TWO DIMENSIONAL DATA MODELING AND CHEBYSHEV APPROXIMATION OF SPLINE

Fauziahanim Che Seman[1], ShahNor Basri[2] and Ahmad Faizal Mohd Zain[1]
[1]*Kolej Universiti Teknologi Tun Hussein Onn*, [2]*Universiti Putra Malaysia*
E-mail: fauziahs@kuittho.edu.my

ABSTRACT

This paper describes the approximation of discrete data using splines. The approximation method is adapted from the Chebyshev approximation. The procedures to find a set of extreme points for incoming discrete data are proposed. Several algorithms using cubic spline and Lagrange polynomial are proposed to diffrentiate the results due to the number of iteration, total number of the set of extreme points and error generated. The results show that the error generated decreases as the total number of extreme points increase. Six extreme points can represent one hundred of points and the generated error can be decreased. However, the algorithm presented uses more number of extreme points and will cause an increase in the total number of iterations.

Keywords : *Chebyshev Approximation, Extreme Points, Lagrange, Splines*

1 INTRODUCTION

Many engineering applications involve signal processing when analysing the incoming discrete data such as in the robotics motion design and speech waveform data. The resulting motion or moving frames of the robot can be represented by splines. The unimportant or non-critical data movement can be reduced or approximated to reduce memory space. The model of a frame of speech data can also be achieved using splines. The speech model involves data that is determined by their frequencies. The high peaks of the frequencies become the critical data and should be considered in the approximation.

The approximation of data is related to data compression. In data compression, Chebyshev approximation has a close relation with common interpolation methods such as spline [1]. The Chebyshev approximation of spline is a good solution to approximate the incoming data.

Spline is the most significant method that is applied into the approximation concept and numerical analysis. Splines have good computational properties such as compact representation and computational stability. For example, data collected by sensor readings or non polynomial discrete functions which require curve fitting need to be approximated or interpolated by splines to be analysed for further studies. In general, curve fitting problems occur in signal processing, graphics, statistical analysis and in geometric modeling. Data compression needs the extreme points that can control the approximation function to perform a similar approximate data according to actual data. The reader can refer to papers which examined the capabilities of spline approximation in the representation of speech spectra [2], image interpolation [3], [4], hysterisis loops [5] and scattered data [6].

Splines have the most attractive properties [7] because splines can be divided into several segments which can prevent the function from oscillating too far from the reference axis. Another important consideration is that such representations are more numerically stable and usually computational. Nevertheless, using splines as the method in Chebyshev approximation will affect the non-linear approximation system [1]. The approximation will be more complicated and complex. Therefore, it is important to analyse the characteristics of approximation functions in order to have the best approximation of data.

Interpolation and approximation are two different concepts of analysis when analysing incoming data. The interpolation function should pass the exact values of the discrete function. Diversely, approximation conveys a simpler function with less number of data to express the entire set of real data. Thus, the approximation creates error that differentiates the correct data from the approximation data. Error has to be measured for the determination of appropriate approximation [8].

2 REVIEW OF PREVIOUS WORKS

Curve fitting can be used on many applications to refer to the behaviour of data [9], [6], [10]. The effectiveness of the approximation function is determined by the minimum number of knots which describes the behavior of data and present less error between real data and approximation data. These knots are the extreme points which can control the approximation function to perform a similar approximate data according to actual data.

Many different data fitting algorithms have been developed to satisfy the criteria needs for an approximation function. Haruki and Horiuchi [4] applied data fitting by the spline function which uses the least square approximation.

Then, computing quantities and errors are examined and the effectiveness and the potential of their approaches are described. This includes the methods that sequentially removes knots and is known as data reduction. The knot will be removed eventually until the approximate function can tolerate between the number of knots presenting the data and error generated.

Kitson [11] evaluated a new technique used in geometric modeling with classical signal processing perspective. The volume of data for high accuracy linear approximate is overwhelming and needs an efficient interpolation/reduction algorithm. His approach introduced three steps of data reduction strategy including Rank, Remove and Approximate. Rank is used to determine an order of knot removal where Remove determines the number of knots to actually remove. Approximate then computes an approximation that is usually calculated in least square sense.

Competing algorithms start with essentially no knots and build up the approximating function by adding a knot. Park [1] modified the equation used by Parks arid McClellan [12] on designing digital filter using Chebyshev approximation. The algorithms using polynomial approximation and Chebyshev of free knot polynomial spline is being developed and the comparison of using both methods is demonstrated using FIR filter implementation.

This paper will focus on the initialisation of several number of knots and the algorithm will iterate until the best set of knots is acquired which represents a set of data known as extreme points. The error generated will be analysed to determine how good the algorithms approximated the real data. The discussion is extended by showing the relation between the total numbers of extreme points and total iteration that is needed for convergence.

3 BACKGROUND THEORY

3.1 Chebyshev Approximation

Assuming the incoming data is in discrete domain and given $f[n]$ is defined as real data, $g[A,n]$ is the approximate data and $\phi[n]$ are basis function defined on $n = [b_s, b_e]$. $f[n]$ can be approximated as

$$\sum_{i=1}^{k} g[A,n_i] = a_1\phi_1[n_i] + \ldots + a_k\phi_k[n_i] = \sum a_i\phi_i[n] \tag{1}$$

Commonly, the values of the function $f[n]$ are not equal to the values of the corresponding $g[A,n]$ for every n ε $[b_s, b_e]$, if not $g[A,n]$ is the identical function $f[n]$ [11]. According to the aproximation function and real data, the Chebyshev error norm, is estimated as

$$\delta = \max\left|f[n] - g[A,n]\right| \tag{2}$$

Adapting the Chebyshev approximation method and the alternation theorem a set of extreme points can be determined. The error norm is maximum at a set of extreme points. Therefore, the error is equal at any extreme points. The algorithms begin by introducing initial discrete extreme points and try to get the best approximation for those points. The algorithms iterate until all the parameters of the approximation function converge. From the best approximation, an error function between the original points and approximations can be obtained. Adapting the Chebyshev approximation, the Chebyshev criteria is satisfied; the mathematical descriptions of the best solution in approximation are as follows;

1. There are k+1 alternation points (at points when slope is equal to zero)
   $b_s = n_1 < n_2 < \ldots < n_{k+1} = b_e$

2. For any n ε $[b_s, b_e]$, $|f[n] - g[A,n]| \leq \delta$

3. $|f[n_i] - g[A,n_i]|$ = for $i = 1,2, \ldots, k+1$

4. $\Delta f[n] = \Delta g[A,n]$ only for $n = n_2, n_3, \ldots, n_k$

where n_i's are the alternation points and is the estimated optimal error corresponding to the set of n_i's.

From equation (2) and (3) Park [1] has shown that the matrix Equation related to the basis function, the errors and the coefficient of Chebyshev approximation in discrete domain as below:

$$\begin{pmatrix} \phi_1|n_1| & \phi_2|n_1| & \phi_3|n_1| & \cdots & \phi_k|n_1| & -1 \\ \phi_1|n_2| & \phi_2|n_2| & \phi_3|n_2| & \cdots & \phi_k|n_1| & 1 \\ \vdots & \vdots & \vdots & \ddots & \vdots & -1 \\ \phi_1|n_3| & \phi_2|n_3| & \phi_3|n_3| & \cdots & \phi_k|n_{k+1}| & |-1|^{k+1} \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ \delta \end{pmatrix} = \begin{pmatrix} f|n_1| \\ f|n_2| \\ \vdots \\ f[n_{n+1}] \end{pmatrix}$$

The algorithm begins by initialising the set of n_i's. Then the parameters δ and $(a_1, a_2, \cdots, a_k)$ will be estimated using the above equation. Using $a_1, a_2, \ldots, a_k$, the approximation function $g[A, n]$ will be generated and a new set of ni's will be obtained by checking new alternation of error function. The iteration continue until the set of n_i's is converged. The algorithm for the above equation becomes more complicated if the non-linear method is applied. The error curve interpolates n+2 points where $|E(f)| = |\delta|$ and it may create n+3, n+4, or n+5 extreme points [12]. If the error curve is greater that n+2 extreme points, the set of extreme points can be chosen by the search procedure.

3.2 Lagrange Interpolation

Lagrange interpolation involves finding a polynomial of order n that passes through the n+l points. Lagrangian interpolating polynomial is given by;

$$f_n(x) = \sum_{i=0}^{n} L_i(x) f(x_i) \tag{4}$$

where n in $f_n(x)$ stands for the n^{th} order polynomial that approximates the function $y = f(x)$ given at $(n+1)$ data points as (x_0, y0, x1, y1, . . . , xn-1, yn-1, xn, yn and

$$L_i(x) = \prod_{j=0,\, j\neq i}^{n} x - x_j / x_i - x_j \tag{5}$$

$L_i(x)$ is a weighting function that includes a product of $(n\text{-}1)$ terms with terms of $j = i$ omitted.

3.3 Splines

A spline $s(x)$ of order n defined over an interval $[x_{min}, x_{max}]$ is composed of sections of polynomial curves $p_k(x)$ of degree n- 1 joined together at k fixed points in the interval. Splines are good for representing a smooth curve $y = s(x)$ or data generated from a smooth curve over a fixed interval. Splines are extremely flexible and can be used to approximate any smooth curve to a given accuracy by choosing sucient number of knots or a high enough order (degree) [13]. Thus, splines can be used to represent very large sets of data. In the splines approximation, the order of the splines is chosen for further computation. For a wide set of knots, the fourth order spline known as cubic spline is the most common choice [14]. The equation defining a cubic spline can be obtained on each piecewise interval. Each $P_i(x)$ is in the form of

$$P_i(x) = c_{0,i} + c_{1,i}(x\text{-}x_i) + c_{2,i}(_{x\text{-}xi})^2 + c_{3,i}(x\text{-}x_i)^3 \tag{6}$$

where c's are the coecients of the cubic splines on each interval and x_i is the values of x between two intervals. Since spline present lower order, it has much greater flexibility in fixing the number and location of the interior. More knots are needed in regions where the curve underlying the data is rapidly changing, fewer knots where the curve is relatively smooth. Spline is expected to be more flexible and able to follow the pattern of the data more closely.

3.4 Determining The Extreme Points

The extreme points play an important role to determine the maximum values. The extreme points are determined by finding those points when the function changes sign. The start and end points are always assigned as extreme points. In a continuous domain, the function changes sign when the derivative of the function is equal to zero. The second derivative needs to be less or equal than zero because there is a possibility that the points is an inflection point. However, in discrete system there is no derivative concept to determine extremes at the alternation points. Therefore, the modification of finding an extreme will be changed [1]. Forward difference and backward difference have been used in discrete domain to achieve the same results of extreme points. In discrete domain, there are possibilities that the points are not co-located because the extreme points may not be selected during sampling. However, there are possibilities that the differences are zeros. When the first differences is zero, the extreme points are checked whether they are the inflection points or not.

$$\Delta f\left[n_{e_i}\right]\nabla f\left[n_{e_i}\right] \le 0 \tag{7}$$

when $\nabla f\left[n_{e_i}\right]=0$, then $\Delta f\left[n_{e_{i-1}}\right]\nabla f\left[n_{e_{i-1}}\right]\le 0$

and when $\Delta f\left[n_{e_i}\right]=0$, then $\Delta f\left[n_{e_{i+1}}\right]\nabla f\left[n_{e_{i+1}}\right]\le 0$

Discreteness also affects the possible choice of extreme points. It eliminates the possibility of finding better extreme points because it is limited to values on the sampling grid. In discrete system, the values between the grids are ignored, thus creating bounding errors that effect the approximation as a valid solution.

4 ALGORITHM

This paper developes new algorithms by applying the simple Chebyshev error norm using Equation (2). Hence, the matrix derived by Park [1] in Equation (3) is not used. For several numbers of extreme points, the Equation (3) approaches ill-condition and becomes an unbalance matrix [1]. Using this new method, the iteration is simpler but there is no guidance on the total number of extreme points.

This paper proposes the algorithm that consists of two parts. The first part is to estimate the difference between actual data and approximate data. The method that is used in this first part will minimise the difference of data. The second part is to find an approximation function based on the difference between actual data and approximate data. This part will estimate the location of the extreme points according to the error generated between the actual data and approximate data. The methods that are used in this second part are very important to get the best approximation of data. Using Lagrange polynomial as the approximation method will cause the set of the extreme points to be unique. This means that the actual data can be represented by a set of the extreme points. The error generated also has linear characteristics.

Since spline has non-linear characteristics, the set of extreme points are not unique. This means that the actual data can be represented by several approximation functions. However, each of the approximation function using different sets of the extreme points will generate different values of errors. The restriction of the algorithm is the approximation function of the extreme points and are only defined at given grids sampling.

A wide area of data is defined on n ε $[b_s, b_e]$. Figure 1 shows that the Algorithm-1 starts by initialising a set of extreme points for i = 2, 3, …, k where the first and the last extreme points are fixed at and. Then, the set of extreme points are approximated by using cubic splines. From equation (2), an estimated error δ is obtained. Since the

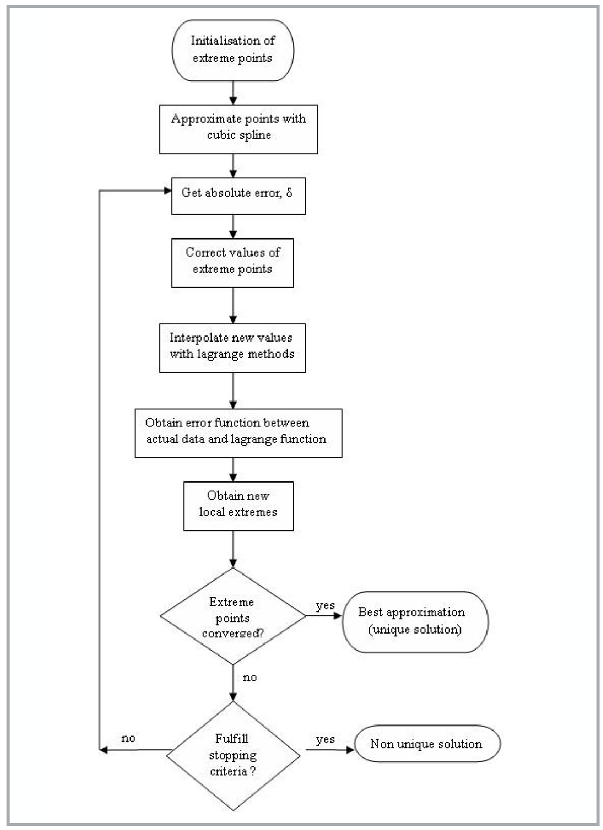

Figure 1: Flow chart of Algorithm-1

algorithm needs to follow Chebyshev characteristics, the values of approximated data at the set of extreme points are corrected by adding and subtracting δ repeatedly. Then, the new values of the extreme points are approximated using Lagrange interpolation. From the approximate values of the Lagrange polynomial, the error function $e[n]$ between approximate and real data is obtained. The error function will follow the distribution of Chebyshev error norm. From the error function $e[n]$, a new set of extreme points can be obtained with the rule in the section discussed before. The algorithm is repeated until the set of extreme points converged. If the algorithm does not converge, the algorithm is modified by adding a stopping criteria. The stopping criterion stops the algorithm if the approximation keep iterating at the same set of extreme points.

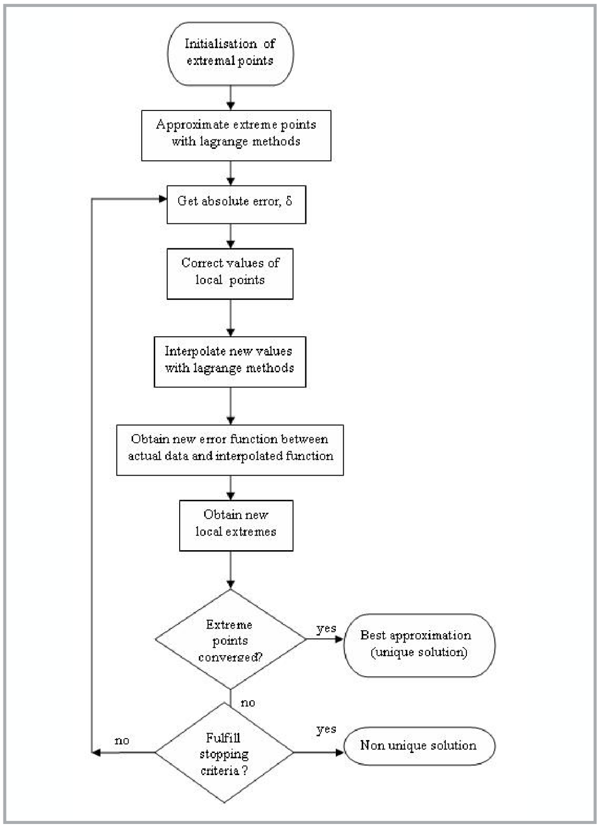

Figure 2: Flow chart of Algorithm-2

The algorithm is modified by choosing a combination of cubic splines and Lagrange polynomial to generate the new algorithms of the approximation data. Figure 2 shows that the Algorithm-2 uses Lagrange polynomial to estimate the absolute error d and interpolate the error function. Figure 3 shows that the Algorithm-3 uses Lagrange polynomial to estimate absolute error δ and cubic spline to interpolate the error function. Figure 4 shows that the Agorithm-4 uses cubic spline to estimate absolute error and interpolate the error function. The best approximation of real data can be achieved by using suitable combi-nation methods of estimated error d and estimated the set of extreme points. Different algorithms will result in difference sets of extreme points and difference in behavior of error function. The algorithms are implemented using MATLAB v 6.1.

5 RESULT AND DISCUSSION

A non polynomial function $f\ [n] = sin\ (2\pi(n\text{-}50)/100$ is defined at n=[0, 100]. This function applies to generate a group of specific data. However, it is also can be applied to generate data of any type mathematical functions such as trigonometric and hyperbolic function. Beside all this, it has certain limitation that must be considered namely, random data. The algorithm applies the same function and the same number of total initial extreme points with the algorithm that was developed by Park [11]. Thus, the results including the set of extreme points and error generated can be compared for further discussions. The itera-tion starts with a set of extreme points {0, 10, 20, 30, 40, 100}. The algorithms fix the number of extreme points. The first and last of the extreme points are the starting and ending points of the data. For Lagrange interpolation, the error norm is

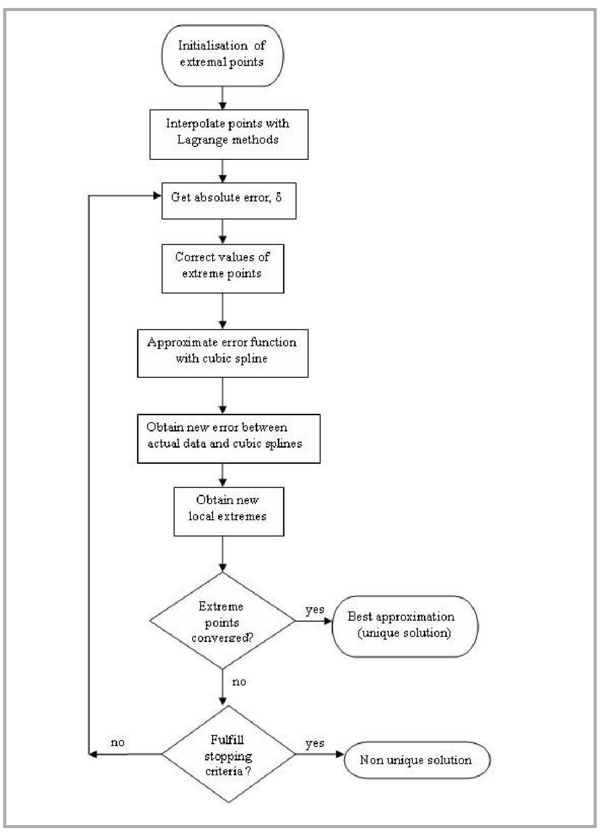

Figure 3: Flow chart of Algorithm-3

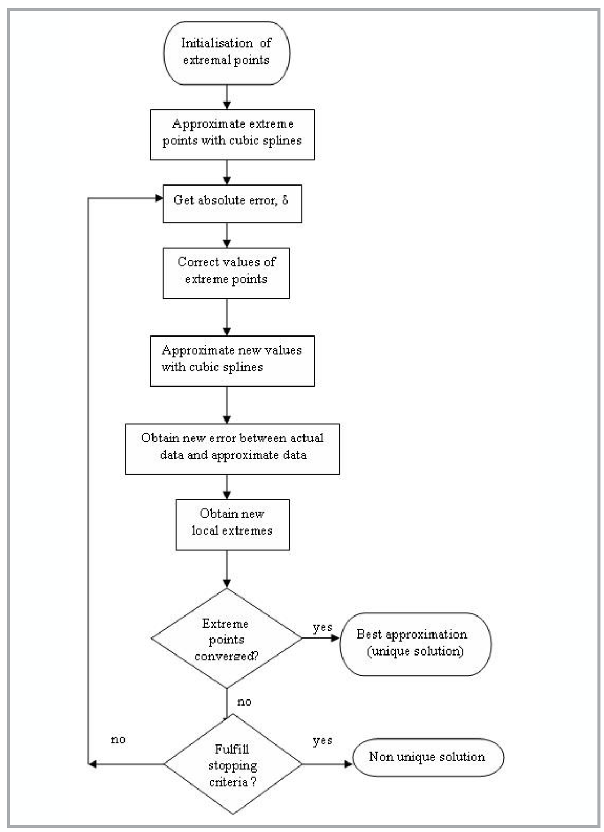

Figure 4: Flow chart of Algorithm-4

Table 1: Results for Algorithm-1

No of iteration	Extreme points	Absolute error
1	0 10 20 30 40 100	0.3956
2	0 5 19 34 85 100	0.2970
3	0 7 23 61 93 100	0.1016
4	0 9 35 67 91 100	0.0410
4	0 11 36 64 89 100	0.0379
5	0 12 36 64 88 100	0.0347

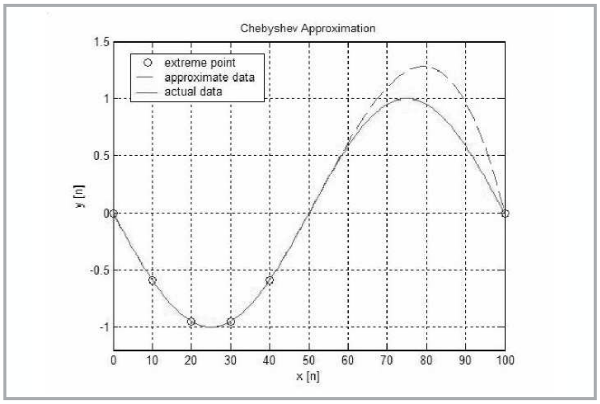

Figure 5: First iteration of Algorithm-1

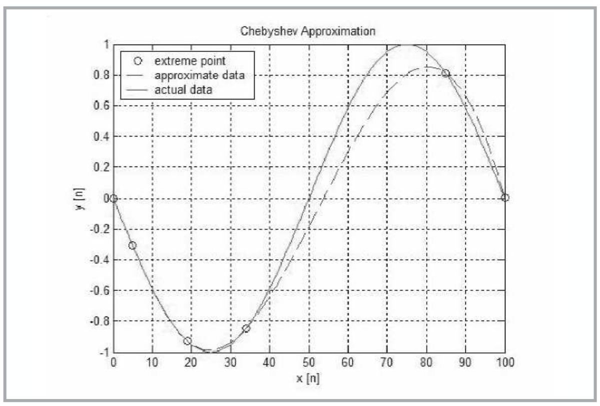

Figure 6: Second iteration of Algorithm-1

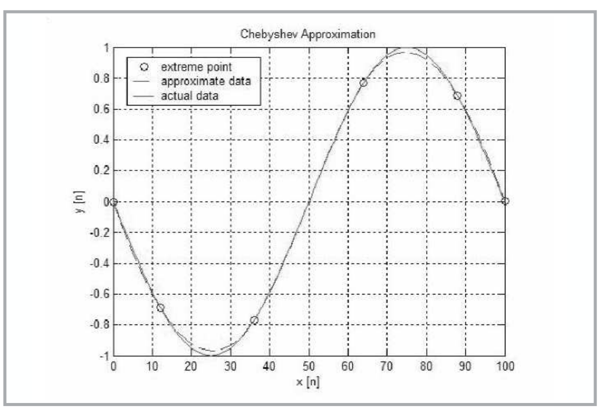

Figure 7: After sixth iteration of Algorithm-1

presented in least squared sense. In Algorithm-1 the extreme points are successfully converged at the 6th iteration at {0, 12, 36, 64, 88, 100}. The overall results are shown in Table 1. It means that the algorithms generate the best approximation solution. The maximum error δ is 0.0347. Compared to algorithms generated by Park [1] this algorithm provides less iteration hence having smaller error norms. The iterative results of Chebyshev approximation are shown in Figures 5, 6 and 7. The maximum error δ is decreased during the iteration thus, the differences between the actual data and the approximate data are closer. The differences can be compared from the first iteration and second iteration in Figures 5 and 6. Figure 8 shows that the generated error satisfies the Chebyshev criteria. Table 2 shows that for the Algorithm-2, from the 5th iteration until infinity the set of extreme points keep iterating at two different sets of extreme points {0, 15, 38, 62, 85, 100} and {0, 14, 37, 63, 86, 100}. The values of Chebyshev error norm for these points are 0.0177 and 0.0159. The approximate data at the 5th

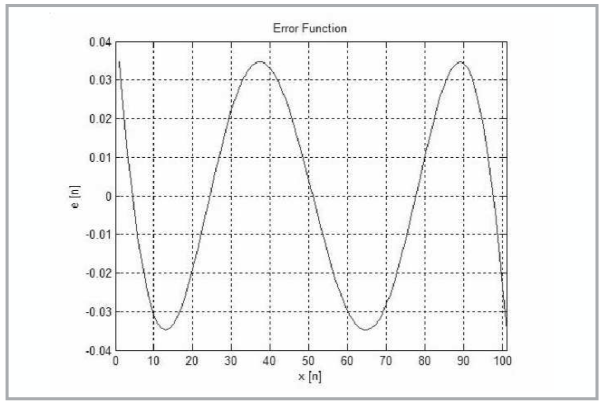

Figure 8: Error function after convergence

Table 2: Results for Algorithm-2

No of iteration	Extreme points	Absolute error
1	0 10 20 30 40 100	0.5599
2	0 5 19 34 85 100	0.0931
3	0 7 23 61 93 100	0.0224
4	0 10 36 66 89 100	0.0135
5	0 15 38 62 85 100	0.0177
6	0 14 37 63 86 100	0.0159
7	0 15 38 62 85 100	0.0177
8	0 14 37 63 86 100	0.0159
9	0 15 38 62 85 100	0.0177

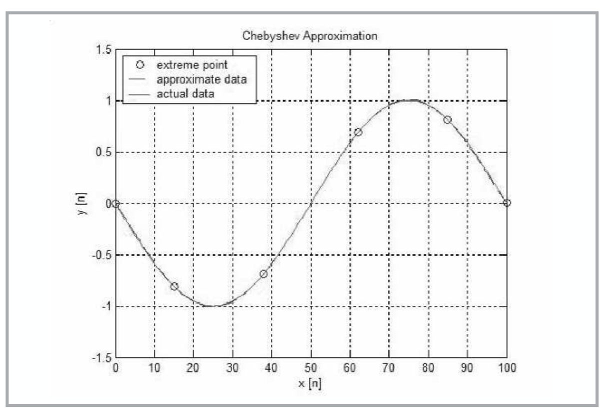

Figure 9: Fifth iteration of algorithm-2

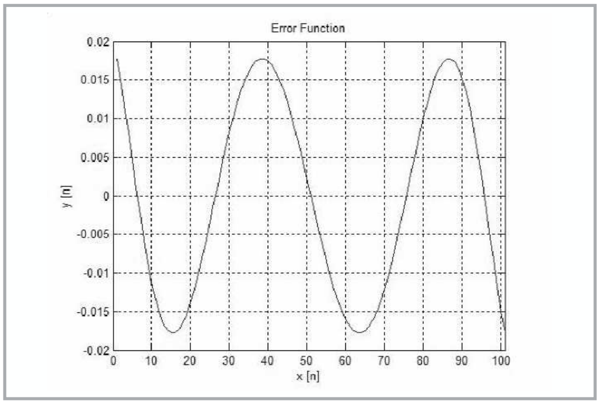

Figure 10: Error function after convergence

Table 3: Results for Algorithm-3

No of iteration	Extreme points	Absolute error
1	0 10 20 30 40 100	0.5599
2	0 7 20 32 78 100	0.0766
3	0 7 21 54 89 100	0.0237
4	0 9 36 69 94 100	0.0171
5	0 6 25 57 93 100	0.0184
6	0 9 40 72 95 100	0.0159
7	0 6 27 58 92 100	0.0155
8	0 9 42 73 95 100	0.0167
9	0 6 28 59 91 100	0.0143
10	0 8 43 73 94 100	0.0152
11	0 5 27 59 91100	.0170
12	0 9 41 73 94 100	0.0160
13	0 6 27 58 91100	0.0157
14	0 9 42 73 94 100	0.0157
15	0 6 27 58 91100	0.0157
16	0 9 42 73 94 100	0.0157

iteration is closer to the actual data because the maximum error is smaller than at the 6th iteration. This is partly due to the constraint of discrete intervals while choosing the new extreme points. Thus, the iteration is being modified because both of the extreme points are similar. Figure 9 shows the result of the 5th iteration which is similar to that of the 6th iteration. Figure 10 shows the error function after convergence at 5th iteration which is similar to the results after convergence at 6th iteration.

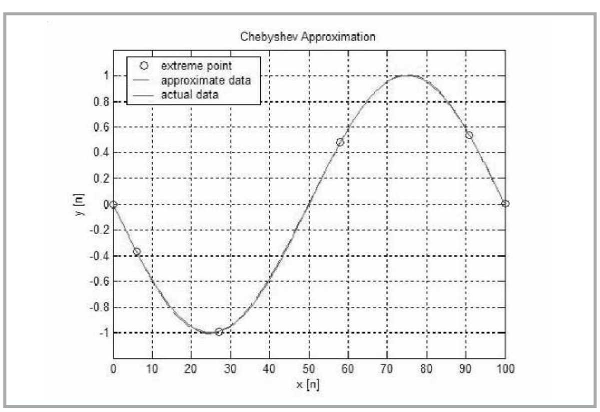

Figure 11: 13th iteration of algorithm-3

The results of Algorithm-3 that is shown in Table 3 shows that from the 13th iteration until infinity the set of extreme points keep iterating at two different sets of extreme points {0, 6, 27, 58, 91, 100} and {0, 9, 42, 73, 94, 100}. Figures 11 and 12 show the 13th and the 14th iteration of Chebyshev approximation of Algorithm-3. The values of maximum error for these points are 0.0157. The generated errors of the algorithm are shown in Figures 13 and 14. This is partly due to the constraint of discrete intervals while choosing the new extreme points. The extreme points at the 13th and 14th look unfamiliar to each other and have many differences. The

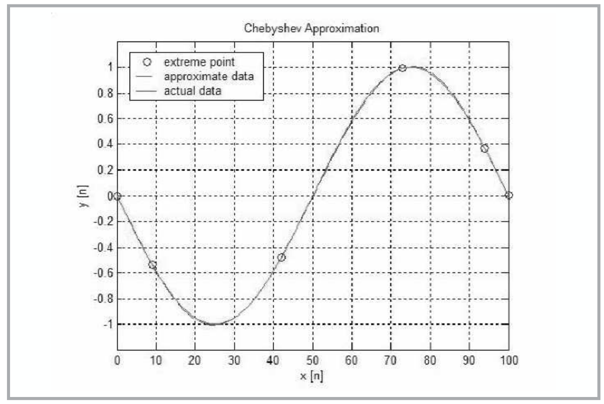

Figure 12: 14th iteration of Algorithm-3

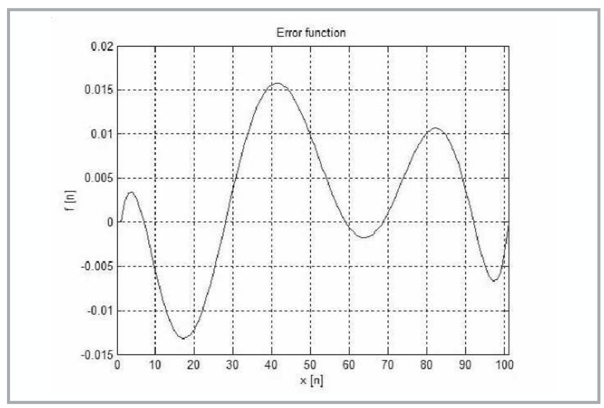

Figure 13: Error function after 13th iteration

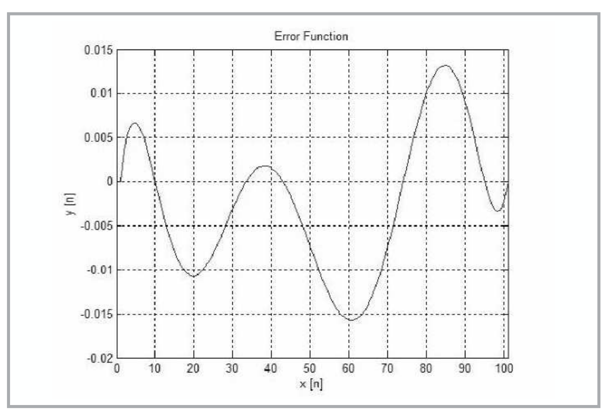

Figure 14: Error function after 14th iteration

algorithm is being modified and is stopped if the algorithms keep iterating at the same points. Although both sets having the same absolute error, the Chebyshev approximation results are totally different. The error function is interpolated by splines function. As splines having non-linear characteristics, algorithms generate non unique solution. However, compared to Algorithm-1 and Algorithm-2, the maximum error norm obtained by Algorithm-3 is smaller.

In Algorithm-4, cubic spline is used to estimate δ and approximate error function. Table 4 shows that from the 7th iteration until infinity the set of extreme points keep iterating at three different sets of extreme points {0, 9, 33, 67, 91, 100},

Table 4: Results for Algorithm-4

No of iteration	Extreme points	Absolute error
1	0 10 20 30 40 100	0.3956
2	0 7 21 49 87 100	0. 1924
3	0 8 28 59 89 100	0.1899
4	0 10 35 66 90 100	0.0866
5	0 8 30 70 92 100	0.0368
6	0 7 37 63 93 100	0.0174
7	0 9 33 67 91100	0.0579
8	0 8 28 72 92 100	0.0285
9	0 7 38 62 93 100	0.0105
10	0 9 33 67 91 100	0.0647
11	0 8 28 72 92 100	0.0285

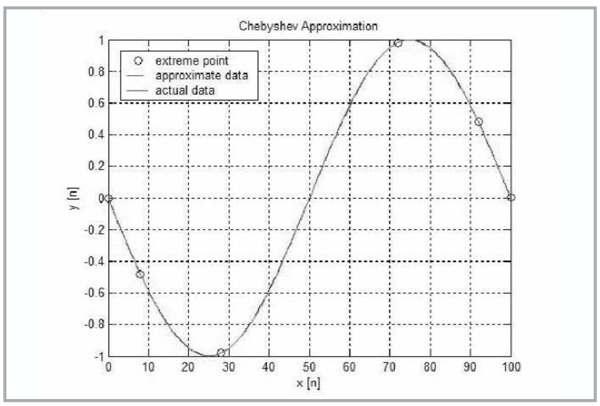

Figure 15: 8th iteration of Algorithm-3

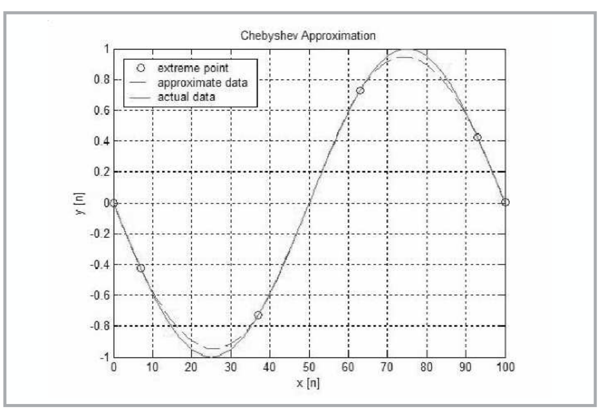

Figure 16: 9th iteration of Algorithm-3

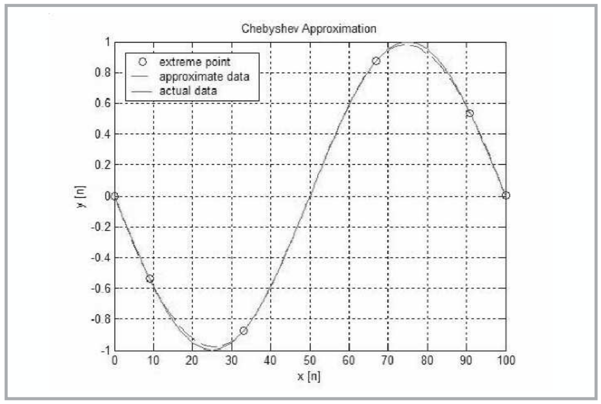

Figure 17: 10th iteration of Algorithm-3

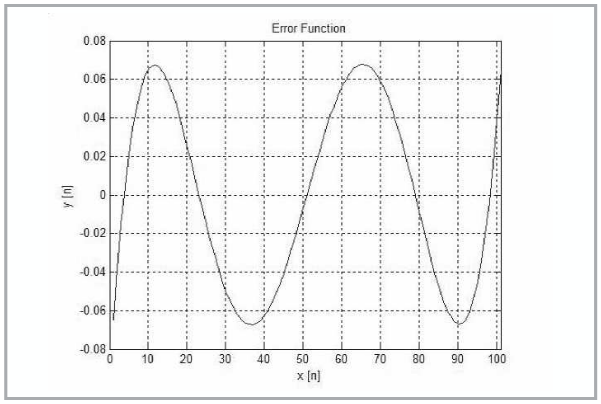

Figure 18: Error function after 8th iteration

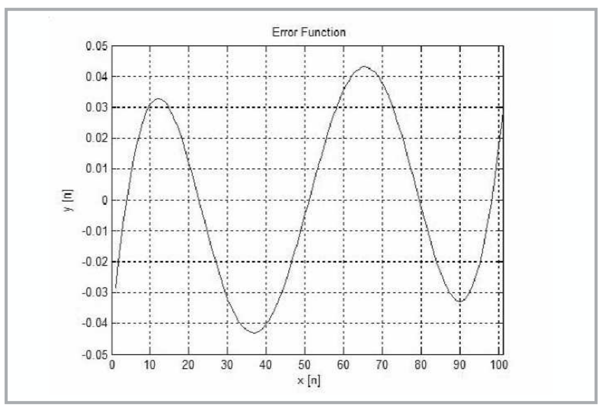

Figure 19: Error function after 9th iteration

{0, 8, 28, 72, 92, 100} and {0, 7, 38, 62, 93, 100} resulting in a maximum error at 0.0647, 0.0285 and 0.0105. Figures 15, 16 and 17 show that the Chebyshev approximation results after the 8th, 9th and 10th iteration. The algorithm can be modified by adding the stopping procedure. From Figure 18 shows that error norm of a set of extreme points {0, 9, 33, 67, 91, 100} does satisfy the Chebyshev criteria. However, the other error norm in Figures 19 and 20 does not satify the Chebyshev criteria and the algorithm creates non unique solution. Thus, the set of extreme points {0, 9, 33, 67, 91, 100} can be implemented for practical application.

Using the cubic spline to approximate the data causes the estimated absolute error δ to be smaller rather than using

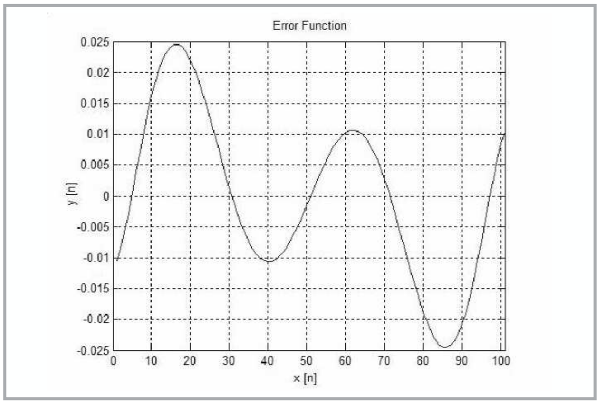

Figure 20: Error function after 10th iteration

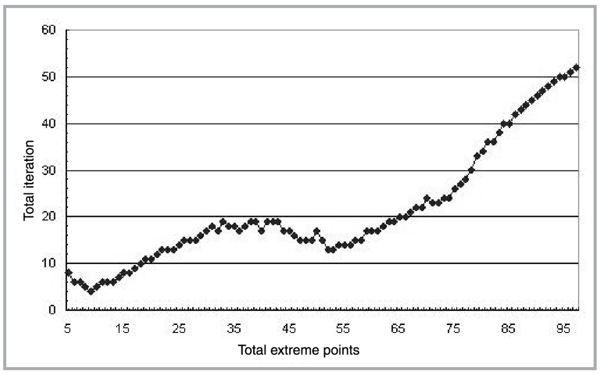

Figure 21: Total iteration versus total of extreme points for Algorithm-1

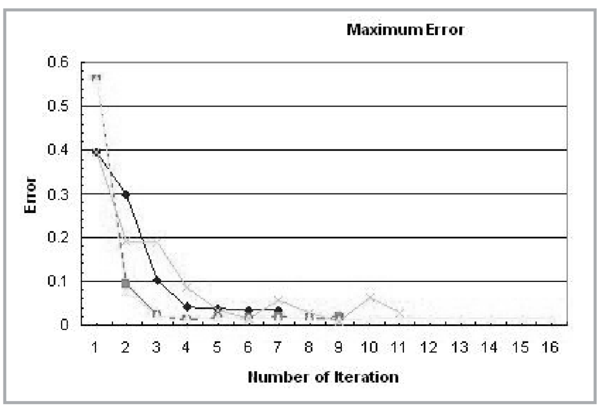

Figure 22: Maximum error generated by each algorithm

Lagrange polynomial. However, using cubic spline in determining the set of extreme points causes the approximation to be non-unique solutions. The approximate data can still be achieved but it creates several approximation functions using different sets of extreme points and error generated.

The initial extreme points do not affect the result of the algorithms to get the same set of extreme points. However the approach to get better extreme points will reduce the number of sequence the algorithm needs to converge. Figure 21 shows the number of iteration of the algorithms can be reduced as maximum error norm if total extreme points are increased. However, at certain number of extreme points, the number of iteration will rise if total extreme points are increased. Thus, application of Chebyshev approximation should compromise between total extreme points and the error generated. The relation between the absolute error and number of iteration for each algorithm is shown in Figure 22.

In signal processing, the total number of extreme points are related to the amount of memory space while the total iterations are correlated to the computation time. Since the total iteration and the computation time are increased, the algorithm is not quite applicable for real time system. If the Chebyshev norm error becomes the critical part of the application, the system needs to tolerate with computation time. If the computation time and memory space are not the constraints of the system, total number of extreme points can be increased. Thus, the Chebyshev error norm will be reduced.

These algorithms have no guidance on the number of total extreme points. The example used in the algorithms needs at least five extreme points, otherwise it will not converge. This means that the behavior of data cannot be represented by a smaller number of extreme points. Thus, the use of the algorithms may be restricted on random data because it needs a greater number of extreme points. Data reduction procedure has better performance to approximate random data. If the algorithms do not converge, it does not mean that the algorithms are cannot be used. The algorithms still can be applied. However, the Chebyshev error norm is not unique, thus generating Chebyshev error norm as correction factor may not be practical.

6 CONCLUSION

The algorithms of the spline approximation are presented in this paper. The study obviously showed that the Chebyshev approximation with spline that involves the total number of the set of extreme points plays a major role in determining the best approximation of discrete data. This is useful for data compression that is required for high level signal processing. The best approximation is determined by a set of extreme points that presents less error. The results show that the error generated decreases as the total number of extreme points increase. However, the algorithm that uses more number of extreme points will cause an increment of the total number of iteration. From the Chebyshev approximation of spline, the most important results are a set of extreme points that controls the approximation function of data and the generated errors. If the error function satisfies the Chebyshev criteria, the error function can be used as a correction factor to correct the approximate data, and the actual data can be achieved using much less number of data which are extreme points and the absolute error 5. In terms of engineering application, this means that besides having the entire data, the data compression that uses Chebyshev approximation for application such as robotic motion and speech model can be extracted using a set of extreme points. The generated error can also produce similar quality data.

ACKNOWLEDGEMENTS

The authors would like to take this opportunity acknowledge Skim Latihan Akademik Pengajar (SLAB), Jabatan Perkhidmatan Awam (JPA) as the spon-sor of author's study and Kolej Universiti Teknologi Thn Hussein Onn (KUiTTHO) for allowing the use of the facilities for this research. ■

REFERENCES

[1] J.H. Park, *Chebyshev Approximation of Discrete Polynomials and Splines*, Unpublished dissertation, 1999.

[2] CD. Covington, *Cubic Spline Modeling of Speech Spectra*, IEEE Inter-national Conference on Acoustics, Speech and Signal Processing, Vol. 10, pp 1125 - 1128, April 1985.

[3] H.S. Hou, and H.C. Andrew, *Cubic Splines for Image Interpolation and Filtering*, IEEE Transactions on Acoustics, Speech and Signal Processing, Vol. 26, No.6, pp 508 - 517, December 1978.

[4] R. Haruki, and T. Horiuchi, *Data Fitting by Spline Functions Using the Biorthornormal Basis of the B-spline Basis*, IEEE International Confer-ence on Pattern Recognition, Vol. 3, pp 270 - 273, September 2000.

[5] D. L. Portigal, *A Magnetic Recording Simulation Program Having an Improved Fit to Actual Hysteresis Loops*, IEEE Transactions on Magnetics, Vol. 11, No. 3 pp 934 - 941, May 1975.

[6] O. Kreylos, and B. Hamann, *On Simulated Annealing and the Construc-tion of Linear Spline Approximations for Scattered Data*, IEEE Trans-actions on Visualization and Computer Graphics. Vol. 7, No. 1, pp 17 -31, March 2001.

[7] L. Crousel, and J.J. Neirynick, *Polynomial Chebyshev Application of The Ideal Filter*, IEEE Transaction on Circuit and System, Vol. 15 No. 4, pp 307-315, December 1968.

[8] F. Robert, *Handbook of Applicable Mathematics Volume III Numerical Methods*, Churchhouse, John Wiley and Sons, 1992.

[9] M. Mizuta, *Algebraic Curve Fitting for Multidimensional Data with Ex-act Squares Distance*, IEEE International Conference on Systems, Man and Cybernatics, Vol. 1, pp 516 - 521, October 1996

[10] K.C.P. Wong, H.M. Ryan, J. Tindle, J. Blackett, M. W. Watts, *Digital Measurement of Lightning Impulse Parameters using Curve Fitting Algorithms*, High Voltage Engineering Symposium, Vol.1, pp 270 - 273, August 1999.

[11] F. L. Kitson, *An algorithm for Curve and Surface Fitting using B-spline*, IEEE International Conference on Acoustics, Speech and Signal Process-ing, Vol. 1, No.89, pp 1207 - 1210, May 1989.

[12] T.W. Parks, and J.H. McClellan, *Chebyshev Approximation for Nonre-cursive Digital Filters with Linear Phase*, IEEE Transaction on Circuit Theory, Vol.19, No. 2, pp 189 - 194, March 1972.

[13] M. C. Cox, A.B. Forbes, P.M. Harris, *Software Support for Metrology Best Practice Guide No. 4 Discrete Modelling*, Unpublished source, 2000.

[14] S.A. Dyer and J.S. Dyer, *Cubic-spline Interpolation: Part 1*, IEEE In-strumentation and Measurement Magazine, Vol. 4, No. 1, pp 44-46, March 2001.