.64	3.78e-08
4096	FSGS	11811520	3214.48	1.66e-04
	FSKSOR	10221	2.79	9.89e-08
	HSKSOR	5792	1.21	1.45e-08
8192	FSGS	38052999	15288.39	6.63e-04
	FSKSOR	19313	8.00	1.50e-07
	HSKSOR	10221	3.07	9.89e-08
16384	FSGS	115439220	58347.49	2.65e-03
	FSKSOR	35897	27.87	3.14e-07
	HSKSOR	19313	9.09	1.50e-07

TABLE 2. Comparison of the number of iterations, execution time (seconds) and the maximum absolute error on iterative methods for example 2.

M	Method	Iter	Time(second)	Error
1024	FSGS	886861	100.44	8.24e-06
	FSKSOR	3823	0.72	1.06e-07
	HSKSOR	1933	0.40	1.01e-06
2048	FSGS	3095807	482.89	3.26e-05
	FSKSOR	7553	1.60	4.96e-08
	HSKSOR	3823	0.70	2.62e-07
4096	FSGS	10576347	3099.27	1.30e-04
	FSKSOR	14909	4.54	6.19e-08
	HSKSOR	7553	1.63	8.76e-08
8192	FSGS	35077202	16192.67	5.21e-04
	FSKSOR	29375	15.24	1.24e-07
	HSKSOR	14907	4.75	7.23e-08
16384	FSGS	111394765	56824.70	2.09e-03
	FSKSOR	57881	52.86	2.48e-07
	HSKSOR	29375	14.85	1.26e-07

TABLE 3. Comparison of the number of iterations, execution time (seconds) and the maximum absolute error on iterative methods for example 3.

M	Method	Iter	Time(second)	Error
1024	FSGS	848604	96.58	7.44e-06
	FSKSOR	2613	0.51	1.35e-07
	HSKSOR	1353	0.32	4.90e-07
2048	FSGS	2975185	466.09	3.02e-05
	FSKSOR	4932	1.10	6.92e-08
	HSKSOR	2613	0.61	1.35e-07
4096	FSGS	10223821	2999.24	1.21e-04
	FSKSOR	9209	2.61	7.46e-08
	HSKSOR	4932	1.24	6.92e-08
8192	FSGS	34187618	15930.80	4.84e-04
	FSKSOR	17449	9.25	1.96e-07
	HSKSOR	9209	2.94	7.46e-08
16384	FSGS	109919813	57115.58	1.94e-03
	FSKSOR	36752	34.11	3.74e-07
	HSKSOR	17449	8.95	1.96e-07

TABLE 4. Reduction percentage of the number of iterations and computational time for the FSKSOR and HSKSOR compared with FSGS iterative method.

		FSKSOR	HSKSOR
Problem 1	Iter	99.72 – 99.97%	99.85 – 99.98%
	Time	99.42 – 99.95%	99.67 – 99.99%
Problem 2	Iter	99.57 – 99.95%	99.78 – 99.97%
	Time	99.28 – 99.91%	99.60 – 99.97%
Problem 3	Iter	99.69 – 99.97%	99.84 – 99.98%
	Time	99.47 – 99.94%	99.67 – 99.98%

The numerical results in Table 1, 2 and 3 have been recorded by imposing the FSGS, FSKSOR and HSKSOR iterative methods. From the results, the HSKSOR iterative method requires less of number of iterations as compared with FSGS and FSKSOR. In term of execution time, the HSKSOR iterative method is the fastest method than the others proposed methods. Table 4 shows that the percentage of HSKSOR is higher than the percentage of FSKSOR iterative method. Clearly, the HSKSOR iterative method is the efficient method as compared with FSGS and FSKSOR iterative methods.

Conclusion

The paper presents the application of the cubic B-spline approach with the HSKSOR iteration to solve two-point boundary value problem. In order to get cubic B-spline approximation equation, cubic B-spline discretization scheme is applied into the proposed problem and it leads to construct the linear system. The families of KSOR iterative methods are considered to solve the linear system. According to the numerical results are recorded, it can be pointed out that the HSKSOR needs less number of iteration and execution time to solve two-point boundary value problem. As conclusion, the HSKSOR is superior method as compared with FSGS and FSKSOR iterative methods. For further works, the quarter-sweep [27,28] iteration concept should be consider as based on B-spline approach to solve two-point boundary value problems.

Acknowledgement

The research is fully supported by UMSGreat, GUG0135-1/2017. The authors fully acknowledged Ministry of Higher Education (MOHE) and Universiti Malaysia Sabah for the approved fund which makes this important research viable and effective.

Reference

[1] Aarao, J., Bradshaw-Hajek, B. H., Miklavcic, S. J. and Ward, D. A., 2010, “The extended domain eigen-function method for solving elliptic boundary value problems with annular domains”, Journal of Physics A: Mathematical and Theoretical, 43, pp. 185-202.

[2] Wang, Y. M. and Guo, B. Y., 2008, “Fourth-order compact finite difference method for fourth-order nonlinear elliptic boundary value problems”, Journal of Computational and Applied Mathematics, 221(1), pp. 76-97.

[3] Robertson, T. N., 1971, “The Linear Two-Point Boundary Value Problem On An Infinite Interval”, Mathematics Of Computation, 25(115), pp. 475-481.

[4] El-Gamel, M., 2007, “Comparision Of The Solution Obtained By Adomian Decomposition And Wavelet-Galerkin Methods Of Boundary-Value Problems”, Applied Mathematics and Computation, 186(1), pp. 652-664.

[5] Jang, B., 2007, “Two-Point Boundary Value Problems By Extended Adomian Decomposition Method”, Computational and Applied Mathematics, 219(1), pp. 253-262.

[6] Mohsen, A. and 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**,** pp. 930-941.

[7] Lin, Y., Enszer, J. A. and Stadtherr, M. A., 2008, “Enclosing All Solutions Of Two-Point Boundary Value Problems For ODEs”, Computer and Chemical Engineering, 32(8), pp. 1714-1725.

[8] Albasiny, E.L. and Hoskin, W.D., 1969. “Cubic Spline Solution to Two Point Boundary Value Problems”, The Computer Journal, 12(2), pp. 151-153.

[9] Ramadan, M. A., Lashien, I. F. and Zahra, W. K., 2007, “Polynomial And Nonpolynomial Spline Approaches To The Numerical Solution Of Second Order Boundary Value Problems”, Applied Mathematics and Computation, 184, pp. 476-484.

[10] Nazan, C. and Hikmet, C., 2008, “B-Spline Methods For Solving Linear System Of Second Order Boundary Value Problems”, Computers and Mathematics with Application, 57(5), pp. 757-762.

[11] Choi, J. W., Curry, R. E. and Elkaim, G. H., 2012, “Minimizing The Maximum Curvature Of Quadratic Bezier Curve With A tetragonal Concave Polygonal Boundary Constraint”, Computer Aided Design, 44, pp. 311-319.

[12] Schoenberg, I .J., 1946, “Contributions to the problem of approximation of equidistant data by analytic functions”, Quart. Appl. Math., 4, pp. 112–141.

[13] Suardi, M. N., Radzuan, N. Z. F. M. and Sulaiman, J., 2017, “Cubic B-spline Solution for Two-point Boundary Value Problem with AOR Iterative Method”, Journal of Phiysics: Conference Series, 890, 012015.

[14] Botella, I. and Shariff, K. 2003, “B-Spline Method In Fluid Dynamics”, International Journal of Computational Fluid Dynamics, 17(12), pp.133-149.

[15] Young, D. M., 1971, Iterative Solution Of Large Linear Systems. London: Academic Press.

[16] Hackbusch, W., 1995, Iterative Solution of Large Sparse Systems of Equations, Springer-Verlag.

[17] Saad, Y., 1996. Iterative Methods for Sparse Linear Systems, International Thomas Publishing.

[18] Abdullah, A. R., 1991, "The Four Point Explicit Decoupled Group (EDG) Method: A Fast Poisson Solver", Int. J. Comput. Math, 38, pp. 61-70.

[19] Alibubin, M. U., Sunarto, A., Akhir, M. K. M. and Sulaiman, J., 2016, "Performance Analysis of Half-sweep SOR iteration with Rotated Nonlocal Arithmetic Mean Scheme for 2D Nonlinear Elliptic Problem", Global Journal of Pure and Applied Mathematics, 12(4), pp. 3415-3424.

[20] Chew, J. V. L. and Sulaiman, J., 2016 "Half-sweep Newton-Gauss-Seidel for Implicit finite Difference Solution of 1D nonlinear Porous Medium Equations", Global Journal of Pure and Applied Mathematics, 12(3), pp. 2745-2752.

[21] Dahalan, A. A., Sulaiman, J. and Muthuvalu M. S., 2014 "Performance of HSAGE Method with Seikkala Derivative for 2-D Fuzzy Poisson Equation", Applied Mathematics Sciences, 18(8), pp. 885-899.

[22] Muthuvalu, M. S. and Sulaiman, J., 2011, "Half-Sweep Arithmetic Mean Method With Composite Trapezoidal Scheme For Solving Linear Fredholm Integral Equations", Appl. Math. Comput., 217(12), pp. 5442-5448.

[23] Hong, E. J., Saudi, A. and Sulaiman, J., 2017, "HSAOR Iteration for Poisson Image Blending Problem via Rotated Five-point Laplacian Operater", Global Journal of pure and applied mathematics, 13(9), pp. 5447-5459.

[24] Chang, J., Yang, Q. and Zhao L., 2011, "Comparison of B-spline Method and Finite Difference Method to Solve BVP Of Linear ODEs", Journal Of Computers, 6(10), pp. 2149-2155.

[25] Youssef, I. K., 2012, "On The Successive Over relaxation Method", J. Math. Stat., 8, pp. 176-184.

[26] Caglar, H. N., Caglar, S. H. and Elfaituri, K., 2006, "B-spline Interpolation Compared With Finite Difference, Finite Element And Finite Volume Methods Which Applied To Two Point Boundary Value Problems", Applied Mathematics and Computation, **175**(1), pp. 72-79.

[27] Muthuvalu, M. S. and Sulaiman, J., 2011, "Half-Sweep Arithmetic Mean method with composite trapezoidal scheme for solving linear Fredholm integral equations", Appl. Math. Comput., 217(12), pp. 5442-5448.

[28] Sulaiman, J., Othman, M. and Hasan, M. K., 2004, "Quarter-sweep Iterative Alternating Decomposition Explicit algorithm applied to diffusion equation", Int. J. Comput. Math., 18(12), pp. 1559-1565.