JURNAL IPTEKS TERAPAN
Research of Applied Science and Education V14.i3 (205-210)

ISSN : 1979-9292
E-ISSN : 2460-5611

THE APPLICATION OF TRIDIAGONAL MATRIX ALGORITHM IN CUBIC SPLINE INTERPOLATION

Elvathna Syafwan[1)*], Efendi[2)], Narwen[3)], Mahdhivan Syafwan[4)]
[1]Mechanical Engineering Study Program , Politeknik TEDC Bandung
[2,3,4] Department of Mathematics, Universitas Andalas
*email: elvathna@poltektedc.ac.id

ABSTRACT

Tridiagonal matrix algorithm is known to be more efficient in the computational process than the Gaussian elimination method in solving linear system problems involving tridiagonal matrices. In this paper, the tridiagonal matrix algorithm is applied to the cubic spline interpolation problem with natural boundary conditions. In this case, the tridiagonal matrix algorithm plays a role in finding the second derivative of each cubic spline sub-function so that it is more efficient in obtaining the coefficients of the third order polynomials that form the cubic spline function.

Keywords: *Tridiagonal matrix algorithm, cubic spline interpolation*

INTRODUCTION

Interpolation is one of the curve matching techniques in numerical methods [1]. Interpolation curves can be constructed by a polynomial function of low order over a subset of data points. This interpolation is called spline interpolation. Spline consists of three types, namely linear splint, quadratic spline, and cubic spline. The most commonly used spline is the cubic spline because it gives better approximation and produces smoother curves [2].

The purpose of cubic spline interpolation is to derive a third order polynomial in each interval between the given data points, where the coefficients of the third order polynomial in each interval will be searched for values. It is known that the second derivative of each sub-function in the cubic spline interpolation determines values of these coefficients [1,2]. This problem can then be modeled in the form of a tridiagonal matrix system where the vector variable contains the second derivative of the sub-sub-function in the cubic spline interpolation.

To determine the solution of a matrix system, the Gauss elimination method is often used. However, because elements with zero values are more dominant in the tridiagonal matrix, especially with very large sizes, it is necessary to apply a method that can streamline the computational process used in solving the tridiagonal system. One method that can be used is to apply a tridiagonal matrix algorithm.

This algorithm was first developed by Llewellyn Thomas, so it is also known as the Thomas algorithm [3].

This paper will discuss how to determine the value of the second derivative of the cubic spline function using the tridiagonal matrix algorithm so

Submitted : 25 Agt 2020 – Accepted : 26 Sep 2020 – Published : 27 Sep 2020

ISSN : 1979-9292
E-ISSN : 2460-5611

that the curve of the cubic spline function can be made.

TRIDIAGONAL MATRIX ALGORITHM

The tridiagonal matrix algorithm is used in solving the tridiagonal system of equations. A tridiagonal system for n unknowns can be written

$$u_i z_{i-1} + v_i z_i + w_i z_{i+1} = r_i,$$

where $u_1 = 0$ and $w_n = 0$. In matrix form, this system is written as follows:

$$\begin{bmatrix} v_1 & w_1 & 0 & \cdots & \cdots & 0 \\ u_2 & v_2 & w_2 & \cdots & \cdots & 0 \\ 0 & u_3 & v_3 & w_3 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & \cdots & u_{n-1} & v_{n-1} & w_{n-1} \\ 0 & 0 & \cdots & 0 & u_n & v_n \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ \vdots \\ z_{n-1} \\ z_n \end{bmatrix} = \begin{bmatrix} r_1 \\ r_2 \\ r_3 \\ \vdots \\ r_{n-1} \\ r_n \end{bmatrix}. \qquad (1)$$

This tridiagonal matrix algorithm consists of two main steps [3]. The first step in the tridiagonal matrix algorithm is to modify the coefficients to be as follows:

$$\widetilde{w}_i = \begin{cases} \dfrac{w_1}{v_1}, & i = 1 \\ \dfrac{w_i}{v_i - \widetilde{w}_{i-1} u_i}, & i = 2, 3, \dots, n-1 \end{cases}$$

and

$$\tilde{r}_i = \begin{cases} \dfrac{r_1}{v_1}, & i = 1 \\ \dfrac{r_i - \tilde{r}_{i-1} u_i}{v_i - \widetilde{w}_{i-1} u_i}, & i = 2, 3, \dots, n \end{cases}$$

The above steps are a form of forward elimination.

The next step is to determine the solution obtained by backward substitution as follows:

$$z_n = \tilde{r}_n \text{ and } x_i = \tilde{r}_i - \widetilde{w}_i z_{i+1};\quad i = n-1, n-2, \dots, 1$$

Note:

1. This algorithm can only be used on matrices where the diagonal is predominant, ie $|v_i| \geq |u_i| + |w_i|$ untuk $i = 1, 2, \dots, n$ and at least one *i* such that $|v_i| > |u_i| + |w_i|$.
2. The number of arithmetic operations required in the algorithm is a number of $10n - 11$ operations. Meanwhile, the elimination of Gauss takes as much $\frac{2n^3}{3} + \frac{3n^2}{2} - \frac{7n}{6}$ arithmetic operations [4]. Thus the tridiagonal matrix algorithm is more efficient in finding a tridiagonal system solution than the Gaussian elimination for $n \geq 3$.

CUBIC SPLINE INTERPOLATION

Suppose there are *n* data points in the *xy*-plane,

$$(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$$

whose values are known. These data points are interpolated using the *cubic spline* method, that is, by constructing a function

$$P(x) = \begin{cases} p_1(x), & x_1 \leq x \leq x_2, \\ p_2(x), & x_2 \leq x \leq x_3, \\ \vdots & \\ p_{n-1}(x), & x_{n-1} \leq x \leq x_n, \end{cases}$$

where $p_j(x)$, $j = 1, 2, \dots, n-1$ is the

Submitted : 25 Agt 2020 – Accepted : 26 Sep 2020 – Published : 27 Sep 2020

ISSN : 1979-9292
E-ISSN : 2460-5611

cube polynomial defined by

$$p_j(x) = a_j + b_j(x - x_j) + c_j(x - x_j)^2 + d_j(x - x_j)^3, \quad (2)$$

that applies at each interval $x_j \le x \le x_{j+1}$, as a_j, b_j, c_j, and d_j are the coefficients whose values will be searched to determine $P(x)$. Note that the distance between two adjacent data points at the x-coordinate can be generally expressed as $h_j = x_{j+1} - x_j$, as $j = 1, 2, \dots, n-1$.

The function $P(x)$ built on cubic spline interpolation is a function that satisfies the following four properties:

(1) $P(x)$ through all data points
(2) $P(x)$ continuous at the interval $[x_1, x_n]$,
(3) $P'(x)$ continuous at the interval $[x_1, x_n]$
(4) $P''(x)$ continuous at the interval $[x_1, x_n]$

Using the four properties above, the values for the coefficients a_j, b_j, c_j, and d_j are obtained in equation (2), as given in the following theorem [4].

Theorem (Cubic Spline Interpolation)
Let n data points are given $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$ *as* $h_j = x_{j+1} - x_j$, $j = 1, 2, \dots, n-1$. *Cubic spline*

$$P(x) = \begin{cases} p_1(x), & x_1 \le x \le x_2, \\ p_2(x), & x_2 \le x \le x_3, \\ & \vdots \\ p_{n-1}(x), & x_{n-1} \le x \le x_n, \end{cases}$$

where

$$p_j(x) = a_j + b_j(x - x_j) + c_j(x - x_j)^2 + d_j(x - x_j)^3,$$

in each interval $x_j \le x \le x_{j+1}$, *which interpolates the given data points, has the following coefficients:*

$$a_j = y_j,$$
$$b_j = \frac{y_{j+1} - y_j}{h_j} - \left[\frac{P''(x_{j+1}) + 2P''(x_j)}{6}\right] h_j,$$
$$c_j = \frac{P''(x_j)}{2},$$
$$d_j = \frac{P''(x_{j+1}) - P''(x_j)}{6h_j},$$

for $j = 1, 2, \dots, n-1$.

From this theorem it can be seen that the coefficients *bj*, *cj*, and *dj* depend on the values of the unknowns $P''(x_1), P''(x_2), \dots, P''(x_n)$. Therefore, it is necessary to find these values in order to determine the cubic spline function $P(x)$.

By performing a mathematical analysis of the properties of the cubic spline function $P(x)$, the following matrix equation is obtained:

$$H\mathbf{m} = 6\mathbf{k}, \quad (3)$$

where

$$H = \begin{bmatrix} h_1 & w_1 & h_2 & \cdots & 0 & 0 & 0 \\ 0 & h_2 & w_2 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & w_{n-3} & h_{n-2} & 0 \\ 0 & 0 & 0 & \cdots & h_{n-2} & w_{n-2} & h_{n-1} \end{bmatrix},$$

Submitted : 25 Agt 2020 – Accepted : 26 Sep 2020 – Published : 27 Sep 2020

JURNAL IPTEKS TERAPAN
Research of Applied Science and Education V14.i3 (205-210)

ISSN : 1979-9292
E-ISSN : 2460-5611

$$\mathbf{m} = \begin{bmatrix} P''(x_1) \\ P''(x_2) \\ \vdots \\ P''(x_{n-1}) \\ P''(x_n) \end{bmatrix},$$

$$\mathbf{k} = \begin{bmatrix} \frac{y_3-y_2}{h_2} - \frac{y_2-y_1}{h_1} \\ \frac{y_4-y_3}{h_3} - \frac{y_3-y_2}{h_2} \\ \vdots \\ \frac{y_{n-1}-y_{n-2}}{h_{n-2}} - \frac{y_{n-2}-y_{n-3}}{h_{n-3}} \\ \frac{y_n-y_{n-1}}{h_{n-1}} - \frac{y_{n-1}-y_{n-2}}{h_{n-2}} \end{bmatrix},$$

as $w_j = 2(h_j + h_{j+1}), j = 1, \dots, n-2.$

Note that system (3) consists of equations with $n-2$ variables. In order to obtain a unique solution, two additional equations are needed. These two additional equations can be obtained from the boundary conditions that apply to $P''(x_1)$ and $P''(x_n)$. There are several types of boundary conditions that can be used, one of which is natural boundary conditions, namely

$$P''(x_1) = 0 \text{ and } P''(x_n) = 0.$$

By adding the two equations above, $P''(x_1)$ and $P''(x_n)$ can be eliminated from system (3), so we get

$$H_1\mathbf{m}_1 = 6\mathbf{k}, \qquad (4)$$

where

$$H_1 = \begin{bmatrix} w_1 & h_2 & 0 & \cdots & 0 & 0 & 0 \\ h_2 & w_2 & h_3 & \cdots & 0 & 0 & 0 \\ 0 & h_3 & w_3 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & w_{n-4} & h_{n-3} & 0 \\ 0 & 0 & 0 & \cdots & h_{n-3} & w_{n-3} & h_{n-2} \\ 0 & 0 & 0 & \cdots & 0 & h_{n-2} & w_{n-2} \end{bmatrix},$$

$$\mathbf{m}_1 = \begin{bmatrix} P''(x_2) \\ P''(x_3) \\ P''(x_4) \\ \vdots \\ P''(x_{n-3}) \\ P''(x_{n-2}) \\ P''(x_{n-1}) \end{bmatrix}.$$

Note that system (2) is a tridiagonal matrix system, thus the solution for m_1 can be determined efficiently by applying the matrix algorithm as discussed in the previous section.

APPLICATION EXAMPLES.

For example, the velocity data of a parachutist (cm / sec) is given at each time (seconds) as in Table 1. Furthermore, the drop velocity is determined at any time along the time interval [1,13] (seconds) by using a spline cubic interpolation with natural boundary conditions.

Table 1. Data of Speed of a Parachutist [5]

Time (seconds)	1	3	5	7	13
Speed (cm / sec)	800	2310	3090	3940	4755

Suppose j represents the number of data, x_j represents the time in the j-th data, y_j represents the speed of the parachutist when x_j and $h_j = x_{j+1} - x_j$ represents the jth time interval. These results are given in Table 2.

Table 2. Data of Parachutist's Speed and Time Interval

j	1	2	3	4	5
x_j	1	3	5	7	13
y_j	800	2310	3090	3940	4755

Submitted : 25 Agt 2020 – Accepted : 26 Sep 2020 – Published : 27 Sep 2020

JURNAL IPTEKS TERAPAN
Research of Applied Science and Education V14.i3 (205-210)
ISSN : 1979-9292
E-ISSN : 2460-5611

h_j	2	2	2	6	

By inputting data in Table 2, system (3) becomes

$$\begin{bmatrix} 8 & 2 & 0 \\ 2 & 8 & 2 \\ 0 & 2 & 16 \end{bmatrix} \begin{bmatrix} P''(x_2) \\ P''(x_3) \\ P''(x_4) \end{bmatrix} = \begin{bmatrix} -2190 \\ 210 \\ -1735 \end{bmatrix}$$

Then use the tridiagonal matrix algorithm as follows:

1. Perform forward eliminations to modify the coefficients to be
$$\widetilde{w}_1 = 0.25$$
$$\widetilde{w}_2 = 0.27$$
and
$$\tilde{r}_1 = -273.75$$
$$\tilde{r}_2 = 101$$
$$\tilde{r}_3 = -125.24$$
2. Perform a backward substitution to obtain a solution
$$P''(x_4) = -125.24$$
$$P''(x_3) = 134.40$$
$$P''(x_2) = -307.35$$
$$P''(x_1) = P''(x_5) = 0.$$

Then by using the Cubic Spline Interpolation Theorem, the values of $a_j, b_j,\ c_j,$ and d_j are obtained as shown in Table 3.

Table 3. Coefficient values of the cubic spline function

j	a_j	b_j	c_j	d_j
1	800	857.45	0.00	-25.61
2	2310	550.10	153.67	36.81
3	3090	377.15	67.20	-21.64
4	3940	386.31	-62.62	-3.48

The values of the a_j, b_j, c_j, dan d_j coefficients then form the cubic spline function $P(x)$ that we want to find. The curve of the cubic spline function is given in Figure 1.

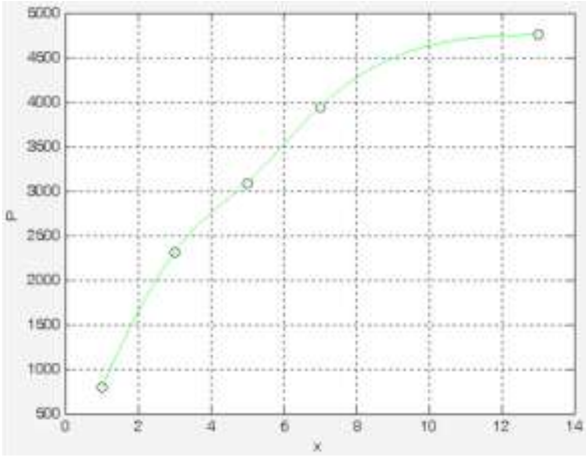

Figure 1. Natural Cubic Spline Curve

CONCLUSION

This paper has discussed the tridiagonal matrix algorithm which is known to be more efficient in the computational process than the Gaussian elimination method. This is because many arithmetic operations required by the tridiagonal matrix algorithm are much less than those required by the Gaussian elimination method.

The tridiagonal matrix algorithm is then applied to the cubic spline interpolation problem with natural boundary conditions. In this case, the tridiagonal matrix algorithm plays a role in finding the value of the second derivative of each cubic spline sub-function so that the third order polynomial coefficients that form the cubic spline function can be obtained.

ACKNOWLEGMENTS

This work was supported financially by a PDP research grant (Ref. 131 / SP2H / LT-MONO / LL4 / 2020, 025.3 / TEDC / MOU-DIR / IV / 2020)

Submitted : 25 Agt 2020 – Accepted : 26 Sep 2020 – Published : 27 Sep 2020

ISSN : 1979-9292
E-ISSN : 2460-5611

from the Ministry of Research and Technology, Agency for Research and Innovation (RISTEK-BRIN), The Republic of Indonesia.

REFERENCES

[1] Mathews, John H. and K.D. Fink. 2004. *Numerical Methods Using MATLAB.* Fourth Edition. Prentice-Hall**:** New Jersey

[2] Howard Anton. 2010. Elementary Linear Algebra. Tenth Edition. John Wiley and Sons: New York

[3] Joe D. Hoffman and Steven Frankel. 2018. *Numerical Methods for Engineers and Scientists*. Second Edition. CRC Press: Boca Raton.

[4] James F. Epperson. 2013. An *Introduction to Numerical Methods and Analysis*. Second Edition. John Wiley & Sons: New York.

[5] Scheid, F. 1992. *Analisis Numerik*. Alih Bahasa: Pantur Silaban. Erlangga, Jakarta.

Submitted : 25 Agt 2020 – Accepted : 26 Sep 2020 – Published : 27 Sep 2020