International Journal of Engineering, Science and Mathematics
Vol. 6Issue 6, October 2017,
ISSN: 2320-0294 Impact Factor: 6.765
Journal Homepage: http://www.ijmra.us, Email: editorijmie@gmail.com
Double-Blind Peer Reviewed Refereed Open Access International Journal - Included in the International Serial Directories Indexed & Listed at: Ulrich's Periodicals Directory ©, U.S.A., Open J-Gage as well as in Cabell’s Directories of Publishing Opportunities, U.S.A

Applications of cubic splines in the numerical solution of polynomials

Najmuddin Ahmad[1] and ,Khan Farah Deeba[2]

Department of Mathematics, Integral University Lucknow

Abstract: In this paper we introduce different algorithm for reconstruction of a one dimensional function from its zero crossings. However, none of them is stable and computable in real time. An algorithm for computing the cubic spline interpolation coefficients for polynomials is presented in this paper. The matrix equation involved is solved analytically so that numerical inversion of the coefficient matrix is not required. For f(t) = t^m , a set of constants along with the degree of polynomial m are used to compute the coefficients so that they satisfy the Interpolation constraints but not necessarily the derivative constraints. Then, another matrix equation is solved analytically to take care of the derivative constraints. The results are combined linearly to obtain the unique solution of the original matrix equation. This algorithm is tested and verified numerically for various examples.

1. Introduction

In the mathematical field of numerical analysis, interpolation is a method of constructing new data point within the range of a discrete set of known data points. In a more informal language, interpolation means a guess at what happens between two values already known. In Engineering and Environmental sciences application, data collected from the field are usually discrete, therefore a more analytically controlled function that fits the field data is desirable and the process of estimating the outcomes in between these desirable data points is achieved through interpolation .
There are two main uses of interpolation. The first use is in reconstructing the function (x) when it is not given explicitly and/or only the values of (x) and certain order derivatives at a set of points, called nodes are known. Secondly interpolation is used to replace the function (x) by an interpolating polynomial (x) so that many common operations like the determination of roots, differentiation, integration etc. which are intended for the function (x) may be performed using the interpolate (x) .
Spline interpolation is a form of interpolation where the interpolate is a special type of piecewise polynomial called spline. Spline interpolation is often preferred over polynomial interpolation because the interpolation error can be made small even when using low degree polynomials for the spline . Spline interpolation avoids the problem of Runge’s phenomenon, in which oscillation occurs between points when interpolating using high degree polynomials .
Cubic Spline interpolation is a special case of spline interpolation that is used very often to avoid the problem of Runge’s phenomenon. This method gives an interpolating polynomial that is smoother and has smaller error than other interpolating polynomials such as Lagrange polynomial and Newton polynomial.

This research work focuses on the application of spline cubic interpolation and piecewise cubic interpolation on heat transfer in selected lakes. The thermal stratification of the lakes informs the temperature difference at each layer of the lake. However, the values of the interval between the thermo cline depth, temperatures at these intervals and the heat flux will be used to derive equations using cubic spline interpolation and piecewise cubic interpolation. The results from these equations will then be used to obtain the thermo cline depths, thermo cline temperatures and the thermo cline heat flux for the selected lakes. A comparative analysis of the values obtained using the cubic spline interpolation and piecewise cubic interpolation will be made to determine the method with the least percentage error.

2. An Overview of approximation and interpolation theory

Interpolation is used to estimate the value of a function between known data points without knowing the actual function. Two main broad categories of interpolation exists; global and piecewise interpolation. Global interpolation methods use a single equation that maps all the data points into an nth order polynomial. These methods result in smooth curves, but in many cases they are prone to severe oscillation and overshoot at Intermediate points. Piecewise interpolation method uses a polynomial of low degree between each pair of known data points. If a first degree polynomial is used, it is called linear interpolation, for second and third degree polynomial; it is called quadratic and cubic spline respectively. The higher the degree of the spline, the smoother the curve spline of degree m, will have continuous derivative up to $(m - 1)$ at the data points.

Curve fitting is the process of finding a curve that could best fit a given set of data. There are two approaches of fitting a curve from a set of data points. The first approach called collocation is the case where the curve is made to pass through all data points. This approach is used either when the data is known to be accurate or the data are generated from the evaluation of some complicated function at discrete set of points. Such function could be polynomial, trigonometric or exponential functions. The second approach is when a given curve is made to represent the general trend of the data. This approach is useful when there are more data points than the number of unknown coefficient or when the data appear to have a significant error or noise.

Interpolating a set of data points can be done using polynomial, spline function or Fourier series. However polynomial interpolation is commonly used and many numerical methods are based on polynomial approximations.
For a given set of (n + 1) data points (x_0, y_0), (x_1, y_1), … , (x_n, y_n), it is of interest to find n^{th} order polynomial function that can match these data points. The n^{th}polynomial function is given as

$$P_n = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n$$

The coefficients can be obtained by solving a set of algebraic equations

$$a_0 + a_1 x_0 + a_2 x_0^2 + \cdots + a_n x_0^n = y_0$$
$$a_0 + a_1 x_1 + a_2 x_1^2 + \cdots + a_n x_1^n = y_1$$

…………………………………..
…………………………………..
…………………………………..

$$a_0 + a_1 x_n + a_2 x_n^2 + \cdots + a_n x_n^n = y_n$$

As the number of data points increases, so also that of the unknown variables and equations, the resulting system of equations may not be so easy to solve. However, there are a number of alternative forms of expressing an interpolating polynomial beyond the familiar format stated above. Among them are Lagrange, Newton's forward and backward difference, and Hermite interpolations.
The most common spline and piecewise interpolation used are linear, quadratic and cubic respectively. To obtain a smoother curve, cubic splines are frequently recommended, because they provide the simplest representation that exhibits the desired appearance of smoothness. They are generally well

behaved and continuous up to the second order derivative at the data points. Even though cubic splines are less prone to oscillation or overshooting due to instability inherent in higher order polynomial than global polynomial equations, they do not prevent it. Thus, to avoid these oscillations, it is common to divide the interval into sub-interval and approximate the function using low degree polynomial on each sub-interval.

2.1 Mathematical treatments

Give a function f on $[a, b]$ and nodes $a = x_0 < x_1 <. . . < x_n = b$, a cubic spline interpolant f satisfied the following conditions

1. f is a cubic polynomial on each subinterval $[x_i , x_{i+1}]$
2. $s_i\ (x_i\ ) = f_i(x_i) for\ i$ = 0,1,2, . . n (i.e the spline matches function values)
3. $s_i(x_{i+1}) = s_{i+1}(x_{i+1})= f_{i+1}$ for i = 0,1, … . n – 2 (i.e the spline is continuous)
4. $s_i'(x_{i+1}) = s_{i+1}'(x_{i+1})$ for i = 0,1, … . n – 2 (i.e the spline C^1)
5. $s_i''(x_{i+1}) = s_{i+1}''(x_{i+1})$ for i = 0,1, … . n – 2 (i.e the spline C^2)
6. i. $s_i''(x_0) = s_i''(x_n)$ = 0 (Natural Spline resulting from free boundary condition).
   ii. $s_i'(x_0) = f_i'(x_0) and\ s_i'(x_n) = f_i'(x_n)$ (clamped end condition resulting from clamped boundary condition)

In the derivation of the cubic spline interpolation polynomial, we follow the above conditions one after the other. The objective of the cubic spline is to derive a third-order polynomial for each interval between knots as represented by a general cubic spline polynomial function given below

$s_i(x) = a_i + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3$ (2.1)

The first condition is that the cubic spline must pass through all data points. Hence we have

$f_i = a_i + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3$ (2.2a)

$=> f_i = a_i + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3$ (2.2b)

Hence $f_i = a_i$ (2.3)

Therefore the constant in each cubic must be equal to the value of the dependent variable at the beginning of the interval. Substituting (2.3) in (2.1), we have

$s_i(x) = f_i + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3$ (2.4)

Next we apply the third condition that each of the cubic spline must join the knot.

For $(i + 1)$ knot, we have

$s_{i+1}(x) = f_i + b_i(x_{i+1} - x_i) + c_i(x_{i+1} - x_i)^2 + d_i(x_{i+1} - x_i)^3$

⇨ $f_{i+1} = f_i + b_ih_i + c_ih_i^2 + d_ih_i^3$ (2.5)

$where\ h_i = x_{i+1} - x_i$

Applying the fourth condition, the first derivation at the interior nodes must be equal.

Hence differentiating (2.4) we get

$s_i'(x) = b_i + 2c_i(x - x_i) + 3d_i(x - x_i)^2$ (2.6)

The equivalent of the derivative at the interior node $(i + 1)$ is given as

$b_i + 2c_ih_i + 3d_ih_i^2 = b_{i+1}$ (2.7)

Applying the fifth condition, the second derivative at the interior nodes must also be equal.

Hence differentiating equation (2.6) we get

$s_i''(x) = 2c_i + 6d_i(x - x_i)$ (2.8)

The equivalent of the second derivative at the interior node $(i + 1)$ is given as

$$c_i + 3d_i h_i = c_{i+1} \quad (2.9)$$

Solving (2.9) for di, we get

$$d_i = \frac{c_{i+1} - c_i}{3h_i} \quad (2.10)$$

Substituting (2.10) into (2.5) we get

$$f_{i+1} = f_i + b_i h_i + \frac{h_i^2}{3}(2c_i + c_{i+1}) \quad (2.11)$$

Substituting (2.10) into (2.7) we get

$$b_{i+1} = b_i + h_i(c_i + c_{i+1}) \quad (2.12)$$

Equation (2.11) can be solved for b_i to get

$$b_i = \frac{f_{i+1} - f_i}{h_i} - \frac{h_i}{3}(2c_i + c_{i+1}) \quad (2.13)$$

Reducing the index by 1, equation (2.12) and (2.13)can be rewritten as

$$b_{i-1} = \frac{f_i - f_{i-1}}{h_{i-1}} - \frac{h_{i-1}}{3}(2c_{i-1} + c_i)\,, \quad (2.14)$$

$$b_i = b_{i-1} + h_{i-1}(c_{i-1} + c_i) \quad (2.15)$$

Substituting equation (2.13) and (2.14) in (2.15) we get

$$\frac{f_{i+1} - f_i}{h_i} - \frac{h_i}{3}(2c_i + c_{i+1}) = \frac{f_i - f_{i-1}}{h_{i-1}} - \frac{h_{i-1}}{3}(2c_{i-1} + c_i) + h_{i-1}(c_{i-1} + c_i)$$

On Further simplification

$$\mathrm{h}_{i-1} c_{i-1} + 2(\mathrm{h}_{i-1} + \mathrm{h}_i)\mathrm{c}_i + \mathrm{h}_i c_{i+1} = 3\left(\frac{f_{i+1} - f_i}{\mathrm{h}_i}\right) - 3\left(\frac{f_i - f_{i-1}}{h_{i-1}}\right) \quad (2.16)$$

Using divided difference notation, $[x_i, x_j] = \frac{f_i - f_j}{x_i - x_j}$

Equation (2.16) can be written as

$$\mathrm{h}_{i-1} c_{i-1} + 2(\mathrm{h}_{i-1} + \mathrm{h}_i)\mathrm{c}_i + \mathrm{h}_i c_{i+1} = 3(f[x_{i+1}, x_i] - f[x_i, x_{i-1}]) \quad (2.17)$$

Equation (2.17) can be written for the interior knots, i =2, 3, . . ., n − 2 , which result in (n − 3) simultaneous tridiagonal equations with (n − 1) unknown coefficient $c_1, c_2, \ldots c_{n-1}$. Using the last condition (boundary condition), the second derivative at the first node (that is equation (2.8)) can be set to zero, given

$$s_i''(x) = 2c_i + 6d_i(x - x_i) = 0 \Rightarrow c_i = 0 \quad (2.18)$$

Similarly the same evaluation can be made at the last node, given as

$$s_{n-1}''(x_n) = 2c_{n-1} + 6d_{n-1}h_{n-1} = 0 \quad (2.19)$$

Recalling (2.9), equation (2.19) became

$$c_{n-1} + 3d_{n-1}h_{n-1} = c_n = 0 \quad (2.20)$$

Thus to impose a zero second derivative at the last node, we set c_n= 0

2.2 Determination of the temperature of the thermo cline depth

The location of the thermocline depth is defined as the inflection point of the temperature-depth curve, that is the point at which $\frac{d^2T}{dz^2}$ =0. We calculate the thermocline depth of the lakes in the study area using $\frac{d^2T}{dz^2}$ =0

2.3 Determination of heat flux across the thermo cline

According to Fourier's law of heat conduction, heat flows from the regions of high temperature to low temperature. For the one-dimensional case, this can be expressed mathematically as $q = -k\frac{dT}{dz}$
Where (x) = heat flux (w/m^2), k = coefficient of thermal conductivity $W/(m.k)$, T = Temperature(k) x = Distance(m).
The temperature gradient is important in its own right because it can be used in conjunction with Fourier's law to determine the heat flux across the thermocline. Heat flux is defined as the amount of heat transferred per unit area per unit time from or to a surface (John and John, 2008). It is calculated using the formula = $-\alpha\rho c\frac{dT}{dz}$,
where J = heat flux$[J/(m^2.s)]$, α = an eddy diffusion coefficient $(10^{-3}\frac{m^2}{s})$, = $density$ $(\cong 1000kg/m^3)$ and c = specific heat $[\cong 4200J/(kgK)]$

From the twenty one equations (21) obtained (i.e. in this study seven equations each for the three lakes sampled), three optimal equations that represents the thermo cline depth for the three lakes were chosen. These optimal equations were differentiated and used to obtain the thermo cline gradient $(\frac{dT}{dz})$. These gradients were substituted into the heat flux equation to obtain the values of the heat flux across the thermocline for the three lakes.

2.4 Absolute Relative Error between the analytic solution and the two methods

The absolute relative approximate error $\in_a$ obtained between the analytic solution of the temperature results with the Cubic Piecewise Interpolation and Cubic Spline Interpolation error is given below:

The error for Analytic solution with the Cubic Spline Interpolation is given by

$$|\in_{aspline}| = |\frac{p^{**}(z)-p(z)}{p^{**}(z)}| \times 100$$

Where $p{*}{*}(z)$ = Temperature obtained by Analytic Solution

(z) = Temperature estimate obtained by Cubic Spline Interpolation

The error for Analytic solution with the Cubic Piecewise Interpolation is given by

$$|\in_{apiecewise}| = |\frac{p^{**}(z)-p^{*}(z)}{p^{**}(z)}| \times 100$$

Where $p{*}{*}$ (z) = Temperature obtained by Analytic Solution

$p{*}(z)$ = Temperature estimate obtained by Cubic Piecewise Interpolation

3. Result

Table 4.1 shows the results of the values of temperatures at various depths that were determined from the three lakes selected for the study . For each of the lakes the temperature reading at 1 m each was taken after 10 minutes interval and their corresponding depths were recorded after 1 m each until a depth of 8 m was reached

Table 3.1 Temperature and depth taken for three lakes

Lake 1		**Lake 2**		**Lake 3**	
Depth(z)	**Temp.(^{0}C)**	**Depth(z)**	**Temp.(^{0}C)**	**Depth(z)**	**Temp.(^{0}C)**
1.0	**30**	**1.0**	**35**	**1.0**	**34**
2.0	**30**	**2.0**	**35**	**2.0**	**34**
3.0	**30**	**3.0**	**34**	**3.0**	**34**
4.0	**28**	**4.0**	**32**	**4.0**	**30**
5.0	**23**	**5.0**	**26**	**5.0**	**23**
6.0	**19**	**6.0**	**22**	**6.0**	**21**
7.0	**18**	**7.0**	**21**	**7.0**	**20**
8.0	**18**	**8.0**	**21**	**8.0**	**20**

Figures 3.1, 3.2 and 3.3 are graphical representation of the analytical results presented in Table 4.1. The graph shows plot for the temperature against depth for the three lake site respectively. The graphs show that with increasing depths of lakes there was decreasing temperature.

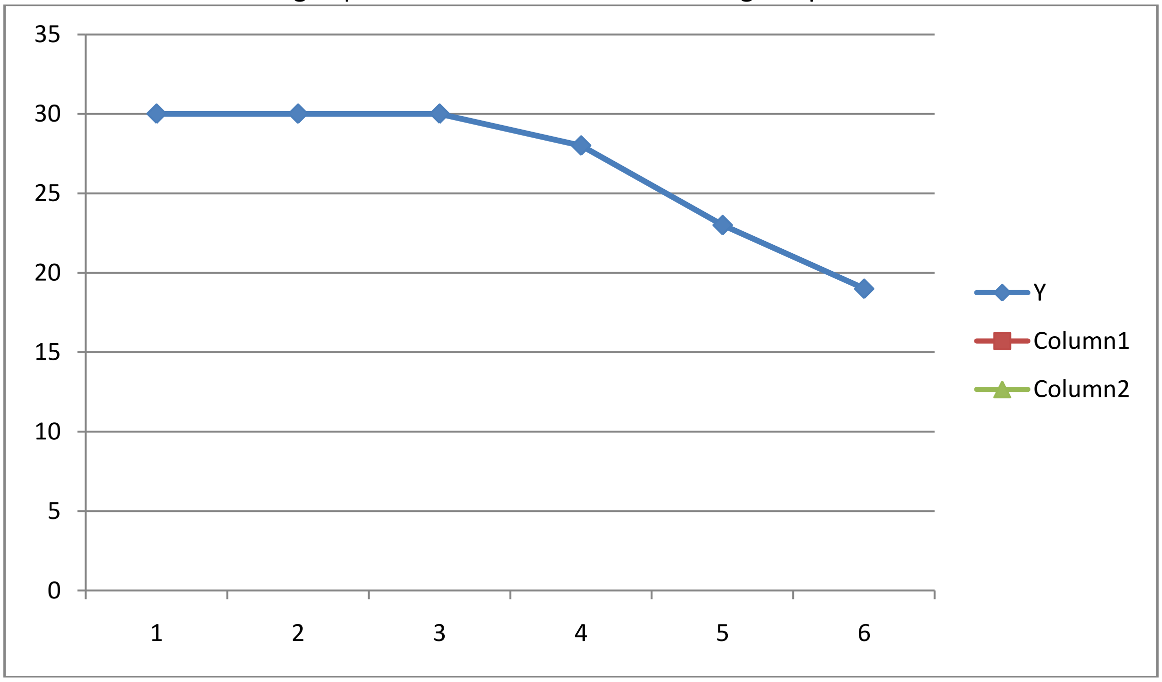

Figure 3.1

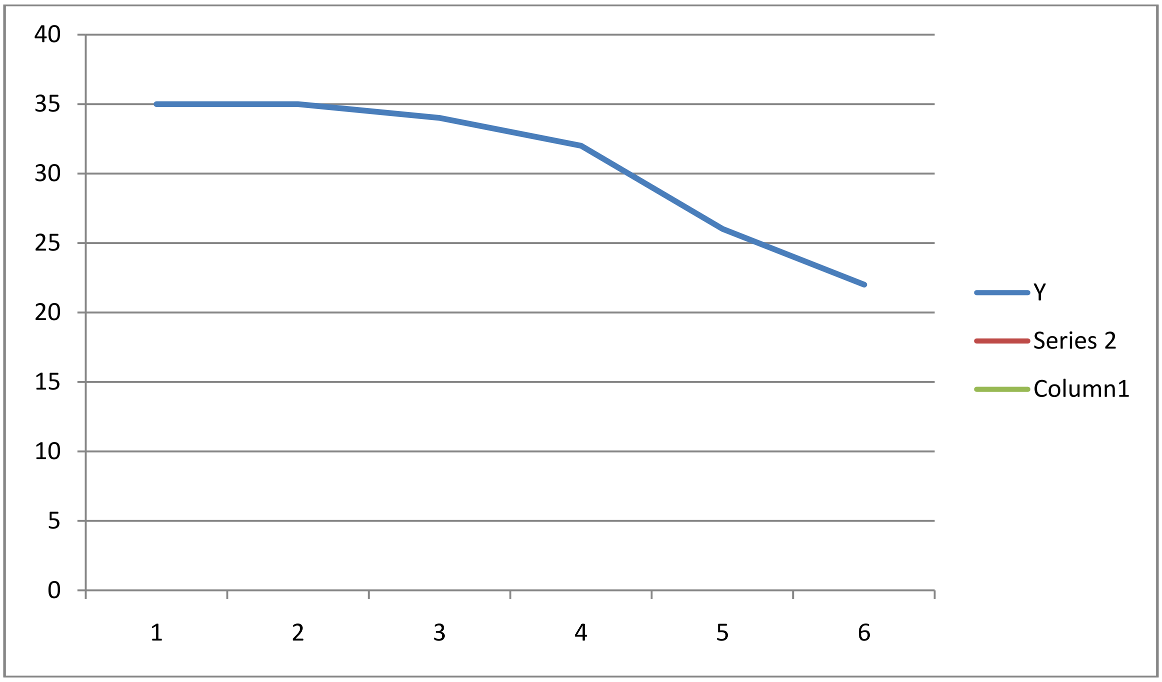

Figure 3.2

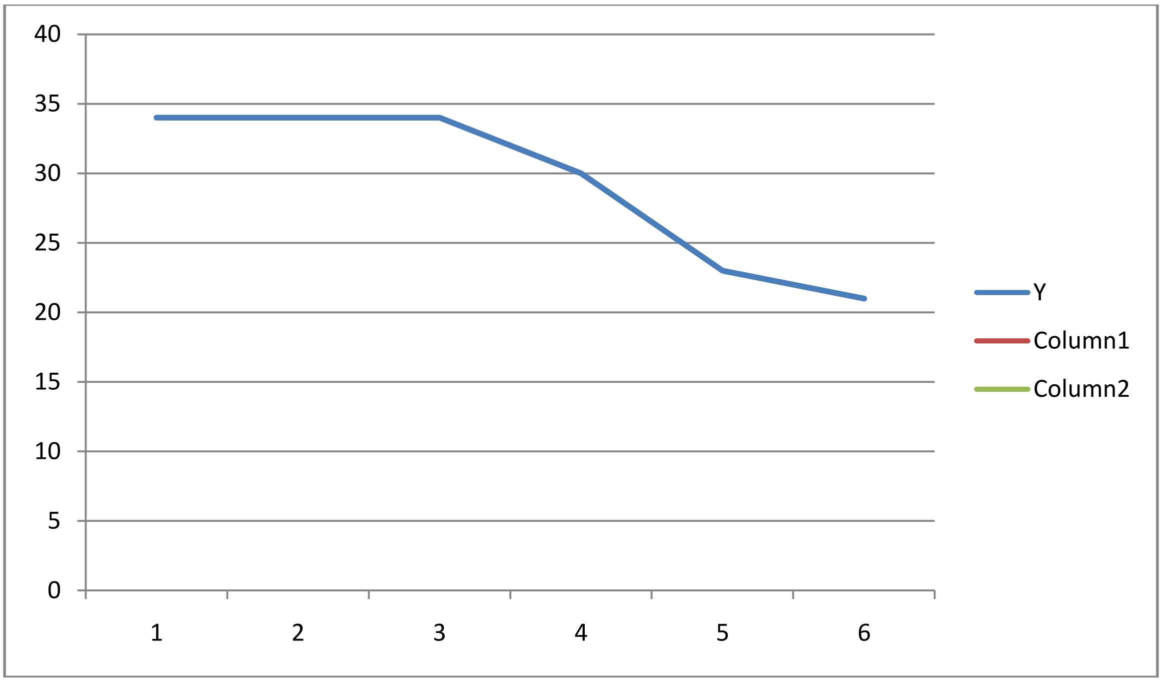

Figure 3.3

Conclusion

The results obtained from this work showed that the thermo cline thicknesses as well as the thermo cline depth vis-a-vis the heat transfer for the lakes were found to be in the interval of 4.0m and 5.0m depth. The cubic spline was derived and the result was use to approximate thermo cline depth and temperature.

The cubic spline interpolation method showed less percentage error when compared to the analytical result, while the results obtained from the cubic piecewise interpolation method showed a higher percentage error when compared to the analytical results.

Acknowledgement : Manuscript communication number (MCN) : IU/R&D/2017- MCN000164 office of research and development integral university , Lucknow.

.Refferences

1. Hazewinkel, Michiel, ed. (2001); "Spline interpolation", Encyclopedia of Mathematics, Springer, ISBN 978-1-55608-010-4.

2. Henrici, P. (1982): Essential of numerical analysis, New York. John Wiley & Sons.

3. Jain, M.K. And Iyenger, S.R.K. Numerical methods for Scientific and Engineering Computation. New Age International Publishers, 2015;

4. Wang, K. A study of cubic spline interpolation, River Academic Journal, Volume 9, Number 2, fall, 2013;

5.Karim, S.A.A. and Rosli, M.A.M. Cubic spline interpolation for petroleum engineering data, Applied Mathematical Sciences, Volume 8, 2014; Number 102.

6. Kim, H.K, (2005): Spline and Piecewise Interpolation. Slightly modified 3/1/09, 2/28/06 Firstly written at March 2005.

7. Rai, D. Computer Based Numerical and Statistical Techniques, 2012;

8. Srivastava, P.K. Study of differential equations with their polynomial and nonpolynomial spline based approximation, 2014;

9. Moon, B.S. An Explicit Solution of the Cubic Spline Interpolation for Polynomials, Korea Atomic Energy Research Institute, Korea 305-600.

10. Mihalic, J., Zavacky, J. and Kuba, I. Spline Interpolation of Image, Department of radioelectronics, Volume 4, number 1, 1995;