Quartic non-polynomial spline solution for solving two-point boundary value problems by using Conjugate Gradient Iterative Method

Article *in* Journal of Applied Mathematics and Computational Mechanics · March 2017

DOI: 10.17512/jamcm.2017.1.04

CITATIONS
4

READS
59

3 authors:

Hynichearry Justine
North Borneo University College, Kota Kinabalu, Sabah, Malaysia
2 PUBLICATIONS **6** CITATIONS
SEE PROFILE

Jackel Chew Vui Lung
Universiti Malaysia Sabah (UMS)
10 PUBLICATIONS **13** CITATIONS
SEE PROFILE

Jumat Sulaiman
Universiti Malaysia Sabah (UMS)
317 PUBLICATIONS **1,122** CITATIONS
SEE PROFILE

JAM CM Journal of Applied Mathematics and Computational Mechanics 2017, 16(1), 41-50
www.amcm.pcz.pl p-ISSN 2299-9965
DOI: 10.17512/jamcm.2017.1.04 e-ISSN 2353-0588

QUARTIC NON-POLYNOMIAL SPLINE SOLUTION FOR SOLVING TWO-POINT BOUNDARY VALUE PROBLEMS BY USING CONJUGATE GRADIENT ITERATIVE METHOD

Hynichearry Justine, Jackel Vui Ling Chew, Jumat Sulaiman

Mathematics with Economics Programme, University Malaysia Sabah
Sabah, Malaysia
cherryjust20@gmail.com, jackelchew93@gmail.com, jumat@ums.edu.my

Received: 23 November 2016; accepted: 10 February 2017

Abstract. Solving two-point boundary value problems has become a scope of interest among many researchers due to its significant contributions in the field of science, engineering, and economics which is evidently apparent in many previous literary publications. This present paper aims to discretize the two-point boundary value problems by using a quartic non-polynomial spline before finally solving them iteratively with Conjugate Gradient (CG) method. Then, the performances of the proposed approach in terms of iteration number, execution time and maximum absolute error are compared with Gauss-Seidel (GS) and Successive Over-Relaxation (SOR) iterative methods. Based on the performances analysis, the two-point boundary value problems are found to have the most favorable results when solved using CG compared to GS and SOR methods.

MSC 2010: *34B05*
Keywords: *two-point boundary value problems, quartic non-polynomial spline, Conjugate Gradient, Successive Over-Relaxation, Gauss-Seidel*

1. Introduction

Numerical methods have numerous significances in the field of sciences, economics, and engineering, and one of them when it comes to the solution of two-point boundary value problems which involve finding an approximate solution iteratively, as it will be time-consuming to solve them with analytical method. Some of the contributions of numerical methods related to the two-point boundary value problems include the modelling of chemical reactions and the modelling of heat transfer, such as in rocket thrust chamber liners and in the fuel elements for nuclear reactors as discussed by Ozisik [1]. On the other hand, Goffe [2] mentioned the modelling of the growth theory, capital theory, investment theory, resource economics and labor economics in the field of economics. Prior to these, many researchers had attempted to initiate different methods in order to accelerate the approximate solution when solving the problems and this can be abundantly

found in previous literary publications. Some of the methods that were readily apparent are the Newton-EGMSOR method [3], the EADM method [4], the shooting method [5], the PTI method [6], the nonlinear shooting method [7], the mean weight method [8], the finite difference, the finite element and the finite volume method [9] and the spline method [10, 11]. Despite all these methods, the solution in this paper was given focus based on the discretization of the quartic non-polynomial spline together with the Conjugate Gradient (CG) iterative method.

Moreover, there are many other iterative methods which are thoroughly discussed by Kelly [12], Burgerscentrum [13], Hestenes and Stiefel [14], Saad [15], Hackbush [16] and Young [17, 18]. According to Ibrahim and Abdullah [19], and Yousif and Evans [20], there are several numbers of the iterative methods family with a different concept, and they emphasized the concept of block iteration. In addition to that, Ul-Islam et al. [21], Ramadan et al. [22], Siddiqi et al. [23] have provided the basis for this paper at a different degree of splines to solve the two-point boundary value problems. In regards to the advantages of the CG iterative method and the spline approach as highlighted in [14] and [21-23] respectively, this present paper aims to develop a solution for the problems by using a quartic non-polynomial spline together with the CG iterative method. As for comparison purposes, Successive Over-Relaxation (SOR) and Gauss-Seidel (GS) were set as control methods so that the performances of the CG method can be determined in respect to its iteration number, execution time and maximum absolute error.

2. Two-point boundary value problems

Generally, the two-point boundary value problems can be expressed as Eq. (1) and subject to boundary conditions (2) as follows:

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

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

where $f(x)$, $q(x)$ and $g(x)$ are known functions restricted by boundary $[a,b]$ and A_i, $i=1,2$ is a constant. The solution for problem (1) cannot be obtained through a random selection of functions $f(x)$, $q(x)$ and $g(x)$ due to the restrictions held by the boundary conditions (2). Furthermore, the process for discretization of problem (1) is made simpler by assuming positive integer $m=2^p$, $p\geq 2$ and letting the solution domain, $[a,b]$ be divided uniformly into a uniform separation of nodes set or subinterval, m, as shown in Figure 1. Then, the length of the uniform subintervals, Δx can be defined as:

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

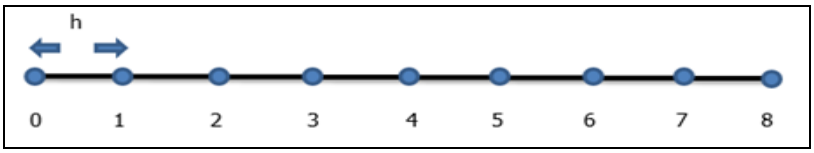

Fig. 1. Distribution of node point for domain solution $m = 8$

Moreover, the solution domain in Figure 1 was used to develop a uniform grid of a/the network as shown in Figure 2 for the derivation of the spline function, and the grid points in the solution domain were labeled as $x_i = a + ih,\ i = 0, 1, 2, \ldots, m$ with function $y(x)$ denoted as $y(x_i) = y_i$. The formulation and implementation of the GS, SOR and CG iterative method were then conducted based on the interior grid points until the convergence test is satisfied.

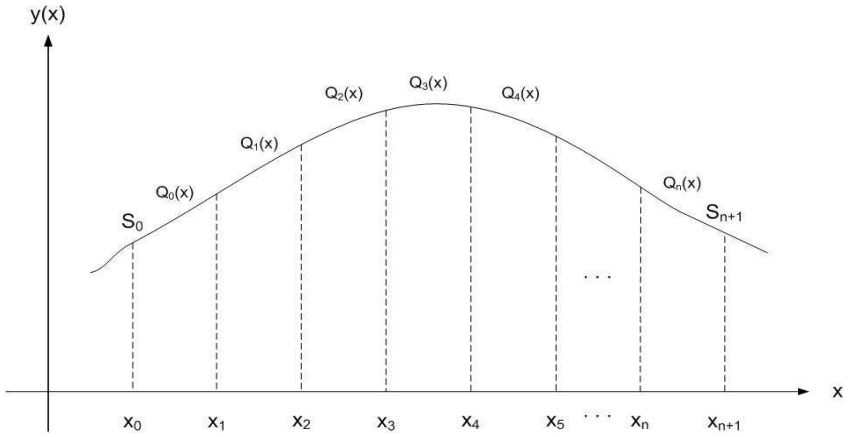

Fig. 2. Illustration of non-polynomial spline function for domain solution $m = 8$

3. Quartic non-polynomial spline approximation equation

The general form of the non-polynomial spline can be expressed as follows:

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

and it was used to discretize the problem (1) so that the approximation equation can be constructed as a system of linear equations in a matrix form. This discretization process was conducted by assuming $y(x)$ as the exact solution for problem (1) and S_i as the quartic non-polynomial spline approximation to $y_i = y(x_i)$ obtained from the mixed splines $Q_i(x)$ as shown in Figure 2 which passing through the points (x_i, S_i) and (x_{i+1}, S_{i+1}). Based on Eq. (4), the quartic non-polynomial spline can be expressed in $Q_i(x)$ as:

$$Q_i(x) = a_i \cos k(x - x_i) + b_i \sin k(x - x_i) + c_i (x - x_i)^2 + d_i (x - x_i) + e_i, \quad (5)$$

where a_i, b_i, c_i, d_i and e_i are constants for $i = 0, 1, 2, \ldots, n$ and k is a free parameter [19]. The function $Q_i(x)$ interpolates $y(x)$ at the points x_i by depending on k and reducing to quartic spline in $[a, b]$ as $k \to 0$.

Then, in order to obtain the necessary conditions for all the constants a_i, b_i, c_i, d_i and, e_i, the function $Q_i(x)$ has been satisfied at x_i and x_{i+1}, at boundary conditions (2) and at the continuity of the common nodes at (x_i, y_i) of first, second and third derivatives. Before deriving the expression for all the coefficients of (6) in terms of $F_i, F_{i+1}, D_i, D_{i+1}, S_i$ and S_{i+1}, we first define the function $Q_i(x)$ at second and fourth derivatives as:

$$\begin{aligned} &Q_i(x_i) = S_i, \quad Q_i(x_{i+1}) = S_{i+1}, \quad Q_i''(x_i) = D_i, \quad Q_i''(x_{i+1}) = D_{i+1}, \\ &Q_i^{(iv)}(x_i) = \frac{1}{2}\left(F_i + F_{i+1}\right) \end{aligned} \quad (6)$$

After performing a straightforward calculation, all the values (7) of constant a_i, b_i, c_i, d_i and, e_i were obtained as follows:

$$\begin{aligned} a_i &= \frac{1}{2k^4} F_i + \frac{1}{2k^4} F_{i+1} - \frac{1}{2k^2} D_i + \frac{1}{2k^2} D_{i+1}, \\ b_i &= \frac{1 - \cos\theta}{2k^4 \sin\theta} F_i + \frac{1 - \cos\theta}{2k^4 \sin\theta} F_{i+1} + \frac{1 + \cos\theta}{2k^2 \sin\theta} D_i - \frac{1 - \cos\theta}{2k^2 \sin\theta} D_{i+1}, \\ c_i &= \frac{1}{4k^2} F_i + \frac{1}{4k^2} F_{i+1} + \frac{1}{4} D_i + \frac{1}{4} D_{i+1}, \\ d_i &= \frac{1}{h} S_{i+1} - \frac{1}{h} S_i - \frac{h}{4k^2} F_i - \frac{h}{4k^2} F_{i+1} - \frac{4 - h^2k^2}{4hk^2} D_i + \frac{4 - h^2k^2}{4hk^2} D_{i+1}, \\ e_i &= S_i - \frac{1}{2k^4} F_i - \frac{1}{2k^4} F_{i+1} + \frac{1}{2k^2} D_i - \frac{1}{2k^2} D_{i+1}, \end{aligned} \quad (7)$$

where $\theta = kh$ and $i = 0,1,2,..,N$.

Now that all the values of constant a_i, b_i, c_i, d_i and, e_i were obtained, we then use the continuity conditions (2) of the quartic spline S_i at its first and third derivatives at the point (x_i, S_i) where the two quartics $Q_{i-1}^m(x)$ and $Q_i^m(x)$ join, and this relation can be written as:

$$Q_{i-1}^m(x) = Q_i^m(x),$$

where the degree of the derivative is $m = 0,1$.

Based on Eqs. (5) and (7), the relation at the first derivative (8) and the third derivative (9) can be expressed in following form:

$$
\begin{aligned}
&F_{i\text{-}1} + 2F_i + F_{i+1} \qquad (8)\\
&= \frac{4hk^2 + 4hk^2\cos\theta - 8\mathrm{k}\sin\theta - 2\mathrm{h}^2k^3\sin\theta}{-2\mathrm{h} + 2\mathrm{h}\cos\theta + h^2k\sin\theta} D_i - \frac{2hk^2 - 2hk^2\cos\theta + 4\mathrm{k}\sin\theta - \mathrm{h}^2\mathrm{k}^3\sin\theta}{-2\mathrm{h} + 2\mathrm{h}\cos\theta + h^2k\sin\theta} D_{i+1}\\
&- \frac{2\mathrm{hk}^2 - 2\mathrm{hk}^2\cos\theta + 4k\sin\theta - \mathrm{h}^2\mathrm{k}^3\sin\theta}{-2\mathrm{h} + 2\mathrm{h}\cos\theta + h^2k\sin\theta} D_{i\text{-}1} + \frac{4\mathrm{k}^3\sin\theta}{-2\mathrm{h} + 2\mathrm{h}\cos\theta + \mathrm{h}^2k\sin\theta}\mathrm{S}_{i+1}\\
&- \frac{8\mathrm{k}^3\sin\theta}{-2\mathrm{h} + 2\mathrm{h}\cos\theta + \mathrm{h}^2k\sin\theta}\mathrm{S}_i + \frac{4\mathrm{k}^3\sin\theta}{-2\mathrm{h} + 2\mathrm{h}\cos\theta + \mathrm{h}^2k\sin\theta}\mathrm{S}_{i\text{-}1}
\end{aligned}
$$

$$
\begin{aligned}
&F_{i\text{-}1} + 2F_i + F_{i+1}\\
&= \frac{-2\mathrm{k}^2(1+\cos\theta)}{1-\cos\theta} D_i + \frac{\mathrm{k}^2(1+\cos\theta)}{1-\cos\theta} D_{i+1} + \frac{\mathrm{k}^2(1+\cos\theta)}{1-\cos\theta} D_{i\text{-}1} \qquad (9)
\end{aligned}
$$

Upon subtraction of Eqs. (8) and (9), it yields the following equation:

$$
\begin{aligned}
&= \frac{4hk^2 + 4hk^2\cos\theta - 8\mathrm{k}\sin\theta - 2\mathrm{h}^2k^3\sin\theta}{-2\mathrm{h} + 2\mathrm{h}\cos\theta + h^2k\sin\theta} D_i - \frac{2hk^2 - 2hk^2\cos\theta + 4\mathrm{k}\sin\theta - \mathrm{h}^2\mathrm{k}^3\sin\theta}{-2\mathrm{h} + 2\mathrm{h}\cos\theta + h^2k\sin\theta} D_{i+1}\\
&- \frac{2\mathrm{hk}^2 - 2\mathrm{hk}^2\cos\theta + 4k\sin\theta - \mathrm{h}^2\mathrm{k}^3\sin\theta}{-2\mathrm{h} + 2\mathrm{h}\cos\theta + h^2k\sin\theta} D_{i\text{-}1} + \frac{4\mathrm{k}^3\sin\theta}{-2\mathrm{h} + 2\mathrm{h}\cos\theta + \mathrm{h}^2k\sin\theta}\mathrm{S}_{i+1} \qquad (10)\\
&- \frac{8\mathrm{k}^3\sin\theta}{-2\mathrm{h} + 2\mathrm{h}\cos\theta + \mathrm{h}^2k\sin\theta}\mathrm{S}_i + \frac{4\mathrm{k}^3\sin\theta}{-2\mathrm{h} + 2\mathrm{h}\cos\theta + \mathrm{h}^2k\sin\theta}\mathrm{S}_{i\text{-}1}
\end{aligned}
$$

Then, a system of linear equations is constructed based on (10) in the following form:

$$\underset{\sim}{A}\underset{\sim}{y} = \underset{\sim}{F} \qquad (11)$$

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

$\underset{\sim}{y} = [y_1 \;\; y_2 \;\; y_3 \;\; \ldots \;\; y_{n-1} \;\; y_n]^T$ and $\underset{\sim}{F} = [F_1 - a_1y_0 \;\; F_2 \;\; F_3 \;\; \ldots \;\; F_{n-1} \;\; F_n - c_ny_{n+1}]^T$.

4. Algorithm of CG method

The CG iterative method was first discussed by Hestenes and Stiefel [14] as mentioned earlier, and this iterative method will converge in less than or equal to the size of the matrix itself in the absence of round-off error, with the assumption that matrix A in (11) is symmetric and positive definite. In fact, this method surpassed the Gauss elimination method. In addition to that, the CG method is much simpler to code when it comes to computer programming and requires less storage space due to its ability to maintain the particular matrix throughout the implementation and improvement which occur at each step of the estimations. In other words, the original data can be used to its maximum. Owing to these advantages of the CG method, the present paper aims to examine its performances in comparison with another two iterative methods which are Successive Over Relaxation (SOR) and Gauss-Seidel (GS) when solving the two-point boundary value problems using the quartic non-polynomial scheme.

By referring to (11), the CG method can be formulated by computing the sequence of n vectors $p_0, p_1, \ldots, p_{n-1}$ which are elements of R^n that satisfy the following conditions

$$p_i^T A p_j = 0, \quad i \neq j \tag{12}$$

and at the same time matrix A is assumed as $n \times n$ symmetrical matrix. As for method GS and SOR, the formulation begins by decomposing the matrix A as

$$A = D + L + U \tag{13}$$

where D is a diagonal matrix, L is a lower triangular matrix and U is upper triangular matrix. Upon imposing (13) onto (11), the formulation of GS and SOR methods can be written as (14) and (15) respectively.

$$\underset{\sim}{y}^{(k+1)} = (1-\omega)\,\underset{\sim}{y}^{(k)} + \omega (D+L)^{-1}\left(-U\underset{\sim}{y}^{(k)} + \underset{\sim}{F}\right) \tag{14}$$

$$\underset{\sim}{y}^{(k+1)} = -(D+L)^{-1}U\underset{\sim}{y} + (L+D)^{-1}F \tag{15}$$

To facilitate the convergence rate of the SOR method, the value of the parameter ω must be obtained first through several computer programs in the range of $1 \leq \omega \leq 2$. The optimal value of the parameter ω is selected based on the smallest iterations number. As for the GS method, the value of the parameter is equal to one if we reduce (14) to the GS method. Since both the GS and SOR methods are implemented for comparison purpose only, therefore only the algorithm for the CG method is presented.

The Algorithm of CG Method

i. Initialize x_0.

ii. Compute the residual $r_0 = f - Ax_0$ and choose a direction of $p_0 = r_0 - fAx_0$.

iii. Obtain the new x_i, $r_i = f - Ax_i$ and the direction p_i then compute the new estimate x_{i+1} and its residual r_{i+1} by using the formulas

$$\alpha_i = \frac{r_i^T r_i}{p_i^T A p_i},\ x_{i+1} = x_i + \alpha_i p_i, r_{i+1} = r_i - \alpha_i A p_i$$

iv. Next find the direction of p_{i+1} by using the formulas and repeat step (iii)

$$p_{i+1} = r_{i+1} - \beta_i p_i \text{ where } \beta_i = \frac{r_{i+1}^T r_{i+1}}{r_i^T r_i}$$

v. Check the convergence. If yes, go to step (vi). Otherwise go back to step (iii).

vi. Display the approximate solutions.

4. Numerical experiment

In order to verify the performances of the CG iterative method, a numerical experiment is conducted by solving the following two-point boundary value problems [8].

Problem 1

$$y'' - 4y = 4\cosh(1),\ x \in [0,1] \tag{16}$$

given that the exact solution for problem (16) is

$$y(x) = \cosh(2x-1) - \cosh(1).$$

Problem 2

$$-\frac{d^2y}{dx^2} = 9\sin(3x),\ x \in [0,1] \tag{17}$$

with its exact solution given by

$$\sin(3x)$$

The analysis and results of the performances in terms of iterations number (Iter), execution time (Second) and maximum absolute error (MAE) for Problem 1 and Problem 2 are presented and discussed in the next section.

5. Result and discussion

Based on the numerical experiment, the performances results of Problem 1 and Problem 2 are successfully tabulated in Table 1 and 2, respectively. Both tables show that as the matrix sizes increasing, the iterations number generated by the three iterative methods are also increasing. This is due to the accumulated round-off error that occurred at every iteration. However, it can be observed that the CG iterative method performs better than SOR and GS iterative methods as the matrix sizes are being incremented, and this is evidently presented through the difference of iterations number, execution time and maximum absolute error at different matrix sizes (128, 256, 512, 1024 and 2048).

Other than lesser iterations number, the CG iterative method also requires shorter execution time in order to iterate and converge to the exact solution, when solving the two-point boundary value problems. In fact, by going down the tables, the performances of the CG iterative method can be seen improving over SOR and GS methods for different matrix sizes especially the accuracy which given by maximum absolute error. This indicates that CG iterative method can cope with the accumulated round-off error better than SOR and GS method when solving the two-point boundary value problems together with the quartic non-polynomial spline scheme. Hence, it can be stated that CG iterative method has better performances compared to SOR and GS iterative methods when solving the problems.

Table 1

Comparison of GS, SOR and CG iterative methods in terms of iterations number (Iter), execution time (Seconds) and maximum absolute errors (MAE) for Problem 1

Method \ Matrix Sizes	128	256	512	1024	2048
Iterations Number					
GS	18173	66139	238353	848604	2975185
SOR	382	807	1438	2821	5367
CG	65	129	257	513	1025
Execution Time					
GS	14.40	49.09	168.88	662.96	83318.09
SOR	1.52	1.45	3.95	6.32	8.75
CG	0.16	0.54	1.44	2.31	2.62
Maximum Absolute Error					
GS	1.1788e-07	4.7242e-07	1.8899e-06	7.5601e-06	3.0241-e05
SOR	4.0171e-10	5.8395e-09	4.3379e-09	9.6408e-09	1.8447-e08
CG	1.1833e-10	7.2046e-12	1.8359e-13	2.1296e-12	8.1399e-12

Table 2

Comparison of GS, SOR and CG iterative methods in terms of iterations number (Iter), execution time (Seconds) and maximum absolute errors (MAE) for Problem 2

Method \ Matrix Sizes	128	256	512	1024	2048
Iterations Number					
GS	25950	94591	341534	1218827	4286118
SOR	735	976	2703	5174	9181
CG	128	256	512	1024	2048
Execution Time					
GS	37.06	88.40	325.63	1220.80	6231.12
SOR	1.68	2.59	4.12	7.35	13.76
CG	0.57	1.14	1.60	1.99	4.21
Maximum Absolute Error					
GS	1.6485e-07	6.6388e-07	2.6560e-06	1.0624e-05	4.2498e-05
SOR	2.6789e-09	6.9686e-10	1.4228e-08	2.8422e-08	5.1772e-08
CG	1.0363e-09	6.4801e-11	4.1241e-12	1.1072e-13	5.3167e-12

6. Conclusion

In conclusion, an approximate equation to solve two-point boundary value problems was successfully developed based on a quartic non-polynomial spline so that a system of linear equations can be constructed. Then, this linear system was solved by using three iterative methods which are the GS, SOR and CG iterative methods. Based on the results of performances experiment, the CG method was found to be superior compared to the GS and SOR method, and it is evidently proven through the comparison shown by the CG method in terms of iterations number, execution time and maximum absolute error at different respective matrix sizes. Therefore, it can be summarized that, the approximate solution obtained from the discretization of two-point boundary value problems by using the quartic non-polynomial scheme to form a linear system is best solved with the CG iterative method.

References

[1] Ozışık M.N., Boundary value problems of heat conduction, Courier Corporation, 1989.

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

[3] Chew J.V., Sulaiman J., Implicit solution of 1D nonlinear porous medium equation using the four-point Newton-EGMSOR iterative method, Journal of Applied Mathematics and Computational Mechanics 2016, 15(2), 11-21.

[4] Jang B., Two-point boundary value problems by the extended Adomian decomposition method, Journal of Computational and Applied Mathematics 2008, 219(1), 253-262.

[5] Kong D.X., Sun Q.Y., Two-point boundary value problems and exact controllability for several kinds of linear and nonlinear wave equations, In Journal of Physics: Conference Series 2011, 290(1), 012008. IOP Publishing.

[6] Chen B., Tong L., Gu Y., Precise time integration for linear two-point boundary value problems, Applied Mathematics and Computation 2006, 175(1), 182-211.

[7] Ha S.N., A nonlinear shooting method for two-point boundary value problems, Computers & Mathematics with Applications 2001, 42(10), 1411-1420.

[8] Mohsen A., El-Gamel M., On the Galerkin and collocation methods for two-point boundary value problems using sinc bases, Computers & Mathematics with Applications 2008, 56(4), 930-941.

[9] Fang Q., Tsuchiya T., Yamamoto T., Finite difference, finite element and finite volume methods applied to two-point boundary value problems, Journal of Computational and Applied Mathematics 2002, 139(1), 9-19.

[10] Albasiny E.L., Hoskins W.D., Cubic spline solutions to two-point boundary value problems, The Computer Journal 1969, 12(2), 151-153.

[11] Ramadan M.A., Lashien I.F., Zahra W.K., Quintic nonpolynomial spline solutions for fourth order two-point boundary value problem, Communications in Nonlinear Science and Numerical Simulation 2009, 14(4), 1105-1114.

[12] Kelley C.T., Iterative Methods for Linear and Nonlinear Equations, SIAM, Philadelphia 1995.

[13] Beauwens R., Iterative solution methods, Applied Numerical Mathematics 2004, 51(4), 437-450.

[14] Hestenes M.R., Stiefel E., Methods of conjugate gradients for solving linear systems, NBS 1952, 49, 1.

[15] Saad Y., Iterative methods for sparse linear systems, SIAM, 2003.

[16] Hackbusch W., Iterative Solution of Large Sparse Systems of Equations, Springer Science & Business Media, 2012.

[17] Young D.M., Iterative solution of large linear systems, Elsevier, 2014.

[18] Young D.M., Second-degree iterative methods for the solution of large linear systems, Journal of Approximation Theory 1972, 5(2),137-148.

[19] Abdullah A.R., Ibrahim A., Solving the two-dimensional diffusion-convection equation by the four point explicit decoupled group (edg) iterative method, International Journal of Computer Mathematics 1995, 58(1-2), 61-71.

[20] Evans D.J., Yousif W.S., The implementation of the explicit block iterative methods on the balance 8000 parallel computer, Parallel Computing 1990, 16(1), 81-97.

[21] Islam S.U., Tirmizi I.A., A smooth approximation for the solution of special non-linear third-order boundary-value problems based on non-polynomial splines, International Journal of Computer Mathematics 2006, 83(4), 397-407.

[22] Ramadan M.A., Lashien I.F., Zahra W.K., Polynomial and nonpolynomial spline approaches to the numerical solution of second order boundary value problems, Applied Mathematics and computation 2007, 184(2), 476-484.

[23] Siddiqi S.S., Akram G., Solution of fifth order boundary value problems using nonpolynomial spline technique, Applied Mathematics and Computation 2006, 175(2), 1574-1581.