B-SPLINE CURVE FITTING WITH DIFFERENT PARAMETERIZATION METHODS

BY

KHENG JIA SHEN

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfillment of the requirements

for the degree of

BACHELOR OF COMPUTER SCIENCE (HONS)

Faculty of Information and Communication Technology

(Kampar Campus)

JAN 2020

UNIVERSITI TUNKU ABDUL RAHMAN

REPORT STATUS DECLARATION FORM

Title: B-spline Curve Fitting with Different Parameterization Methods

Academic Session: JAN 2020

I KHENG JIA SHEN

(CAPITAL LETTER)

declare that I allow this Final Year Project Report to be kept in

Universiti Tunku Abdul Rahman Library subject to the regulations as follows:

1. The dissertation is a property of the Library.
2. The Library is allowed to make copies of this dissertation for academic purposes.

Verified by,

(Author's signature) (Supervisor's signature)

Address:

PT 1520 JLN SBJ PUTRA, 10

KG PASIR PEKAN,

16250 WAKAF BHARU, KELANTAN

Ts. Dr. Lim Seng Poh

Supervisor's name

Date: 23/04/2020 **Date**: 23/04/2020

B-SPLINE CURVE FITTING WITH DIFFERENT PARAMETERIZATION METHODS

BY

KHENG JIA SHEN

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfillment of the requirements

for the degree of

BACHELOR OF COMPUTER SCIENCE (HONS)

Faculty of Information and Communication Technology

(Kampar Campus)

JAN 2020

DECLARATION OF ORIGINALITY

I declare that this report entitled "**B-SPLINE CURVE FITTING WITH DIFFERENT PARAMETERIZATION METHODS**" is my own work except as cited in the references. The report has not been accepted for any degree and is not being submitted concurrently in candidature for any degree or other award.

Signature : ____________________________

Name : Kheng Jia Shen

Date : 23/04/2020

ACKNOWLEDGEMENTS

I would like to express my sincere thanks and appreciation to my supervisor, Dr. Lim Seng Poh who has given me this bright opportunity to engage in this research. Dr Lim has given me unconditional support and guidance throughout the research. A million thanks to you.

Finally, I must say thanks to my parents and my family for their love, support and continuous encouragement throughout the course.

ABSTRACT

B-spline curve is important in the geometric modelling field and Computer Aided Design (CAD) in the visualization and curve modelling. B-spline is considered as one of the approximation curves as it is flexible and could provide a better behaviour and local control. The shape of the curve is influenced by the control points. Parameterization of the curve or surface is important in computer graphics as it can improve the overall quality of the visualization. Various parameterization methods such as uniform, centripetal, chord length and exponential had been used for the B-spline data fitting. However, there is an issue in many fields which is to construct an optimal curve with the given data points. Therefore, a comparison is made between the parameterization methods in this research in order to determine the optimal method for the B-spline curve fitting. This research is only focused on B-spline curve and four parameterization methods. In addition, uniformly spaced and averaging knot vector generations are used in generating the knot vector. After generating control points, distance between the generated and original data points is used to identify the error of the algorithm. Later, genetic algorithm and differential evolution optimization are used to optimise the error of the curve. Based on the result produced, each of the parameterization generated varied curve shape due to the different properties of the datasets.

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

CAD	Computer Aided Design
RE	Reverse Engineering
NURBS	Non-Uniform Rational B-spline
PC-Curve	Parametric Cubic Curve
OGH	Optimized Geometric Hermite
PIA	Progressive Iterative Approximation
AIC	Akaike Information Criteria
IDE	Integrated Development Environment
GA	Genetic Algorithm
DE	Differential Evolution

CHAPTER 1 INTRODUCTION

1.1 Overview

Geometric Modelling is the mathematical approach used for modelling the object in computer graphics. Reverse Engineering is able to reconstruct the model based on the data points which is part of the real object. In this research, both interpolation and approximation of B-spline curve are used for the curve fitting and study about the differences. Parameterization is an important part of the curve fitting as it will determine whether the B-spline curve is in a good shape. However, the parameters obtained could not handle all types of datasets. Therefore, optimization is needed to optimise the accuracy and minimize the distance between the data points and fitting the curve.

In this chapter, the problem domain and motivation of this research are discussed to ensure that the research will have improvement in solving the problem. Besides, the objectives and the scope of the research highlight the purpose and focus of the methods and a general flow of the research methodology. This chapter also gives an idea about the impact, significance and contribution about the research. Lastly, the report organization of this research is stated.

1.2 Research Background

Geometric Modelling discusses about the mathematical approach used to model objects for the purpose of computer graphics and computer aided design (CAD) in 3D modelling. This concept is useful in providing understanding in CAD and computing the model for evaluation. The modelling concepts make use of the geometric object that has part of information and structure which represent the real object. The geometric object is constructed using curve and surface as they are easy to create the shape or structure of real object and preserve the complex structure of object.

Reverse Engineering (RE) is an approach used to reconstruct a computer model from a group of point data that are obtained from a part of the object's surface. RE is important for some situation. First situation is when the model is created using CAD. Second situation is when there are frequent changes in design model and the initial model is outdated. The third situation is when there is lack of engineering drawing to construct the essential part of product (Chouychai, 2015). The RE process normally starts from a set of data point acquired from a scanner device and later locally modelled the points to be either interpolated or approximated with the use of surface patches (Haron, et al., 2012). A device such as coordinate measuring machine or laser beam is used to capture the point cloud on the structure of the physical object. The point cloud obtained is unstructured and later converted into triangular surface mesh. After that, the points are used for interpolation or approximation which are the methods for curve or surface fitting (Marinić-Kragić, et al., 2018).

Curve is an essential object used in computer graphics to create a 2D model. It is similar to line but it can form different shapes such as circle, parabola, arc and so on. Curve is a group of points that are connected to each other and all the points have 2 neighbours except for the endpoints. It is easier to be stored and able to be created on any resolution and represent the real object easily. The properties of the curve are used to describe the type of curve created and some of the properties required the understanding of the whole curve. Local properties are essential in describing the curve without the understanding of the complete curve. There are 3 ways to represent a curve and surface which are implicit, explicit and parametric. Explicit representation is rarely used in CAD as it is axis dependent and cannot represent multivalued functions effectively. Implicit and parametric representations are commonly used in CAD. However, implicit curve still axis dependent.

Curve fitting is a process or mathematical function to construct a curve based on the point cloud with the existence of the constraints. The curve can be used for data visualization, summarization of the relationships between 2 or more variables, and infer values of function when there are no available data (Ravichandran & Arulchelvan, 2017). There are 2 methods for the curve fitting which are approximation method and interpolation method. Approximation is stated as the evaluation of a function based on simpler function and Interpolation is considered as the evaluation of the function based on the identified values of the function (Lee, et al., 2008). The curve generated using approximation method does not pass through all the data points but pass near to the points whereas the curve produced using the interpolation method passes through all the data points (Ueda, et al., 2018). However, both of the methods stated have respective benefits and problems. For approximation method, the operational time will be longer and not efficient if the process of finding the minimum allowable error and have a large number of data points. This method minimises the impact of noise when constructing the curve and suitable for irregular data points. For interpolation method, the construction of the curve is affected by the noise exists in the data points and the curve produced will not be smooth and consists of distortion. This method is used for those data points which are smooth and accurate (Yang, et al., 2017).

Polynomial curves are a suitable choice for the curve or surface fitting due to their mathematical properties. In most of the industries nowadays, use free-form parametric curves such as Bezier, B-spline and Non-Uniform Rational B-spline (NURBS) as they are flexible and can represent the shape of efficiently by using less parameter to do the computation (Iglesias, et al., 2015). These curves are affected by the parameter of the curves. In this research, the curve will be modelled using B-spline curve. B-spline had become a standard in the industry for data representation because of the advantages in terms of flexibility and control (Kumar, et al., 2003). Local control can be made to the control points of the B-spline curve to change the shape of a section of the curve. The B-spline is one of the effective curve representation because of its' spatial uniqueness, continuity and boundedness, local shape controllability, and property of invariance to affine transformation (Liu, et al., 2014). B-spline curve express blending function value to be nonzero for all the parametric value by just passing near those control point and the sum of the function will always be 1 and the slopes between the curves are constant.

Parameterization of the curve or surface point cloud is important in the CAD application for the curve and surface fitting in order to improve the visual quality of the model. The examples of the applications are texture mapping, typography, remeshing, and curve or surface fitting (Haron, et al., 2012). The method is used to decrease the number of control points that are used to compute the design of the model. There are quite a number of parameterization methods that can be used in B-spline curve and is assumed as an approach of reverse engineering the parameters of the model. A suitable parameterization method used is important in order to generate a good curve fitting.

There are no parameters that could handle various types of data points such as collinear and consecutive points with large distance (Roslan & Yahya, 2016). Manufacturers and industrial corporations usually faced the problems to optimize the processes in order to improve the performance of the products in terms of speed and accuracy. Optimization is to select the best inputs in order to obtain a desired result. Because of the rapid growth of modern technologies, area of optimization had become popular among the development of software, parallel processors and artificial (Chong & Zak, 2001). There are a few popular soft computing methods such as Genetic Algorithm (GA) and Differential Evolution (DE) used to solve various complicated optimization problems. In the construction of B-spline curve, the knots and parameters selection and the acquisition of the control points are the concerns to optimize the curve fitting.

1.3 Problem Statement

B-spline is a free-form parametric curve widely used in the data representation because of the flexibility of the curve. A smooth shape of the curve can be constructed by using just a few parameters and this will save in terms of storage and memory. The results generated will be greatly affected by the parameterization method. However, obtaining a suitable parameterization and the representation of curve are difficult as the curve is parametric and it can lead to continuous, multimodal, multivariate, overdetermined non-linear optimization problem (Gálvez & Iglesias, 2013). The control points for all the datasets are used to generate the B-spline curve. However, not all the parameterization methods are suitable for each of the datasets as the calculation for the parameter and distance of the points is different. Most of the previous work are focused on uniform, chord length and centripetal parameterization method. These problems can lead to the undesirable result of the curve or surface generated. The accuracy of which the distance between the discrete data points and the B-spline curve generated could not be guaranteed (Hasegawa, et al., 2013).

Therefore, based on the problem stated above, a comparison between four parameterization methods for the B-spline curve fitting is made. Different characteristic of the curve will have different parameter values and different shape of the curve will be generated. Genetic algorithm (GA) and differential evolution (DE) is used to solve the optimization problems and improve the accuracy of the B-spline curve fitting. The expected outcome of this research is to analyse different parameterization methods and knot vector generation for constructing the curve, which is more accurate and fits the data point well. The motivation of this research is to determine the optimal method that is the best to fit all the data points for all the dataset in this research and generate an optimal graph to represent the dataset.

1.4 Project Objectives

Based on the problem statement, the purpose of this research is to identify the suitable parameterization method to construct the best curve fitting of the data points based on the different characteristic of datasets for B-spline curve. The objective of this research is intended to accomplish:

1. To analyse different parameterization methods for B-spline curve.
2. To optimize the accuracy of B-spline curve.

1.5 Project Scope

This research is implemented using C++ programming language as this language is simple and fast. This research developed an algorithm to identify the fittest curve using different parameterization method. This research will focus on the B-spline curve with four parameterization methods.

B-spline curve is chosen as it is a standard for CAD data representation nowadays. This is because the data points and the shape of B-spline curve are able to be controlled and the curve is considered flexible as multiple knot values is used in knot vector, type of knot vector can be changed, and the order of degree (k) of the basis function can be changed. The parameter values from the parameterization methods indicate the distribution or structure of the point cloud. The parameterization methods used in this research are uniform, chord length, centripetal and exponential methods. Most of the previous works focus on using one or two methods that were thought to be the most suitable. The methods results vary from one another and this depends on the algorithm implemented in the research. GA and DE algorithm are used to handle the optimization problems.

In the starting of the research, an analysis of literature review is done to have a further understanding about the methods or algorithms implemented by others that are related to this research. After that, data collection is done and this research used the datasets from (Haron, et al., 2012). Later, curve fitting process is done. In the process, parameterization method is implemented on the data points to obtain the parameter values and the values are used to generate knot vectors. Uniformly spaced and averaging knot vector generation is used to generate the knot vectors. The knot vectors are then applied in the calculation of B-spline control points and basis function. The B-spline curve is then visualised. GA and DE algorithm are used to optimize the accuracy of the B-spline curve fitting. Furthermore, testing process is done on the curve using distance between original data points and generated data points to determine the accuracy of the curve. Lastly, documentation of this research is made.

1.6 Report Organization

This research organization is described as follows:

Chapter 2 discussed about literature review. This chapter introduced the Hermite Curve, Bezier Curve, B-spline Curve, parameterization method and the advantage and disadvantage of each curve are discussed. GA and DE optimization is also discussed in the chapter. Chapter 3 presented the research methodology and introduced the methods involved to have a better view of this research. The timelines of the research are stated. Chapter 4 presented the experimental result which consists of analysis and discussion of the research. Chapter 5 concluded the research and some future work.

CHAPTER 2 LITERATURE REVIEW

2.1 Overview

Reconstruction of object using curve and surface is an important task for the modern era in many fields such as gesture recognition, medical, archaeology, robotic and augmented reality. Surface representation of an object is popular in the CAD. Some parametric curves such as Hermite, Bezier and B-spline are introduced for the modelling of the object. Parametric curve is used because the curve is free form and the shape of the curve is flexible. Parameterization method is applied to the curve to determine the parameter value and calculate the function in order to generate the curve.

The information for the Hermite curve, Bezier curve, B-spline curve and the parameterization methods used in the research are discussed in this chapter to provide a brief understanding about the research. The advantages and disadvantages of the curves and parameterization methods are stated. Some previous works related to the topic are shown for each of the curves.

2.2 Hermite Curve

Hermite curve is known as Parametric Cubic Curve (PC-Curve). It is a 2D curve and the fundamental of interactive curve design. The curve is an interpolation curve as it passes through the control points. Hermite curve is geometrically determined by the coordinates and tangent of the starting point and ending points. The shape of the curve can be adjusted by changing the tangents after the control points is positioned (Chi, et al., 2008). It interpolates the points $P_1, P_2, \ldots, P_n$ in order to obtain a smooth function. It has only up to C^1 continuity which the first derivative is continuous at the points but the second derivative is discontinuous (Ismail, 2015).

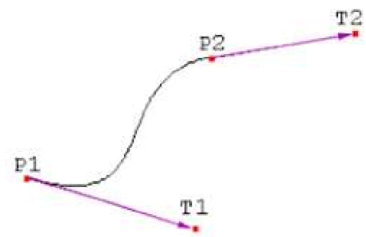


Figure 2.2.1 Hermite Specification

Figure 2.2.1 shows the Hermite Specification. The following vectors are needed to compute the Hermite curve which are: P_1 which is the starting point, T_1, the tangent which represent the direction and magnitude of how the curve leaves the starting point, P_2 which is the ending point and T_2 which is the tangent for P_2.

Hermite curve is represented by the x, y and z coordinates and each of the coordinates are a cubic function with parameter value of u. Equation 1 is the general parametric equation for Hermite Curve. Equation 3 – Equation 5 can be generalised to become Equation 1 and Equation 2 and can be retrieved from Mortenson (2006). Equation 6 represents the matrix form of the parametric equation.

$$\vec{P}(u) = \sum_{i=0}^{3} \vec{a_i} u^i \qquad 0 \leq u \leq 1 \qquad (1)$$

$$\vec{P}(u) = \vec{a}_0 + \vec{a}_1 u + \vec{a}_2 u^2 + \vec{a}_3 u^3 \qquad (2)$$

$$x(u) = a_{0x} + a_{1x} u + a_{2x} u^2 + a_{3x} u^3 \qquad (3)$$

$$y(u) = a_{0y} + a_{1y} u + a_{2y} u^2 + a_{3y} u^3 \qquad (4)$$

$$z(u) = a_{0z} + a_{1z} u + a_{2z} u^2 + a_{3z} u^3 \qquad (5)$$

In matrix form:

$$\vec{P}(u) = [U]^T \quad [C] \qquad (6)$$

Where $\vec{P}(u)$ represents the vector point function, a_i is the vector of the polynomial coefficient and u^i represents the parameter of the point at i. Both the variables depend on the control point at i^{th} position, $x(u), y(u)$ and $z(u)$ represent the *x, y* and *z* coordinates function of the point given the parameter *u*. For the matrix form, $[U]^T =$

$[u^0\ u^1\ u^2\ u^3]^T$ shows the parameters and $[C] = [a_0\ a_1\ a_2\ a_3]^T$ where $[C]$ is the coefficient vector.

Equation 1 shows that the vector function of control point of the curve is the summation of the polynomial coefficients and the parameter value. The parameter value, u will always be in the range of 0 to 1. Equation 2 is the expended vector from Equation 1. Equation 3 – Equation 5 show x, y and z coordinates of the control points.

Hermite curve segment with two endpoints, P_0 and P_1 can be considered and apply the boundary condition given that P_0 at u = 0 and P_1 at u =1. After some calculation and substitution, Equation 7 is given as according to Ching-Shoei (2015):

$$\bar{P}(u) = [u^0\ u^1\ u^2\ u^3] \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \overline{P_0} \\ \overline{P'_0} \\ \overline{P_1} \\ \overline{P'_1} \end{bmatrix} \tag{7}$$

Where $\bar{P}(u) = [U]^T\ [M_H]\ [V]$

$[M_H]$ is the Hermite matrix and $[V]$ represents the geometric coefficient vectors which consists of the control points and tangents. The equation is based on the starting and ending control points. The combination of $[U]^T\ [M_H]$ will give the basis function for Hermite curve. The function determines the shape and characteristic of the curve.

There some previous works that are based on Hermite curve. For the research done by Ching-Shoei (2015), cubic Hermite curve is used in designing the image and curve. In the research paper, a few definition and theorem of the unit circle Hermite curve are provided. The objective is to find the medical axis transform for a certain region bounded by the curve. Moreover, according to Ostaszewski and Kuzmierowski (2015), they proposed and algorithm to identify the trajectory of transition of the assigned signal line using Hermite Interpolation Method. The researchers introduced traversal time T to solve the problem of unsatisfied trajectory generated due to high value of velocity vector which facing a small difference between the points. Uhlmann, et al. (2016) integrated a new model of snake based on the Hermite Spline Interpolation method to segment the bio-images semi-automatically. The method could produce a curve with sharp corners of the contour and could give a better guidance for the automated segmentation. Chi, et al. (2008) presented L-OGH which is the curve length minimization of optimized geometric Hermite (OGH). OCG is extended from standard

Hermite method and is used to ensure that the Hermite curve is geometrically smooth by optimising the magnitudes of tangent vectors.

Hermite curve is consists of linear combination of the tangent and the coordinates of the control points for each of the parameter *u*. The curve can be used to generate geometrically intuitive curve. Hermite curve is usually used for interpolation of control points so that a smooth continuous curve is obtained. It is easy for the Hermite interpolation to construct a smooth piecewise curve. The symmetry of the data points in Hermite curve is important in geometric design where each of the points consists of tangent vector. There are some limitation for the Hermite curve. The curve is global controlled where changes in the position of a data point will affect the shape of the curve. Besides, Hermite curve is limited to 3^{rd} degree of polynomial. It is difficult to identify the specification of tangents for the control points as sometime the informations are not given. The tangent vector is a must in Hermite curve in order to calculate the basis function. Therefore, Bezier Curve is introduced.

2.3 Bezier Curve

Bezier curve is one of the important curve in CAD application. The curve is defined by the shape of the control polygon that consists of segment of other control points. The curve is an approximating curve which will pass through the first and last control points as there are multiple Bezier curve combined. The curve may interpolate or extrapolate the control points. The basis function of a curve will determine the structure of the control points and curve. For the Hermite curve, there are two control points which represent the endpoints and two parametric derivatives which are the tangent vectors. For Bezier curve, two control points is used to define the endpoints and another two data points control the tangents in a geometric way. Paul de Castlejau was the one who first developed Bezier curve using Castlejau’s algorithm. The curve become popular when a French designer, Pierre Bezier used the curve in designing automobiles in 1962 (Li, et al., 2018).

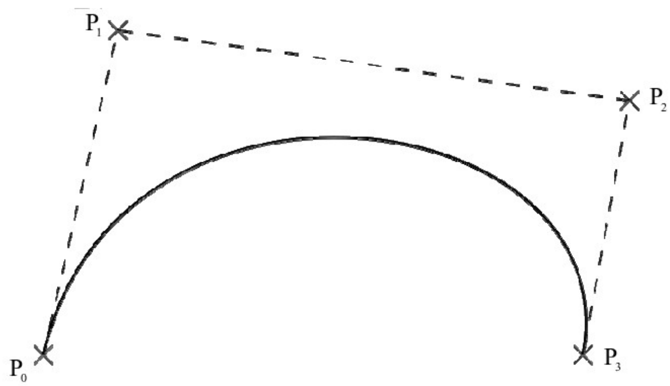


Figure 2.3.1 Bezier Curve

Figure 2.3.1 is the example of Bezier curve. P_0, P_1, P_2 and P_3 are the control points. From the figure, tangent vectors are not used to determine the characteristic. This allow the designer to observe the relationship between the control points and curve. The points will control the characteristic polygon or control polygon. The curve will only pass through the starting and ending points and the curve is always tangent to the first and last polygon segment.

In the Bezier curve, Bernstein polynomials are used as the weighting function for each of the control points. In the mathematical form, Bernstein polynomial is a linear combination of Bezier basis function. Equation 8 represents the Berstein polynomial function and retrieved from (Chantakamo & Dejdumrong, 2013). Equation 9 is the Bezier basis function and equation 10 show the coefficient of the basis function.

$$\bar{P}(u) = \sum_{i=0}^{n} \bar{P_i}\, B_{i,n}(u) \qquad 0 \leq u \leq 1 \qquad (8)$$

$$B_{i,n}(u) = C(n,i)u^i(1-u)^{n-i} \qquad (9)$$

$$C(n,i) = \frac{n!}{i!(n-i)!} \qquad (10)$$

Where $\bar{P}(u)$ is the control points of the curve, $B_{i,n}(u)$ represents the Bezier basis function, u is the parameter value, n is the number of points and $C(n,i)$ shows the coefficient of the function.

From Equation 8, Berstein polynomial is the summation of the Bezier basis function with the control points given the parameter, u is in the range of 0 and 1 for $n+1$ data points. The number of control points determine the order of the Bezier curve. Four control points will always produce a Bezier curve with the order of 3. Equation 9 is the basis function and together with the coefficient which represented by equation 10 will determine the shape of the curve. The point that is corresponding to u is the average weightage of all the control points.

As shown in Iglesias, et al. (2015), they implemented a new method using bat algorithm which is a kind of optimization method for polynomial Bezier curve fitting. This is the first paper to use the algorithm in the field of data fitting for geometric modelling, therefore they lack of benchmark to analyse and compare the result. The approach able to provide a more suitable parameter of data points for the Bezier curve fitting. After that, Chantakamo and Dejdumrong (2014) use Progressive Iterative Approximation (PIA) method to convert the rational Bezier curve into polynomila Bezier curve by calculating a new control point. It is proven to be more efficient than other algorithms in terms of computational time because of the simpler algorithm. Bezier curve is used by Song, et al. (2015) to track path without any comand. The curve always pass through the first and last control point and there are tangent to the line connecting the control points. The researchers studied on the parameter and factors that will affect the efficeincy of Bezier curve in path tracking and predict the future path. The previous work done by Mohanty, et al. (2015) used Bezier Curve Interpolation method to remove noise from the image while maintaning the structure of image. Firstly, they find out the 4 control points from the uncorrupted pixels of the image and the location of the corrupted pixels. Next, they used Bezier Curve Interpolation to restore and preserve the edge of the image with the corrupted pixels matrix.

There are some improvements over the Hermite curve. First derivatives which represent the tangent vectors are not used to construct the Bezier curve. The shape of the curve is determined by the control points. The degree of the Bezier curve can up to n^{th} degree curve which is related to the number of data points, $n+1$. Because of the higher-order derivatives, Bezier curve is considered smoother. The conection between the control points will form a control polygon which ensure that the Bezier curve is contructed smoothly in the polygon following the control points. The curve is also invariant to translation and rotation. However, Bezier curve is difficult to construct a complicated curve which is a very high-order polynomial. The curve requires a lot of control points and this will slow down the computational time of the construction. Bezier basis function have global support over the entire curve. Movement of a control point will affect the shape of the whole curve. Therefore, B-spline curve is introduced to overcometh problems faced.

2.4 B-spline Curve

B-spline method is generalised from the Bezier curve in order to solve the difficulty faced while modifying the control point. The curve is more complex than Bezier curve and requires more information about the curve such as degree of curvature and knot vector. It consists of *n+1* control points. The basis function for the curve is non-negative for all the parameter value and the summation of the value is 1. B-spline curve is more flexible than Bezier curve as the basis function and curves are controlled by the knot vectors, the k value and number of position of control polygon vertices. B-spline curve is a polynomial function that use less amount of control points to provide local approximations to curve (Liu, et al., 2014). This means that changes in a control points will only affect in the range of knot vectors. It has the ability to increase the number of control points without increasing the degree of curve. B-spline curve able to join several Bezier curve of lower degree however it is tedious and difficult to maintain the continuity of the desired order at the control points. B-spline equation is recursive. Equation 11 is the B-spline curve function with parameter u. Equation 12 is the B-spline basis function.

$$C(u) = \sum_{i=0}^{n} P_i \, N_{i,k}(u) \qquad 0 \leq u \leq u_{max} \tag{11}$$

$$N_{i,k}(u) = \frac{u-t_i}{t_{i+k-1}-t_i} \, N_{i,k-1}(u) + \frac{t_{i+k}-u}{t_{i+k}-t_{i+1}} \, N_{i+1,k-1}(u) \tag{12}$$

Where

$$N_{i,k}(u) = \begin{cases} 1, & if\ t_i \leq u \leq t_{i+1} \\ 0, & otherwise \end{cases}$$

and

$$t_i = \begin{cases} 0, & i < k \\ i-k+1, & k \leq i \leq n \\ n-k+2, & i > n \end{cases}$$

Where $C(u)$ represents B-spline curve function, P_i is the control point at i^{th} position, $N_{i,k}(u)$ is the normalised B-spline basis function, u is the parameter with the range from 0 to u_{max}, k is the order with degree, t is the knot value.

Equation 11 shows that the B-spline curve is the summation of the control points and B-spline basis function for $n+1$ data points. The summation of every points of the curve is equals to 1. Equation 12 is the basis function defined using the Cox-de Boor algorithm. The parameter k is independent to the number of control points and is used to control the degree of curve (k-1). The denominators for the equation can be 0 and that section will be presumed as 0. The knot value, t can be defined using equation 14. It is used to determine the non-periodic or open curve knots. The relation between the parameter shows that a minimum of two, three and four control points are needed to construct respective B-spline curve. Normally, cubic B-spline curve is sufficient for most of the applications.

Some previous works used B-spline curve for their research. As shown in Ravari and Hamid (2016), they implemented adaptive group testing to find out the salient point in the B-spline curve fitting problem and update the curve model. The researchers repeated the process of updating the B-spline model until it reached the Akaike Information Criteria (AIC). Besides, an interpolation and approximation methods using iterative geometric algorithm was introduced for the B-spline data fitting. They made a comparison between the iterative geometric-based interpolation and approximation methods and the standard curve fitting methods. The proposed method had a better performance and local control over the B-spline curve however it difficult to fit the high quality data and was slower if the data points were sparse (Kineri, et al., 2012). As shown in Yang, et al. (2012), they applied evolutionary algorithm to the B-spline curve fitting process and reverse control point process. They select the optimal control point for the curve and used them in the B-slpine curve fitting to obtained the optimal fitting curve. The research done by Wu and Tao integrated geometric constraints for the shape modification of B-spline curve. The control points of the curve were adjusted using constrained optimization.

B-spline composed of several connected polynomial curves. Each of the segments of the curve are affected by the k degree of control points and each of the control points will affect the k segment. The degree of the curve is defined by the designer and is independent to the number of contol points. The curve is located inside the convex hull which is similar to Bezier curve and it is symmetric. Therefore, B-spline curve is smooth. Local modification can be done where by changing the control points will only affect the local neighbour in the knot vectors. B-spline curve is considered flexible because the curve can be controlled by many methods. The methods are changing the knot vector type, changing the k of the basis function, adjusting the number and postion of the control points, using multiple polygon vertices and also including multiple knot values in the knot vector. However, B-spline curve is polynomial curve and is difficult in representing many useful and complex curves. Thus, NURBS which is a generalization of B-spline is required. For this research, B-spline curve will be the focus. Figure 2.4.1 shows the step for the B-spline curve fitting. There are four processes with different steps in the research. The flowchart is adapted from Lim and Haron (2014).

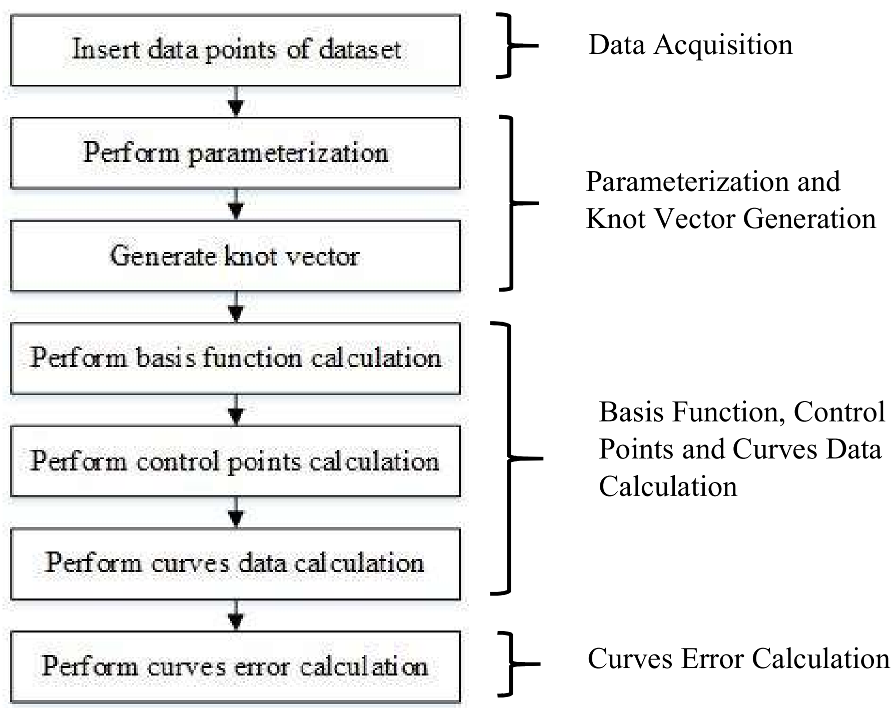


Figure 2.4.1 Flowchart for B-spline curve approach

Data Acquisition

The data used for this research is from (Haron, et al., 2012) which consists of six datasets with different number of data points. In this research, only five datasets are focused. The data points are in 3D coordinates which are *x, y*, and *z* coordinates. Each of the datasets will undergo parametrization and knot vector generation and then proceed to the calculation to obtain the basis function, and control points.

Parameterization and Knot Vector Generation

Four parameterization methods is used to define the length of the parametric interval and model the B-spline curve. Different parameterizations are implemented in this research to identify the arrangement between the knots. The 3D data points are the input in this process. Throughout the process, the parameters for each of the data points are obtained. The parameters values are then used to generate the knot vectors. The knot vectors are calculated using the uniformly spaced and averaging knot vector generation. Different from parameter values, knot vectors require each of the value to be incremented. Then the knot vectors for each of the data points are calculated.

Basis Function, Control Points and Curves Data Calculation

Once the parameter values and the knot vectors are obtained, the B-spline basis function, $N_{i,k}(u)$ is computed using the values obtained. Each of the data points have respective basis function and the function are performed separately. After the basis functions are calculated, the values are used to compute the control points of the curve.

Curves Error Calculation

Least-squares method will be applied to calculate the control point for each of the data points by using Equation 13.

$$min \sum_{j=1}^{n}(C(u_j) - Q_j)^2 \qquad (13)$$

Where

$C(u_j)$ is the B-spline function and Q_j is the data points at j^{th} position.

This method can approximate the curve well if it is a smooth curve (Dung & Tjahjowidodo, 2017). With the generated results, a new curve data is computed. A comparison between the original and new data could be made in the next process.

2.5 Parameterization Method

Parametrization of the 2D or 3D data points is the important process needed for the curve fitting. The core of parameterization is to create a one-dimensional equation. Later transforms the equation into the extended power flow equation with the equation generated before the parameterization. Parameter is a unique value of the coordinate that are located on the curve or surface. It shows the distribution of the points. In general, a curve consists of many data points and parameter is the one that transforms the data points into the curve. There are some previous works that worked on various parameterization method such as uniform, chord length, exponential, universal, centripetal and hybrid methods. Most common parameterization methods used are uniform, chord length and centripetal parameterization (Wan & Yin, 2014).

Equation 14 represents the mathematical formulation for uniform method. Equation 15 shows the calculation for chord length method. Equation 16 is about the centripetal parameterization method (Wan & Yin, 2014). Equation 17 which represent the exponential method is retrieved from Ammad and Ramli (2019) if the α value is equal to 0.8.

$$\overline{u_1} = 0, \overline{u_i} = \frac{i}{n}, (i = 2, 3, \dots, n-1), \overline{u_n} = 1 \tag{14}$$

$$\overline{u_1} = 0, \overline{u_i} = \overline{u_{i-1}} + |Q_i - Q_{i-1}|, (i = 2, 3, \dots, n-1), \overline{u_n} = 1 \tag{15}$$

$$\overline{u_1} = 0, \overline{u_i} = \overline{u_{i-1}} + \sqrt{|Q_i - Q_{i-1}|}, (i = 2, 3, \dots, n-1), \overline{u_n} = 1 \tag{16}$$

$$L = \sum_{i=1}^{n} (|Q_i - Q_{i-1}|)^{\alpha} \tag{17}$$

Where

$\overline{u_i}$ = ith parameter variable, Q_i = ith data point, *i* = knot interval

All the parameterization methods have respective advantages and disadvantages. For the uniform parameterization, the parameter values can be calculated easily as the knot spacing is uniformly distributed. Nevertheless, this method only can be used for the model which has same length at both side and may cause some random stretching during the rendering. This is because the interpolation of the curve is not the best fit. The advantage of chord length parameterization is the adjustment of the textures will be minimised. This method considers about the distance between data points which is $|Q_i - Q_{i-1}|$. It able to handle the disadvantage of uniform method in representing the data points. However, the control polygon generated have a huge bulge and the data points are twisted. In certain situations, smoother curve is produced using centripetal parameterization. This is due to the evenly spacing of the control points of the curve for texture mapping results. Small loop on the curve may be created if the data points are collinear. For exponential parameterization, designers are able to choose the resulting curve which is a generalization of uniform, chord length and centripetal parameterization methods. The suitable parameter values are difficult to obtain to construct a better curve. (Haron, et al., 2012). In this research, only four commonly used parameterization methods are used such as uniform, chord length, centripetal and exponential method.

As shown in Wan and Yin (2014), they integrated a new parameterization which applied point selection methods on the data points for the curve fitting in 3D. In their research, they explained the uniform, chord length and centripetal parameterization. Three of the parameterization methods have own disadvantages, therefore they proposed the dynamic parameterization method. The curve constructed by the methods proposed was the same as using chord length parameterization if the data points are collinear. If there are points of deflection, the curve will be the same when using centripetal parameterization. The point cloud used for the curve fitting were dense and not suitable for other distributed points. Based on the previous work done by (Haron, et al., 2012), they introduced a new parameterization method for the B-spline curve fitting. The researchers enhanced the hybrid parameterization method by implementing the exponential parameterization method at the beginning. They made a comparison between their proposed parameterization method and the other methods. The proposed method able construct a better curve which the data points are collinear and have a far distance between two successive data points. Suszyński & Świta (2017) used chord length, uniform and centripetal parameterization for the approximation of B-spline surface. The researchers use the B-spline to filter the thermal images. They made a comparison between the three methods. They claimed that chord length parameterization method produced better result with less overhead of computation.

As discussed in the previous section, they focus on identifying the best parameterization methods for curve fitting which is similar to the objective of this research. They stated that designer is the one to determine the optimal parameterization method for the curve at that situation.

2.6 Knot Vector Generation

In order to construct B-spline curve, knot vector is needed to calculate the basis function of the curve. Knot vectors are the one which determine the shape of the curve. There are 2 types of knot vectors which are periodic and non-periodic (open). For periodic knots, the first and the last knots are not duplication. The example of the periodic knots is (0, 1, 2, 3). The curve does not interpolate through the endpoints. Periodic knots are usually used to create closed curve where the first and last control points are the same. For non-periodic knots, the first and last values for the knots are repeated for k times. The example is (0, 0, 0, 1, 2, 3, 3, 3). The curve will pass through the endpoints. The knot vectors are located in one of the two categories which are uniform and non-uniform. Each of the individual knot vector is evenly separated such as (0, 1, 2, 3, 4). After that the values are normalized into the range of 0 and 1 like (0, 0.25, 0.5, 0.75, 1.0). With the uniform B-spline, the shape of the curve can be adjusted by a few ways. First is by moving the control points. If there is a need to pass through the control point, multiple control points are placed at the adjacent points so that the curve can pass through the point. Next is by increasing the order k and joining the ends of the curve to create a closed loop. For non-uniform B-spline, there may have multiple knot vectors and the spaces between are not equal. For example, is like (0, 1, 2, 2, 4).

There are two types of knot vectors generation which are the uniformly spaced knot vector generation and averaging knot vector generation. Parameters of the curve of each data points are needed to calculate the knot vectors. For uniformly spaced knot vector, this method is simple and not affected by the parameters. Equation 18, 19 and 20 is the calculation for the uniformly spaced knot vector.

$$u_0 = u_1 = \cdots = u_p = 0 \qquad (18)$$

$$u_{j+p} = \frac{j}{n-p+1} \qquad for\ j = 1, 2, \ldots, n-p \qquad (19)$$

$$u_{m-p} = u_{m-p+1} = \cdots = u_m = 1 \qquad (20)$$

Where

p = degree of curve

n = numbers of parameters

m = numbers of knots

For the B-spline curve that have degree p, $n + 1$ parameters are needed to calculate the knot vector with $m + 1$ knots. The number of knots, m is calculated with $m = n + p + 1$. The curve is clamped as the first $p + 1$ knots are 0 and the last $p + 1$ knots are 1 which is represented in the equation 18 and equation 20. For the remaining $n - p$ knots, the knots are uniformly spaced so that could satisfy certain desired condition. Equation 19 shows that each of the position of the parameters is divided by the difference of number of parameters and the degree of curve. However, this method has an issue which the linear equation of the system would be singular if the method is used together with the chord length parameterization method of the global interpolation (Liu, 2003).

Another famous method is the averaging knot vector generation which is suggested by de Boor. The following equation is the computation of the knot vector:

$$u_0 = u_1 = \cdots = u_p = 0 \tag{21}$$

$$u_{j+p} = \frac{1}{p}\sum_{i=j}^{j+p-1} t_i \qquad for\ j = 1, 2, \dots, n-p \tag{22}$$

$$u_{m-p} = u_{m-p+1} = \cdots = u_m = 1 \tag{23}$$

Where

p = degree of curve

n = numbers of parameters

m = numbers of knots

The equations are quite similar to the uniformly spaced knot vector as the first $p + 1$ and the last $p+1$ knots are 0 and 1 respectively. The main difference is the calculation of the remaining $n - p$ knots which is represented in equation 22. The equation is the summation of the parameters of the internal knots (Shene, 2014).

2.7 Optimization

In this research, two optimization techniques are used to improve the performance in terms of accuracy of the B-spline curve fitting which are Genetic Algorithm (GA) and Differential Evolution (DE). Different methods will produce different results as the algorithm performed are different. In year 1975, John Holland introduced the GA which is according to the principle of genetics and evolution which is used to describe the growth of the populations associated with the problems (Lim & Haron, 2013).

GA is considered as one of the popular computational paradigms for the optimization and search problems among other methods. The algorithm is a powerful tool to solve some of the complex optimization problems in modern days. In the GA operations, the populations are prepared as abstract representations or so called chromosomes of an individual or phenotypes to an optimization problem. The chromosomes will be divided into several group of genes. The evolution of the operation begins with randomly generated individuals in a populations and this happens in generations. In each of the generation, the fitness or quality of every individual in the populations is evaluated by using the objective function. The function is used to test the strength of chromosomes against the problems (Lim & Haron, 2013). The fitness values obtained will be used to select the parent in most of the selection operation techniques as the best genetic materials are carried forward to next generation. In an ideal situation, the algorithm is likely to evolve into a better solution throughout the time although it could not be assured that the algorithm convergence to global optima (Gálvez, et al., 2012). New generation of chromosomes will replace old generation chromosomes if the fitness value is better than the parents. GA is used because of the objective function as it is multimodal and the number of variables changes according to the number of knots. Besides, the objective function is difficult to differentiate by knots because the knots are nonlinear variables for the function and the number of the knots are not known in advance. The following figure 2.7.1 is the general steps of GA represented in (Yoshimoto, et al., 1999).

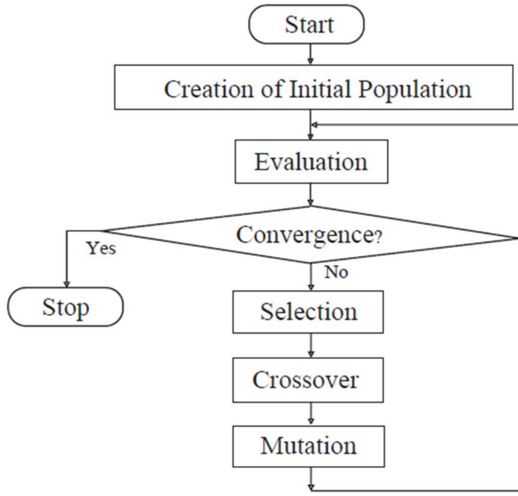


Figure 2.7.1 Flow of Genetic Algorithm

At the beginning of the algorithm, initial population of the will be created and the chromosomes will be evaluated using the fitness function to identify the fitness or strength. The fittest chromosomes will be selected in the selection operation so that better genetic materials will be carried forward to next generation. In the crossover operation, the genetic materials will be merged and generate a new generation. After that, in mutation operation, new genes will be added into the chromosome. Each of the iteration shown is considered as a generation.

In year 1997, Storn and Price introduced DE. This algorithm is a stochastic direct search method which represents population-based search strategy for mathematical optimization for multidimensional functions. It is simple for implementation compared to other algorithms and has fast convergence characteristic (Hasegawa, et al., 2013). The algorithm improves the population of individuals for several generations through the operation of mutation, crossover and selection for global optimization. The genes for the chromosomes are generated by using random function and the fitness of the chromosome is evaluated using the fitness function. DE is considered unique because of the differentiation which realized the differences between individuals in a fast and simple linear operator. The algorithm is widely used to solve many optimization problems such as Representation of Multi Sensor Fusion, Optimization of Industrial Compressor Supply System and Non-Imaging Optical Design. The performance of DE is evaluated based on size of population, strategy and parameter setting in order to produce trial vectors and replacement scheme (Hasan, et al., 2018). Figure 2.7.2 is the general step of the DE algorithm (Pandunata & Shamsuddin, 2010).

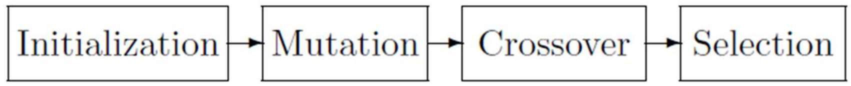


Figure 2.7.2 Flow of Differential Evolution

For the initialization phase, population for each of the generation will be generated with random values. In the mutation phase, new vectors or populations will be created using the process between two elements of the population and later merge with the third element characteristic. Crossover is where the process of adding the mutation vectors into the population and remain some of the original vectors. The last phase which is the selection will determine the best vector for the next generation using the objective function.

2.8 Summary

Different previous works were discussed and studied in order to determine and understand the focus of this research. B-spline curve is chosen for this research is because of its good characteristic and local control. B-spline curve is better compared with the Hermite and Bezier curve as the curve able to handle the difficulties faced by both of the curve. The calculation is more complex and the curve requires more information such as the degree of curve and knot vector. The shape of the curve is controlled by the control polygons. Besides, four parameterization methods are used in this research which are uniform, chord length, centripetal and exponential. These four methods are widely used by previous researcher because of the simple and similar calculation. The other parameterization such as universal, foley and hybrid method are not used because of the complexity of the calculation which involve basis function and angle. For the knot vector generation, both uniformly spaced and averaging knot vectors are used to compare the differences between them and the results produced. General concepts and the flow of GA and DE algorithm were discussed.

CHAPTER 3 RESEARCH METHODOLOGY

3.1 Overview

This chapter discussed about the research methodology involved in this research. A brief description about the methodologies of the research is explained. The overview of the research is about comparing the parameterization methods among the four parameterization methods chosen to construct a smooth B-spline curve and fit the data points well. The phases of the methodology are further discussed.

3.2 Research Methodology

3.2.1 Methodologies

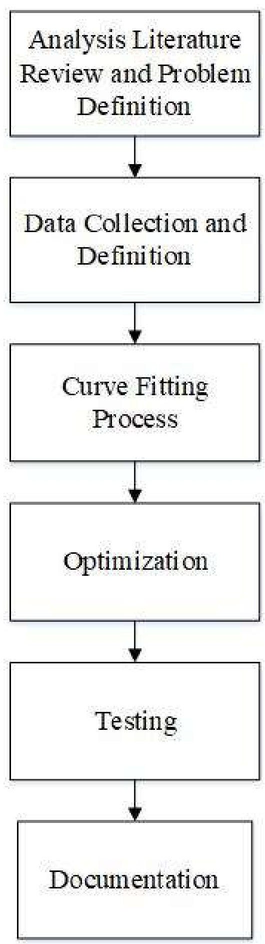


Figure 3.2.1.1 Research Framework

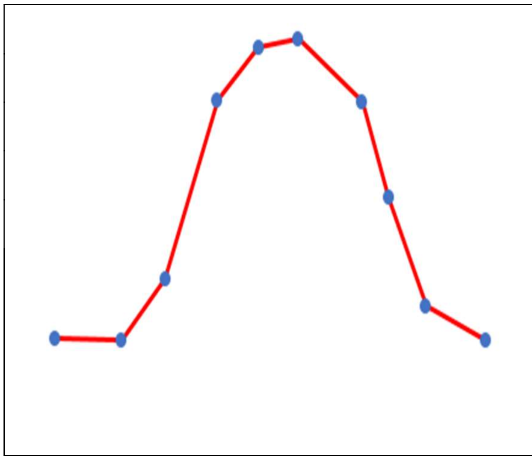
Figure 3.2.1.1 represents the research framework of this research. The framework is based on the waterfall model. There are some modifications made to suit this research. The model is an example of plan-driven process which the processes are needed to be planned and schedule before starting the research. Theoretically, one phase is needed to complete in order to continue to the next phase. However, there will be iteration between the stages when it is practically done (Sommerville, 2011). The outcome of this research is to determine the ideal parameterization methods for B-spline curve fitting. This can be done by comparing different parameterization methods that can fit the data points in the datasets used for the curve. Optimization will be carried out to optimize the accuracy of the curve fitting. Algorithm will be made and testing will be performed.

Analysis Literature Review and Problem Definition

For the first phase of the research framework, analysis literature review and problem definition are done. In this phase, different types of curves such as Hermite, Bezier and B-spline curves are studied. To have a better understanding of the curves and determine the advantages and disadvantages of those curves, some reviews on the related previous works are carried out. This research focused on the curve which is the foundation of the surface. B-spline curve is chosen because the curve supports local control and is flexible. The reviews on the parameterization methods are done to explore the four methods used in this research and identify the advantages and disadvantages. GA and DE will be used in the research to handle the optimization problems. After reviewing the previous works, the project objective and scopes are defined.

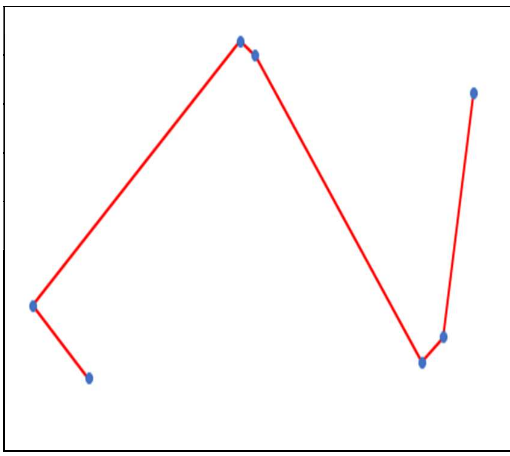
Data Collection and Definition

The datasets used in this paper are the dataset from (Haron, et al., 2012). There are six datasets used where each of the datasets consists of 3D data points which are the x, y and z coordinate. Only five datasets are focused in this research. Each of the datasets have their own characteristic which will later generate different structure of curve. There are no collinear and consecutive data point in Dataset 1 as only consists of simple data points. Dataset 2 consists of data points which has large distance and can form a straight line. The data points at the start and end are collinear in dataset 3. Dataset 4 is the combination of the previous datasets. The characteristic of data points of dataset 5 is similar to dataset 2 but with the existence of the collinear points. Table 3.2.1.1 – Table 3.2.1.5 show the graph and the data points for each of the dataset provided. The details of the dataset will be shown in Appendix A.



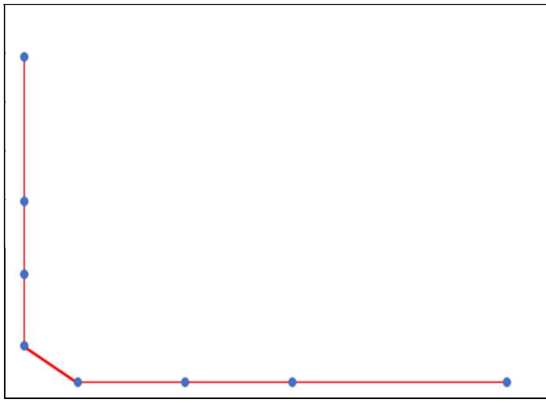
x	y	z
0.58072	2.08688	0
3.50755	2.05734	0
5.36585	3.24051	0
7.64228	6.74273	0
9.40767	7.79716	0

Table 3.2.1.1 Dataset 1



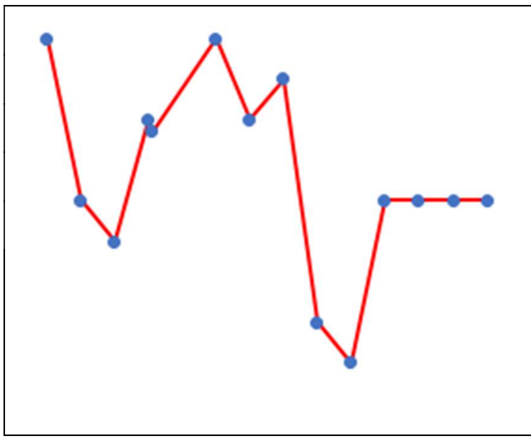
x	y	z
1.83761	1.33271	0
0.726496	2.6504	0
4.82906	7.40482	0
5.12821	7.15185	0
8.4188	1.62069	0

Table 3.2.1.2 Dataset 2



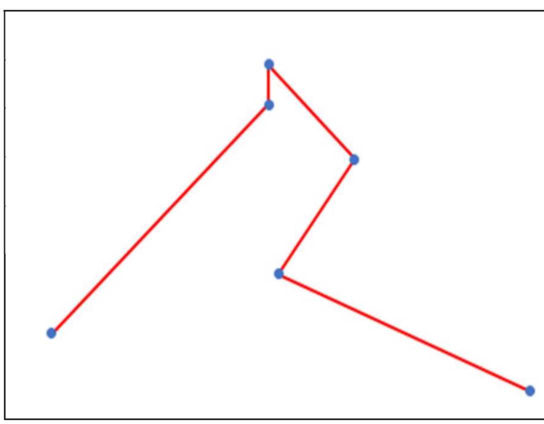
x	y	z
0	9	0
0	5	0
0	3	0
0	1	0
1	0	0

Table 3.2.1.3 Dataset 3



x	y	z
32	14	0
34	10	0
36	9	0
38	12	0
38.2	11.7	0

Table 3.2.1.4 Dataset 4



x	y	z
1.40152	2.75472	0
4.84848	8.41509	0
4.84848	9.4	0
6.21212	7.0566	0
5	4.22642	0

Table 3.2.1.5 Dataset 5

Curve Fitting

In the curve fitting process, various operations are carried out such as the parameterization, knot vector generation, basis function and control points calculation. Figure 2.4.1 is the flow in this process. With the input of the data points obtained, the parameters and knot vectors are computed. There are four parameterization methods used which are uniform, chord length, centripetal and exponential methods. Uniformly spaced and averaging knot vector generation are used to calculate the knots using the parameters. The values acquired will be used for the next few phase such as basis function and control points calculation.

Optimization

There are two optimization methods that are used in this research which are Genetic Algorithm (GA) and Differential Evolution (DE) to optimize the error. In GA optimization, there are selection, crossover and mutation operation involved. Firstly, chromosomes in each population are initialized within the given range. Next, the chromosomes are evaluated using the fitness function. After the evaluation, tournament selection is used in the selection operation. According to Lim, et al. (2018), tournament selection is better compared to other methods such as random and roulette wheel selection. This is because of more chromosomes are involved in the selection operation. Therefore, there is a higher chance for the chromosomes which have higher fitness value to be selected. After that, one random number is generated and compared with the crossover probability. If the random number is smaller, crossover operation is performed with the input in selection operation. Next is mutation operation. Similar to previous operation, one random number is generated and compared with the crossover probability. If the random number is smaller, mutation operation is performed to produce the children chromosomes. The children chromosomes are evaluated using the same fitness function. For the replacement operation, weak parent replacement is used to replace the lower fitness value parent with higher fitness value children. The operations are repeated if the termination criteria which refer to the generation is not achieved.

For DE optimization, the sequence of selection, crossover and mutation operation is difference from GA optimization. Firstly, chromosomes in each population are initialized within the given range. Next, the chromosomes are evaluated using the fitness function. Next is the mutation operation. Three chromosomes are randomly selected to generate mutant vector. After that is crossover operation. One random number is generated. Trial vector is generated in this operation if the random number is smaller than the crossover probability. The trial vector is evaluated using the fitness function. The following operation is selection operation. The fitness value for the trial vector and target vector are compared and the vector with smaller fitness value is selected. The operations are repeated if the termination criteria which refer to the generation is not achieved.

No	Parameter	Value
1.	Number of generation	1000
2.	Population size	40
3.	GA crossover probability	0.7
4.	GA mutation probability	0.01
5.	DE crossover probability	0.8
6.	DE differential weight	0.3
7.	Number of testing	10

Table 3.2.1.6 Parameters setting for GA and DE

Table 3.2.1.6 shows the parameters setting in GA and DE optimization. The parameters are referred to the research done by Lim, et al. (2013).

Testing

The error is tested using the difference of the distance between the original data points and generated data points.

$$Error = \sum_{i=1}^{n}(|D_O - D_G|) \qquad (23)$$

It is also used to quantify the performance of the algorithm.

Documentation

After all the phases are done, documentation will be carried out. Report of the research will be done so that others could have an understanding about what had been done.

3.2.2 Hardware and Software Requirements

In this research, several tools, hardware and software are used so that it can be successfully carried out. The details of the software are stated below and the hardware specification is shown in Table 3.2.2.1.

Hardware

The hardware specification is based on the laptop that will be used in this research.

Hardware	**Description**
Operating System	Window 10 64-bit
Processor	Intel®Core™ i5-6200U CPU @ 2.30GHz
RAM	8GB
GPU	Nvdia GeForce 940M

Table 3.2.2.1 Hardware Specification

Software

Microsoft Visual Studio Community 2017

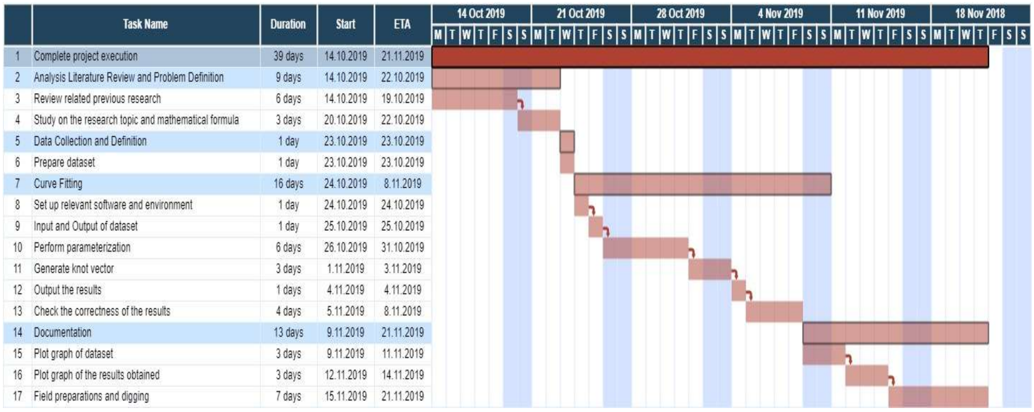
Microsoft Visual Studio Community 2017 is an integrated development environment (IDE) created by Microsoft. The software usually used to develop computer programs. Visual Studio consists of code editor which can support different programming languages such as C++. It includes many library functions which is useful for the research.

Microsoft Excel 2016

Microsoft Excel is a spreadsheet program of Microsoft Office. It is used to perform some tabling and calculation. The tables presented consists of rows and columns which have values for mathematical manipulation. Microsoft Excel offers some other features in addition to the standard features such as extensive graphing and charting capabilities.

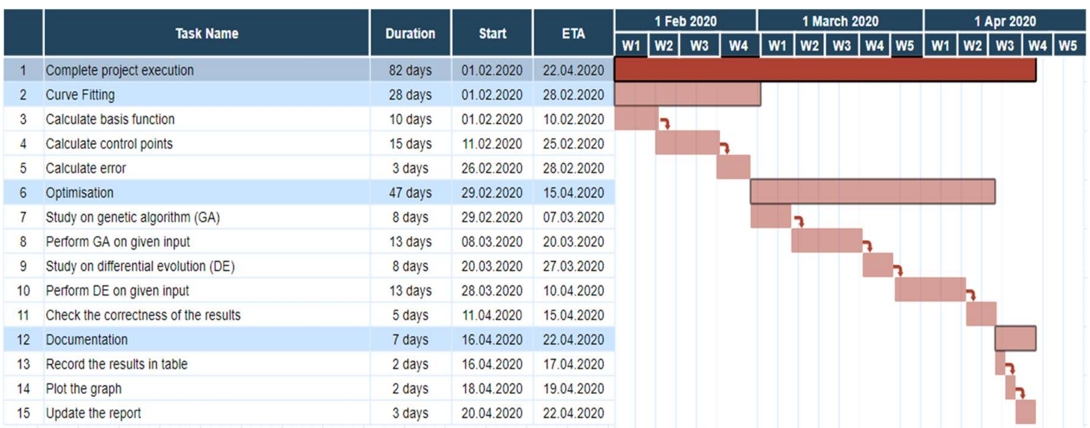
3.3 Timeline

Figure 3.3 shows the timeline of FYP1 and FYP 2



	Task Name	Duration	Start	ETA
1	Complete project execution	39 days	14.10.2019	21.11.2019
2	Analysis Literature Review and Problem Definition	9 days	14.10.2019	22.10.2019
3	Review related previous research	6 days	14.10.2019	19.10.2019
4	Study on the research topic and mathematical formula	3 days	20.10.2019	22.10.2019
5	Data Collection and Definition	1 day	23.10.2019	23.10.2019
6	Prepare dataset	1 day	23.10.2019	23.10.2019
7	Curve Fitting	16 days	24.10.2019	8.11.2019
8	Set up relevant software and environment	1 day	24.10.2019	24.10.2019
9	Input and Output of dataset	1 day	25.10.2019	25.10.2019
10	Perform parameterization	6 days	26.10.2019	31.10.2019
11	Generate knot vector	3 days	1.11.2019	3.11.2019
12	Output the results	1 days	4.11.2019	4.11.2019
13	Check the correctness of the results	4 days	5.11.2019	8.11.2019
14	Documentation	13 days	9.11.2019	21.11.2019
15	Plot graph of dataset	3 days	9.11.2019	11.11.2019
16	Plot graph of the results obtained	3 days	12.11.2019	14.11.2019
17	Field preparations and digging	7 days	15.11.2019	21.11.2019

Figure 3.3.1 Gantt chart of FYP1



	Task Name	Duration	Start	ETA
1	Complete project execution	82 days	01.02.2020	22.04.2020
2	Curve Fitting	28 days	01.02.2020	28.02.2020
3	Calculate basis function	10 days	01.02.2020	10.02.2020
4	Calculate control points	15 days	11.02.2020	25.02.2020
5	Calculate error	3 days	26.02.2020	28.02.2020
6	Optimisation	47 days	29.02.2020	15.04.2020
7	Study on genetic algorithm (GA)	8 days	29.02.2020	07.03.2020
8	Perform GA on given input	13 days	08.03.2020	20.03.2020
9	Study on differential evolution (DE)	8 days	20.03.2020	27.03.2020
10	Perform DE on given input	13 days	28.03.2020	10.04.2020
11	Check the correctness of the results	5 days	11.04.2020	15.04.2020
12	Documentation	7 days	16.04.2020	22.04.2020
13	Record the results in table	2 days	16.04.2020	17.04.2020
14	Plot the graph	2 days	18.04.2020	19.04.2020
15	Update the report	3 days	20.04.2020	22.04.2020

Figure 3.3.2 Gantt chart of FYP2

3.4 Summary

In this research, C++ language is used as the programming language to create the program using Visual Studio Community 2017. The research have six phases which are analysis literature review and problem definition, data collection and definition, curve fitting process, optimization, testing and lastly documentation.

CHAPTER 4 ANALYSIS AND DISCUSSION

4.1 Overview

This chapter presented the analysis and discussion on the results. Different type of parameterization methods and knot vector generation are analysed and discussed to have an understanding about the differences between each of the methods. The comparison on original data points and generated data points are included. The error distance for each parameterization methods using different knot vector generation and error after using GA and DE optimization are analysed and discussed.

4.2 Analysis and Discussion

After some of the calculation, the results of the parameterization and knot vector generations are obtained. Table 4.2.1 – Table 4.2.10 are the results generated from the calculation for each of the dataset.

Parameterization methods / Number of data point	Uniform	Chord Length	Centripetal	Exponential
1	0.00000	0.00000	0.00000	0.00000
2	0.11111	0.12339	0.11797	0.12142
3	0.22222	0.21627	0.22032	0.21815
4	0.33333	0.39236	0.36126	0.37953
5	0.44444	0.47905	0.46014	0.47108
6	0.55556	0.55472	0.55253	0.55319
7	0.66667	0.68291	0.67277	0.67837
8	0.77778	0.77540	0.77491	0.77479
9	0.88889	0.88860	0.88791	0.88812
10	1.00000	1.00000	1.00000	1.00000

Table 4.2.1 Parameterization of Dataset 1

Uniform		Chord Length		Centripetal		Exponential	
A	B	A	B	A	B	A	B
0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
0.14286	0.22222	0.14286	0.24401	0.14286	0.23318	0.14286	0.23970
0.28571	0.33333	0.28571	0.36256	0.28571	0.34724	0.28571	0.35625
0.42857	0.44444	0.42857	0.47538	0.42857	0.45797	0.42857	0.46794
0.57143	0.55556	0.57143	0.57223	0.57143	0.56181	0.57143	0.56755
0.71429	0.66667	0.71429	0.67101	0.71429	0.66674	0.71429	0.66879
0.85714	0.77778	0.85714	0.78230	0.85714	0.77853	0.85714	0.78043
1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000

A = Uniformly Spaced

B = Averaging

Table 4.2.2 Knot Vector Generation of Dataset 1

Parameterization methods / Number of data point	Uniform	Chord Length	Centripetal	Exponential
1	0.00000	0.00000	0.00000	0.00000
2	0.16667	0.08670	0.13292	0.10462
3	0.33333	0.40258	0.38665	0.39894

4	0.50000	0.42229	0.45002	0.43092
5	0.66667	0.74603	0.70688	0.73108
6	0.83333	0.77762	0.78712	0.77773
7	1.00000	1.00000	1.00000	1.00000

Table 4.2.3 Parameterization of Dataset 2

Uniform		Chord Length		Centripetal		Exponential	
A	B	A	B	A	B	A	B
0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
0.25000	0.33333	0.25000	0.30386	0.25000	0.32320	0.25000	0.31149
0.50000	0.50000	0.50000	0.52363	0.50000	0.51451	0.50000	0.52031
0.75000	0.66667	0.75000	0.64865	0.75000	0.64800	0.75000	0.64658
1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000

A = Uniformly Spaced

B = Averaging

Table 4.2.4 Knot Vector Generation of Dataset 2

Parameterization methods / Number of data point	Uniform	Chord Length	Centripetal	Exponential
1	0.00000	0.00000	0.00000	0.00000
2	0.14286	0.22970	0.18440	0.21130
3	0.28571	0.34455	0.31479	0.33266
4	0.42857	0.45939	0.44518	0.45401
5	0.57143	0.54061	0.55482	0.54599
6	0.71429	0.65545	0.68521	0.66734
7	0.85714	0.77030	0.81560	0.78870
8	1.00000	1.00000	1.00000	1.00000

Table 4.2.5 Parameterization of Dataset 3

Uniform		Chord Length		Centripetal		Exponential	
A	B	A	B	A	B	A	B
0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
0.20000	0.28571	0.20000	0.34455	0.20000	0.31479	0.20000	0.33266
0.40000	0.42857	0.40000	0.44818	0.40000	0.43826	0.40000	0.44422
0.60000	0.57143	0.60000	0.55182	0.60000	0.56174	0.60000	0.55578
0.80000	0.71429	0.80000	0.65545	0.80000	0.68521	0.80000	0.66734
1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000

1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000

A = Uniformly Spaced

B = Averaging

Table 4.2.6 Knot Vector Generation of Dataset 3

Parameterization methods / Number of data point	Uniform	Chord Length	Centripetal	Exponential
1	0.00000	0.00000	0.00000	0.00000
2	0.07692	0.11405	0.09718	0.10774
3	0.15385	0.17107	0.16589	0.16963
4	0.23077	0.26302	0.25315	0.26031
5	0.30769	0.27221	0.28074	0.27469
6	0.38462	0.38548	0.37759	0.38185
7	0.46154	0.45761	0.45487	0.45653
8	0.53846	0.51464	0.52358	0.51841
9	0.61538	0.67592	0.63915	0.66058
10	0.69231	0.73295	0.70786	0.72246
11	0.76923	0.84699	0.80504	0.83021
12	0.84615	0.89799	0.87003	0.88680
13	0.92308	0.94900	0.93501	0.94340
14	1.00000	1.00000	1.00000	1.00000

Table 4.2.7 Parameterization of Dataset 4

Uniform		Chord Length		Centripetal		Exponential	
A	B	A	B	A	B	A	B
0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
0.09091	0.15385	0.09091	0.18271	0.09091	0.17207	0.09091	0.17923
0.18182	0.23077	0.18182	0.23543	0.18182	0.23326	0.18182	0.23488
0.27273	0.30769	0.27273	0.30690	0.27273	0.30383	0.27273	0.30562
0.36364	0.38462	0.36364	0.37177	0.36364	0.37107	0.36364	0.37102
0.45455	0.46154	0.45455	0.45258	0.45455	0.45201	0.45455	0.45226
0.54545	0.53846	0.54545	0.54939	0.54545	0.53920	0.54545	0.54517
0.63636	0.61538	0.63636	0.64117	0.63636	0.62353	0.63636	0.63382
0.72727	0.69231	0.72727	0.75195	0.72727	0.71735	0.72727	0.73775
0.81818	0.76923	0.81818	0.82598	0.81818	0.79431	0.81818	0.81316
0.90909	0.84615	0.90909	0.89799	0.90909	0.87003	0.90909	0.88680
1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000

A = Uniformly Spaced

B = Averaging

Table 4.2.8 Knot Vector Generation of Dataset 4

Number of data point \ Parameterization methods	Uniform	Chord Length	Centripetal	Exponential
1	0.00000	0.00000	0.00000	0.00000
2	0.20000	0.36099	0.27999	0.32885
3	0.40000	0.41464	0.38793	0.40040
4	0.60000	0.56233	0.56702	0.56127
5	0.80000	0.73003	0.75786	0.73936
6	1.00000	1.00000	1.00000	1.00000

Table 4.2.9 Parameterization of Dataset 5

Uniform		Chord Length		Centripetal		Exponential	
A	B	A	B	A	B	A	B
0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
0.33333	0.40000	0.33333	0.44599	0.33333	0.41165	0.33333	0.43017
0.66667	0.60000	0.66667	0.56900	0.66667	0.57094	0.66667	0.56701
1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000

A = Uniformly Spaced

B = Averaging

Table 4.2.10 Knot Vector Generation of Dataset 5

As shown in Table 4.2.1, Table 4.2.3, Table 4.2.5, Table 4.2.7 and Table 4.2.9 which consist of the parameterization results, the first and last parameters are equal to 0 and 1 respectively for all the parameterization method used. Uniform method generated uniformly increment of the parameter values. For chord length, centripetal and exponential methods, the parameter values increase inconsistently because of the calculation which involve not only the summation of data points, but also consider the distance between the points.

Based on equation 14 which is the mathematical formula for uniform method, starting from second data points the parameter value is equal to the position of the data points divide by the summation of the number of the points. Therefore, the values are increasing equally. Equation 15 is the formula for the chord length method. This method takes the distance of the data points into consideration. After the distance is calculated, the result will be added with the previous value. Similar to the equation 14, the final result will be divided by the total data points. Equation 16 shows the calculation for centripetal method. The equation is different from equation 15 as there is an additional square root on the distance of the data points. Therefore, the sequence is proportional to the results of the calculation.

According to E.T.Y.Lee (1989), the equation of uniform, chord length and centripetal can be generalised into a general formula:

$$t_i - t_{i-1} = \frac{|Q_i - Q_{i-1}|^e}{\sum_{j=1}^{n} |Q_i - Q_{i-1}|^{e'}} \quad 1 \leq i \leq n \qquad (24)$$

If e = 0, then the equation will become uniform method. If e = 0.5, then the equation is for centripetal method. For chord length method, the value of e is 1. For exponential method, the e value is 0.8 according to (Haron, et al., 2012). Between the values 0 and 1, there are still many possibilities that can be used to modify the parameter values. In this research, the value of e will be focused on 0, 0.5, 0.8 and 1 which represents the four parameterization methods.

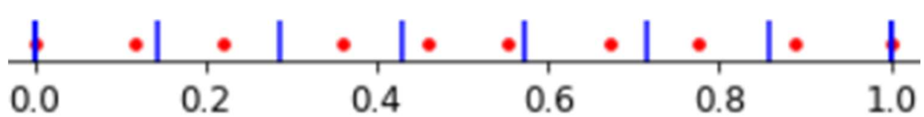
Table 4.2.2, Table 4.2.4, Table 4.2.6, Table 4.2.8 and Table 4.2.10 are the results for the knot vector generation. For the first and last four knots, the values are 0 and 1 respectively according to the equation 18 and equation 20. For the remaining knots or so called internal knots, the calculation for both knot vector generations are different. By using the uniformly spaced knot vector generation, the values of the knots are the same for all the data points. Equation 19 shows that the knot value is calculated using the position divided by the summation of control points and the degree of the curve. For averaging knot vector, the internal knots are computed by summing up the parameter values and then divided by the degree of the curve. Therefore, the values for the internal knots using the uniformly spaced knot vector are the same for four of the parameterization methods whereas the internal knots values using the averaging knot vector are different for each of the parameterization methods. This is because the averaging knot vector generation consider the parameter values obtained from previous process.

The next section is about the parameters and knots distribution for all the parameterization methods for each of the dataset. The control points generated are plotted together with the original data points and the errors for all parameterization methods using both knot vector generation are also included.

Dataset 1

(a)

(b)

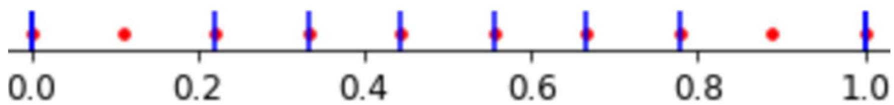


(c)

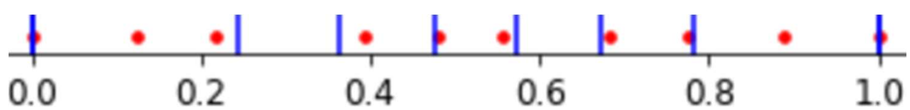
(d)

Figure 4.2.1 Parameter and Uniformly Spaced Knot Distribution of

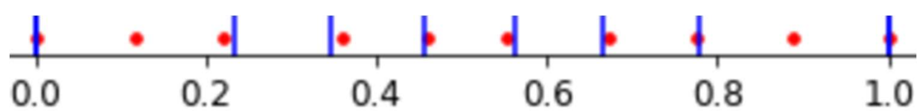
(a)Uniform, (b)Chord Length, (c)Centripetal and (d)Exponential



(a)



(b)



(c)

(d)

Figure 4.2.2 Parameter and Average Knot Distribution of

(a)Uniform, (b)Chord Length, (c)Centripetal and (d)Exponential

The red dots in the figure represent the parameter and the blue bar represent the knot value. Figure 4.2.1 shows the parameter and knot distribution of each of the parametrization methods using the uniformly spaced knot vector generation. The distribution is similar for all the methods. Each of the range consists of at least one of the parameters as the knot vectors are generated uniformly. Figure 4.2.2 is the parameter and knot distribution using averaging knot vector generation. For Figure 4.2.2 (a), the second and the ninth parameters fall in the beginning and last range whereas the remaining parameters fall on the knots generated. For other parametrization methods, not every range of the knots consists of parameters because the parameters were average and not equally distributed.

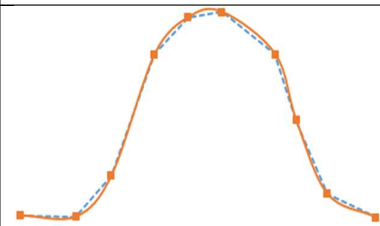
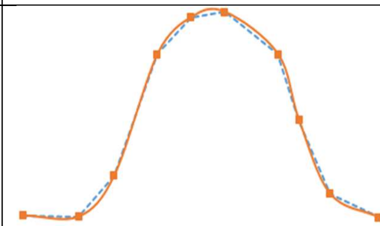
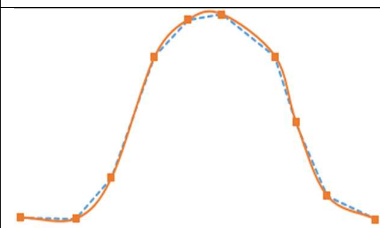
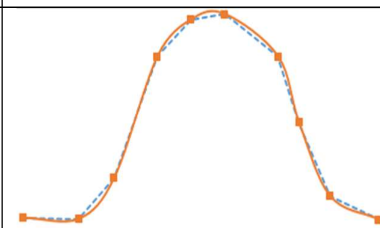
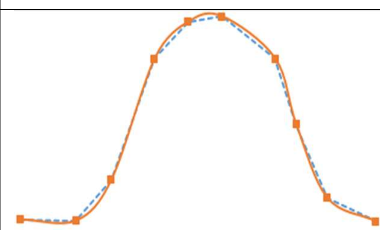
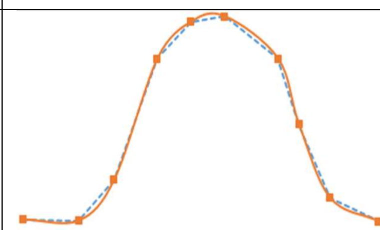
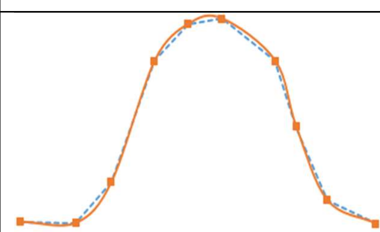
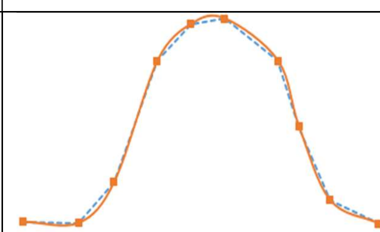
Parameterization Methods	Knot Vector Generation	
	Uniformly Spaced	Averaging
Uniform		
Centripetal		
Exponential		
Chord Length		

Table 4.2.11 Visualisation of Control Points and Data Points

Parameterization Methods	Knot Vector Generation	
	Uniformly Spaced	Averaging
Uniform	26.44794	16.78655
Centripetal	29.17968	17.14180
Exponential	31.23465	18.04786
Chord Length	32.48294	19.14408

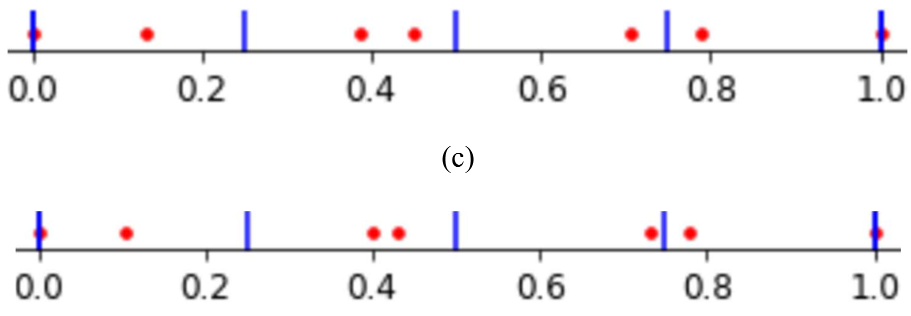
Table 4.2.12 Error of Four Parameterization Methods

Table 4.2.11 is the visualisation of control points and original data points for four parameterization methods using uniformly spaced and averaging knot vector generation. The blue dotted line is the control points and the red curve is the data points. Generally, most of the figures as shown in Table 4.2.11 are quite similar. Table 4.2.12 is the error results for each parameterization using different knot vector generation. From the table, for both knot vector generation, uniform parametrization had the least error which are 26.44794 and 16.78655 respectively. Chord length parametrization method had the highest error which are 32.48294 and 19.14408. Besides that, it is noticed that uniform parameterization performed better compared to other parameterization methods in dataset 1. The data points generated by averaging knot vector generation is better than uniformly spaced knot vector generation in dataset 1. Therefore, uniform parameterization with averaging knot vector generation is the best method in generating the data points for dataset 1.

Dataset 2

(a)

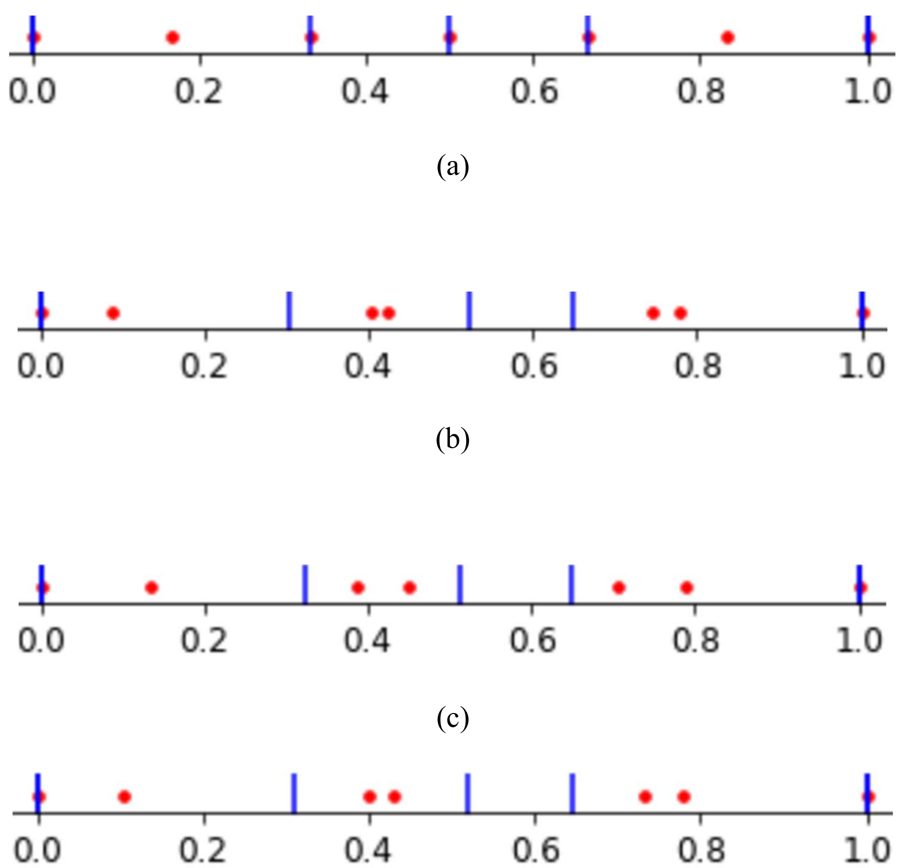
(b)



(c)

(d)

Figure 4.2.3 Parameter and Uniformly Spaced Knot Distribution of (a)Uniform, (b)Chord Length, (c)Centripetal and (d)Exponential



(a)

(b)

(c)

(d)

Figure 4.2.4 Parameter and Average Knot Distribution of (a)Uniform, (b)Chord Length, (c)Centripetal and (d)Exponential

Figure 4.2.3 represents the parameter and knot distribution of each of the parametrization methods using the uniformly spaced knot vector generation. For figure 4.2.3 (a), the parameters and knot vector are equally distributed whereas for figure 4.2.3 (b), (c) and (d), the figures show that the distance of the consecutive points at the starting and ending are far away and the distance of the parameter in the middle are near to each other. Figure 4.2.4 shows the parameter and knot distribution using averaging knot vector generation. For Figure 4.2.4 (a), the second and the sixth parameters fall in the beginning and last range whereas the remaining parameters fall on the knots generated. For other parametrization methods, it is similar to the uniformly spaced knot vector as the distance of the consecutive points are far from each other. The parameters which are in the same range have a closer distance.

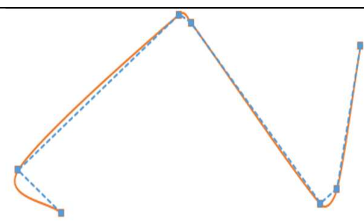
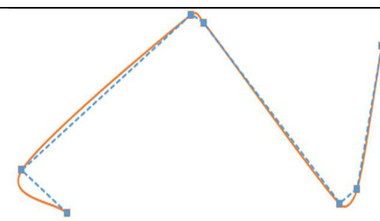
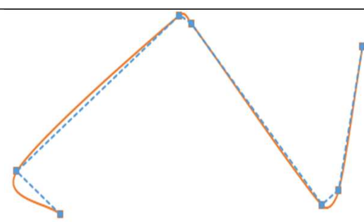
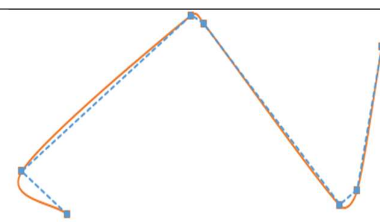
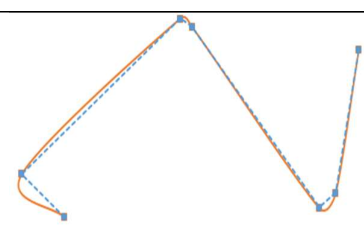
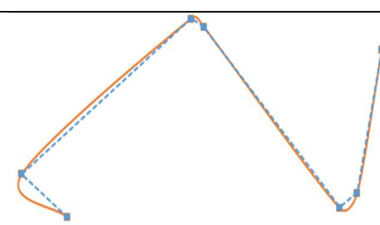
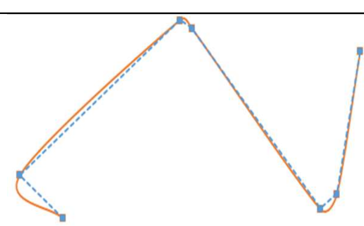
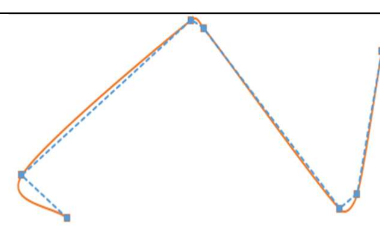
Parameterization Methods	Knot Vector Generation	
	Uniformly Spaced	Averaging
Uniform		
Centripetal		
Exponential		
Chord Length		

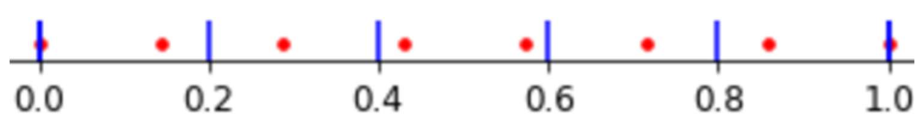
Table 4.2.13 Visualisation of Control Points and Data Points

Parameterization Methods	Knot Vector Generation	
	Uniformly Spaced	Averaging
Uniform	11.68430	12.14099
Centripetal	13.72661	12.66538
Exponential	13.54248	12.41007
Chord Length	13.05738	12.05698

Table 4.2.14 Error of Four Parameterization Methods

Table 4.2.13 shows the visualisation of control points and original data points for four parameterization methods using uniformly spaced and averaging knot vector generation. The curves constructed for each parameterization methods in the table are similar to each other. Table 4.2.14 shows the errors of the difference in distance for the generated data points and the original data points for various parametrization methods and knot vector generation. From the table, uniform parametrization had the least error which is 11.68430 compared to other methods for uniformly spaced knot vector generation. Besides, chord length parameterization had the least error which is 12.05698 for averaging knot vector generation. Basically, uniform parameterization performed better compared to other parameterization methods using uniformly spaced knot vector whereas chord length parameterization performed better using averaging knot vector in dataset 2. Overall, the data points generated using averaging knot vector generation performed better compared to uniformly spaced knot vector generation for dataset 2. Therefore, uniform parameterization with averaging knot vector generation is the best method in generating the data points for dataset 2 as it had the least error result.

Dataset 3



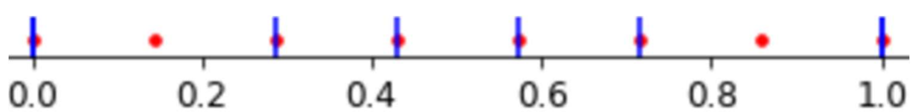
(a)

(b)

(c)

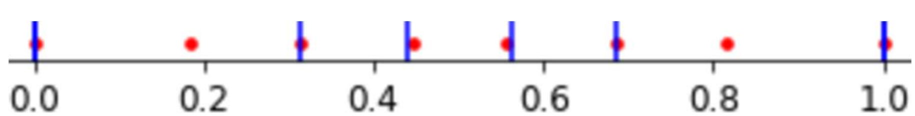
(d)

Figure 4.2.5 Parameter and Uniformly Spaced Knot Distribution of (a)Uniform, (b)Chord Length, (c)Centripetal and (d)Exponential



(a)

(b)



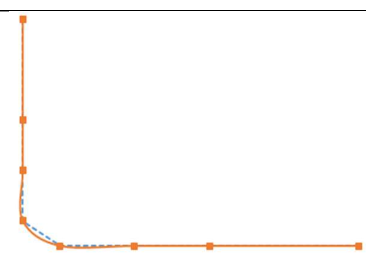
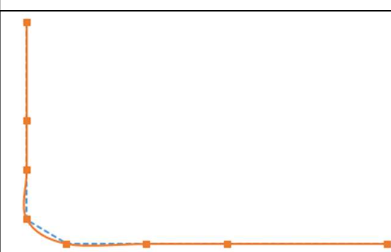
(c)

(d)

Figure 4.2.6 Parameter and Average Knot Distribution of

(a)Uniform, (b)Chord Length, (c)Centripetal and (d)Exponential

Figure 4.2.5 presents the parameter and knot distribution of each of the parametrization methods using the uniformly spaced knot vector generation. For figure 4.2.5 (a), the parameters and knot vector are equally distributed whereas for figure 4.2.5 (b), (c) and (d), the figures show that the distance of the consecutive points at the starting and ending are far away. The distance of the second parameter until the seventh parameter is slightly considered as uniform. Figure 4.2.6 shows the parameter and knot distribution using averaging knot vector generation. For Figure 4.2.6 (a), the second and the seventh parameters fall in the beginning and last range whereas the remaining parameters fall on the knots generated. For other parametrization methods, the distribution of the parameters and knots value beside the starting and ending are near to each other and far away from the starting and ending points.

Parameterization Methods	Knot Vector Generation	
	Uniformly Spaced	Averaging
Uniform		

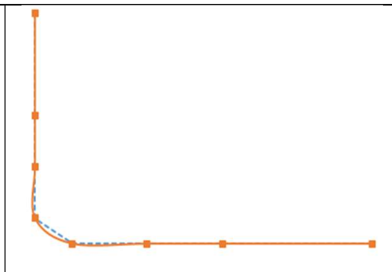
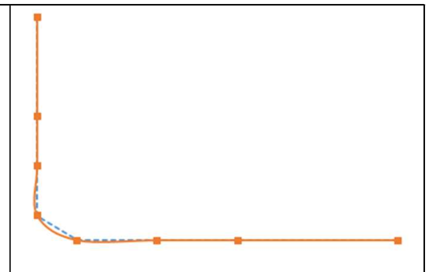
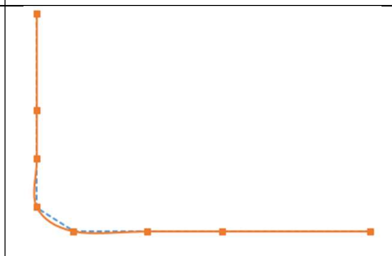
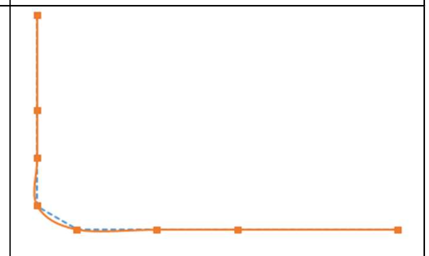
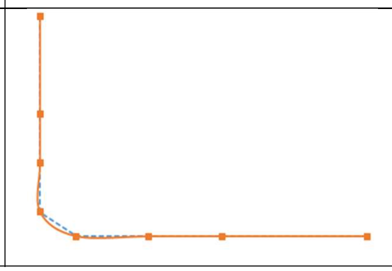
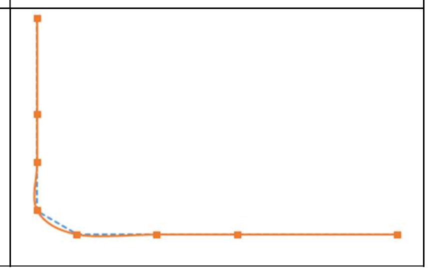
Centripetal		
Exponential		
Chord Length		

Table 4.2.15 Visualisation of Control Points and Data Points

Parameterization Methods	Knot Vector Generation	
	Uniformly Spaced	Averaging
Uniform	9.23759	8.08561
Centripetal	12.81471	9.93196
Exponential	14.82720	11.22931
Chord Length	15.87321	12.11663

Table 4.2.16 Error of Four Parameterization Methods

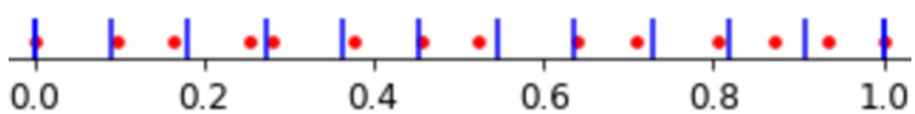
Table 4.2.15 presents the curves constructed based on control points and original data points for four parameterization methods using uniformly spaced and averaging knot vector generation. Generally, the figures shown in Table 4.2.15 had similar shape. Table 4.2.16 presents the error results for each parameterization using uniformly spaced and averaging knot vector generation. Based on the table, uniform parametrization had the least error which are 9.23759 and 8.08561 for each knot vector generation. Chord length parametrization method had the highest error which are 15.87321 and 12.11663.

Uniform parameterization had better performance compared to other parameterization methods in dataset 3. The data points generated by averaging knot vector generation is better than uniformly spaced knot vector generation. Therefore, uniform parameterization with averaging knot vector generation provides a better result in generating the data points for dataset 3.

Dataset 4

(a)

(b)

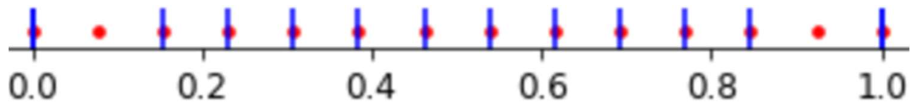


(c)

(d)

Figure 4.2.7 Parameter and Uniformly Spaced Knot Distribution of

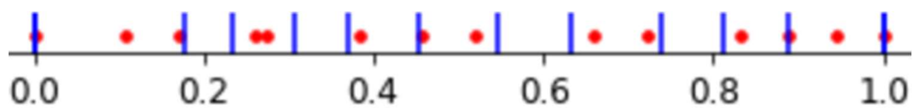
(a)Uniform, (b)Chord Length, (c)Centripetal and (d)Exponential



(a)

(b)

(c)



(d)

Figure 4.2.8 Parameter and Average Knot Distribution of

(a) Uniform, (b)Chord Length, (c)Centripetal and (d)Exponential

Figure 4.2.7 presents the parameter and knot distribution of each of the parametrization methods using the uniformly spaced knot vector generation. For figure 4.2.7 (a), the parameters values and knot vector values are equally distributed whereas for figure 4.2.7 (b), (c) and (d), the fourth and seventh of the knots range having less or none parameters. Figure 4.2.8 shows the parameter and knot distribution using averaging knot vector generation. For Figure 4.2.8 (a), the second and the thirteenth parameters fall in the beginning and last range whereas the remaining parameters fall on the knots generated. For figure 4.2.8 (b), (c) and (d), the distributions are similar to the uniformly spaced knots which were distributed inconsistently.

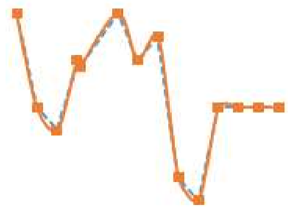
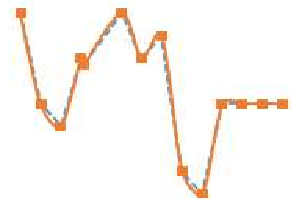
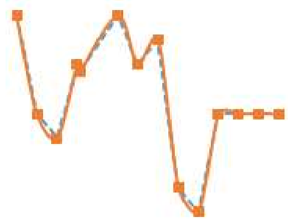
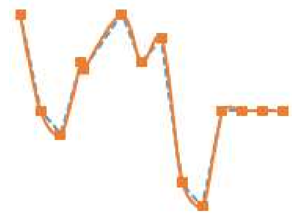
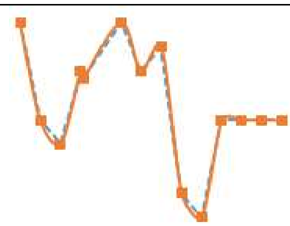
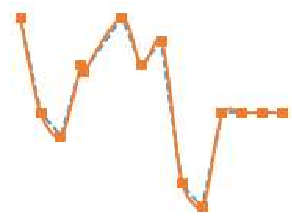
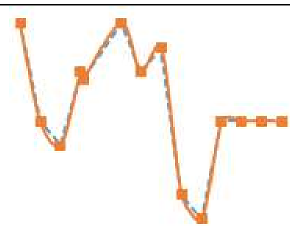
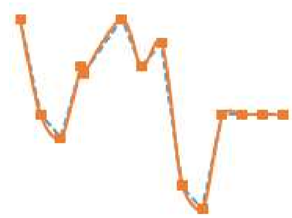
Parameterization Methods	Knot Vector Generation	
	Uniformly Spaced	Averaging
Uniform		
Centripetal		
Exponential		
Chord Length		

Table 4.2.17 Visualisation of Control Points and Data Points

Parameterization Methods	Knot Vector Generation	
	Uniformly Spaced	Averaging
Uniform	134.90471	74.75813
Centripetal	129.22529	99.07210
Exponential	145.38275	109.98126
Chord Length	158.32117	120.90907

Table 4.2.18 Error of Four Parameterization Methods

Table 4.2.17 shows the visualisation of control points and original data points for four parameterization methods using uniformly spaced and averaging knot vector generation. The shape of the graph constructed for each parameterization methods in the table are similar to each other. Table 4.2.18 shows the errors of the difference in distance for the generated data points and the original data points for various parametrization methods and knot vector generation. As shown in the table, centripetal parameterization had the least error which is 129.22529 using uniformly spaced knot vector generation. Besides, uniform parameterization had the least error which is 74.75813 for averaging knot vector generation. Generally, centripetal parameterization performed better compared to other parameterization methods using uniformly spaced knot vector whereas uniform parameterization performed better using averaging knot vector in dataset 4. Overall, the data points generated using averaging knot vector generation performed better compared to uniformly spaced knot vector generation for dataset 4. Therefore, uniform parameterization with averaging knot vector generation is the best method in generating the data points for dataset 4.

Dataset 5

(a)

(b)

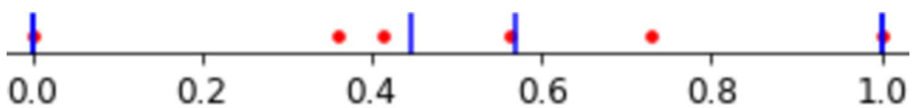
(c)

(d)

Figure 4.2.9 Parameter and Uniformly Spaced Knot Distribution of

(a)Uniform, (b)Chord Length, (c)Centripetal and (d)Exponential

(a)



(b)

(c)

(d)

Figure 4.2.10 Parameter and Average Knot Distribution of

(a)Uniform, (b)Chord Length, (c)Centripetal and (d)Exponential

Figure 4.2.9 shows the parameter and knot distribution of each of the parametrization methods using the uniformly spaced knot vector generation. For figure 4.2.9 (a), the parameters values and knot vector values are equally distributed and all the range contain at least one parameter. For figure 4.2.9 (b), (c) and (d), the parameters tend to gather near the middle range. The first knot range for figure 4.2.9 (b) and (d) are empty. Figure 4.2.10 presents the parameter and knot distribution using averaging knot vector generation. For Figure 4.2.10 (a), the second and the fourth parameters fall in the beginning and last range whereas the remaining parameters fall on the knots generated. For figure 4.2.10 (b), (c) and (d), the distributions are similar to the uniformly spaced knots which were distributed inconsistently.

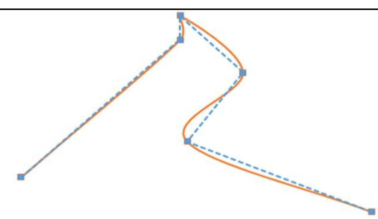
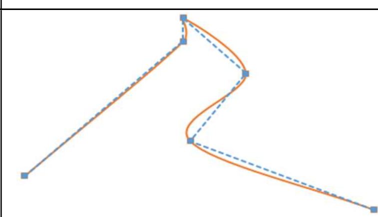
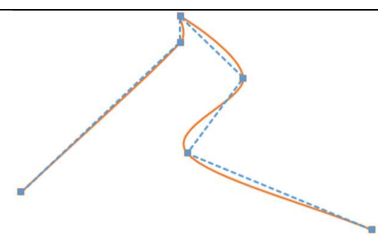
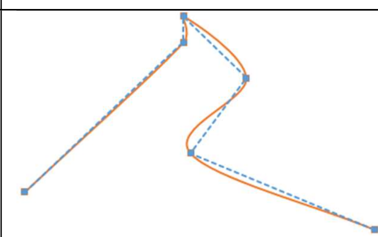
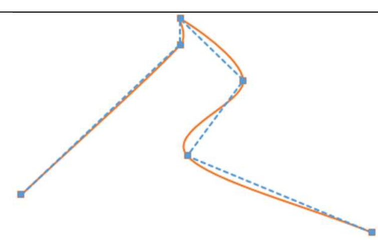
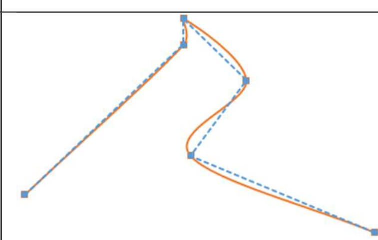
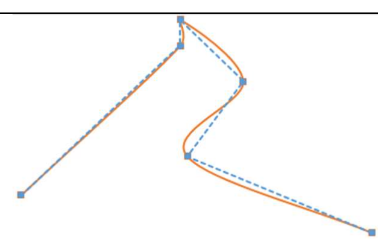
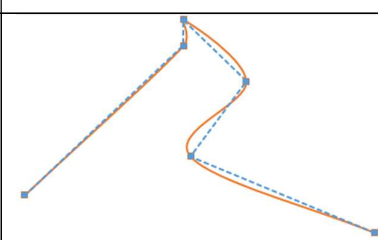
Parameterization Methods	Knot Vector Generation	
	Uniformly Spaced	Averaging
Uniform		
Centripetal		
Exponential		
Chord Length		

Table 4.2.19 Visualisation of Control Points and Data Points

Parameterization Methods	Knot Vector Generation	
	Uniformly Spaced	Averaging
Uniform	9.26437	11.41900
Centripetal	13.96101	15.89795
Exponential	17.35259	19.08679
Chord Length	19.31934	21.06089

Table 4.2.20 Error of Four Parameterization Methods

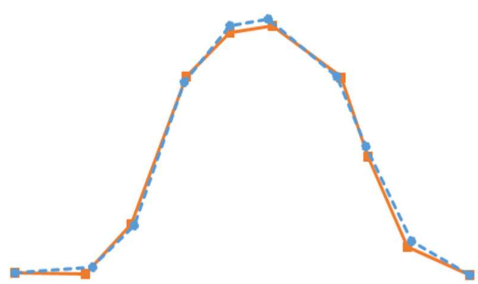
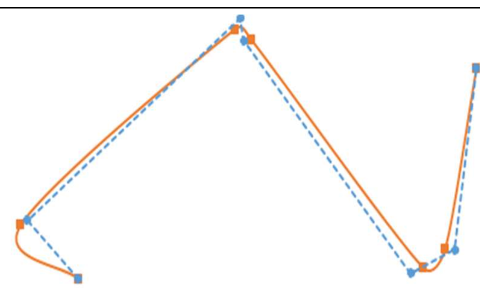
Table 4.2.19 is the visualisation of control points and original data points for four parameterization methods using uniformly spaced and averaging knot vector generation. Generally, the figures shown in Table 4.2.19 are quite similar. Table 4.2.20 is the error results for each parameterization using uniformly spaced and averaging knot vector generation. From the table, uniform parametrization had the least error which are 9.26437 and 11.41900 for each knot vector generation. Chord length parametrization method had the highest error which are 19.31934 and 21.06089. Uniform parameterization performed better compared to other parameterization methods in dataset 5. The data points generated by uniformly spaced knot vector generation is better than averaging knot vector generation. Therefore, uniform parameterization with uniformly spaced knot vector generation provides a better result in generating the data points for dataset 5.

Dataset	Parameterization Methods	Knot Vector Generation					
		Uniformly Spaced			Averaging		
		Original	GA	DE	Original	GA	DE
1	Uniform	26.44794	25.94418	25.19507	16.78655	16.60393	15.86650
	Centripetal	29.17968	28.24147	27.92184	17.14180	16.92640	16.16987
	Exponential	31.23465	30.13695	29.80171	18.04786	17.82270	16.95304
	Chord Length	32.48294	31.39069	31.13509	19.14408	18.95485	18.13135
2	Uniform	11.68430	11.56139	11.12791	12.14099	12.02465	11.59085
	Centripetal	13.72661	13.62629	13.27011	12.66538	12.57396	12.25383
	Exponential	13.54248	13.44002	13.07648	12.41007	12.32746	12.06007
	Chord Length	13.05738	12.96363	12.65604	12.05698	11.98050	11.74178
3	Uniform	9.23759	8.28460	8.53788	8.08561	6.75487	7.44312
	Centripetal	12.81471	10.30345	11.91829	9.93196	8.27010	9.20684
	Exponential	14.82720	12.61257	13.95337	11.22931	9.18694	10.58145
	Chord Length	15.87321	12.88129	15.03274	12.11663	8.86969	11.32366
4	Uniform	134.90471	134.52617	132.86375	74.75813	74.50104	73.52093
	Centripetal	129.22529	128.90828	127.44849	99.07210	98.81697	97.56874
	Exponential	145.38275	145.10842	143.75472	109.98126	109.10996	108.61979
	Chord Length	158.32117	158.06627	120.19056	120.90907	156.89676	119.84287
5	Uniform	9.26437	8.79423	8.83480	11.41900	11.05764	11.08478
	Centripetal	13.96101	12.22011	13.44314	15.89795	14.17900	15.50208
	Exponential	17.35259	15.34906	16.84237	19.08679	15.06754	18.58727
	Chord Length	19.31934	16.72663	18.81654	21.06089	18.71501	20.56162

Table 4.2.21 Error of Datasets

Furthermore, the error results obtained after optimization are analyse and discussed. Table 4.2.23 is the errors for all datasets with respective parameterization, knot vector generation and optimization methods. The results shown in the table is the original error of each datasets and average error of GA and DE optimization on each datasets for various parameterization methods and knot vector generation methods. The complete results for all optimization of datasets are located in Appendix A.

As shown in the tables, with the implementation of optimization, the errors obtained are decreased. For dataset 1, the table shows that DE further optimized the error of the original error to 15.86650 for uniform parameterization method with averaging knot vector generation. Moreover, for dataset 2, the original error is best optimized by DE for uniform parameterization with uniformly spaced knot vector generation where the optimized error is 11.12791. From the table, GA had a better optimized error which is 6.75487 for uniform parameterization with averaging knot vector for dataset 3. For the fourth dataset, DE further optimised the original error to 73.52093 for uniform parameterization with averaging knot vector. Lastly, for dataset 5, uniform parameterization with uniformly spaced knot vector has the least optimized error, 8.79423 by using GA optimization. GA performed faster than DE because in GA, the replacement was done once per generation whereas in DE, there was update for previous chromosomes and iterated through the population to compare the trial vector. Overall, DE performed better having lower error results but sometimes GA performed better when the data points are lesser. The operations in both optimizations depend on the probability given.

Dataset	Graph	Explanation
1		DE with uniform parameterization method and averaging knot vector generation
2		DE with uniform parameterization method and uniformly spaced knot vector generation

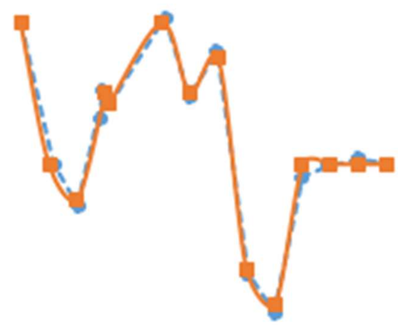
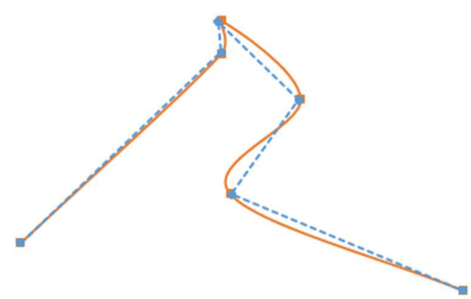
3		GA with uniform parameterization method and averaging knot vector generation
4		GA with uniform parameterization method and averaging knot vector generation
5		GA with uniform parameterization method and averaging knot vector generation

Table 4.2.22 Visualisation of Control Points and Data Points

Table 4.2.22 is the visualisation of the optimized control points and original data points with the explanation which stated the best parameterization, knot vector generation and optimization methods. The graphs are plotted according to the minimum results which is placed in the Appendix A. As this research is about interpolation curve, the number of control points and number of data points are similar so this lead to similar graph.

Based on the table, uniform parameterization is the best among four parameterization methods for all the datasets. Averaging knot vector generation had better performance for most of the datasets except dataset 2. With the input data points, parameters are generated through parameterization method. After that, the parameters are used to calculate the knot vector. From equation 24, uniform parameterization is formed if e = 0 and the parameters are uniformly distributed. For averaging knot vector generation, the knot vector is generated by averaging the summation of the knots according to the parameters. For most of the datasets, GA provided a better minimum results compared to DE.

CHAPTER 5 CONCLUSION

5.1 Conclusion

This research is about the B-spline curve fitting with different parametrization methods. Interpolation of B-spline curve is generated based on the datasets given. By using the input of dataset given, four parameterizations methods are done to obtain the parameter values for each of the data points. After that uniformly spaced knot vector and averaging knot vector are calculated with the use of parameters obtained previously. After control points are obtained, GA and DE optimization are done to optimize the error between generated data points and original data points. Analysis had been done to compare the four parameterization methods which are uniform, chord length, centripetal and exponential methods. Both the knot vector generation are analysed to identify the differences between them. GA and DE are analysed and discussed.

The research is done using various datasets with different properties. Based on the analysis, curves generated using the uniform parameterization had better results for most of the datasets. The parameters generated by uniform parameterization are uniformly generated. Overall, averaging knot vector generation produced better curves compared to uniformly spaced knot vector generation. The knot vectors are averagely calculated with the input of previous parameters results. Although DE had lower optimized error results for most of the datasets, GA provided a better minimum optimized error results.

5.2 Future Work

In the near future, more parameterization methods could be implemented so that able to analyse the different results obtained. In this research, x coordinate and y coordinate were used and the z coordinate is identical. So, z coordinate could be different to implement later. Furthermore, modification in the operation for GA and DE could be done to further improve the optimization. Next, combination of the optimization method could be implemented. GA and DE could merge together to further optimise the error results.

BIBLIOGRAPHY

Ammad, M. & Ramli, A., 2019. *Cubic B-Spline Curve Interpolation with Arbitrary Derivatives on its Data Points.* Adelaide, 2019 23rd International Conference in Information Visualization – Part II.

Chantakamo, A. & Dejdumrong, N., 2013. *Conversion of Rational Bezier Curves into Non-rational Bezier Curves Using Progressive Iterative Approximation.* Macau, 2013 10th International Conference Computer Graphics, Imaging and Visualization.

Chantakamo, A. & Dejdumrong, N., 2014. *The Analysis of PIA Algorithm Efficiency in Converting Rational Bézier Curves into Corresponding Non-rational Curves.* Singapore, 2014 11th International Conference on Computer Graphics, Imaging and Visualization.

Chi, J., Zhang, Y. & Zhang, C., 2008. *Optimized Geometric Hermite Curve Based on Curve Length Minimization.* Sydney, IEEE 8th International Conference on Computer and Information Technology Workshops.

Ching-Shoei, C., 2015. *The medial axis transform of the region defined by Hermite curve tangent to unit circle.* Cambridge, 2015 10th International Conference on Computer Science & Education (ICCSE).

Chong, E. K. P. & Zak, S. H., 2001. *An Introduction to Optimization.* 2nd ed. United States of America: Wiley-Interscience.

Chouychai, B., 2015. *Point cloud data reduction with tangent function method for nurbs curve and surface fitting.* Chonburi, 2015 7th International Conference on Knowledge and Smart Technology (KST).

Dung, V. T. & Tjahjowidodo, T., 2017. A direct method to solve optimal knots of B-spline curves: An application for non-uniform B-spline curves fitting. *PLOS ONE,* 12(3).

E.T.Y.Lee, 1989. Choosing nodes in parametric curve interpolation. *Computer-Aided Design,* 21(6), pp. 363-370.

Gálvez, A. & Iglesias, A., 2013. *An Electromagnetism-Based Global Optimization Approach for Polynomial Bezier Curve Parameterization of Noisy Data Points.* Yokohama, 2013 International Conference on Cyberworlds.

Gálvez, A., Iglesias, A. & Puig-Pey, J., 2012. Iterative two-step genetic-algorithm-based method for efficient polynomial B-spline surface reconstruction. *Information Sciences,* 182(1), pp. 56-76.

Haron, H. et al., 2012. Parameterization Method on B-Spline Curve. *Mathematical Problems in Engineering,* Volume 2012, pp. 1-22.

Hasan, Z. A., Yahya, Z. R., Rusdi, N. A. & Roslan, N., 2018. *Curve Reconstruction In Different Cubic Functions Using Differential Evolution.* s.l., MATEC Web of Conferences.

Hasegawa, A. Y., Jr, R. S. U. R. & Tsuzuki, M. S. G., 2013. Bézier Curve Fitting With A Parallel Differential Evolution Algorithm. *11th IFAC Workshop on Intelligent Manufacturing Systems,* 46(7), pp. 233-238.

Iglesias, A., Gálvez, A. & Collantes, M., 2015. *Bat Algorithm for Curve Parameterization in Data Fitting with Polynomial Bézier Curves.* Visby, 2015 International Conference on Cyberworlds (CW).

Ismail, O., 2015. A matrix Method for Interval Hermite Curve. *International Journal of Video and Image Processing and Network Security IJVIPNS-IJENS,* 15(3), pp. 7-11.

Kineri, Y., Wang, M., Lin, H. & Maekawa, T., 2012. B-spline surface fitting by iterative geometric interpolation/approximation algorithms. *Computer-Aided Design,* 44(7), pp. 697-708.

Kumar, G., Kalra, P. & Dhande, S., 2003. *Parameter optimization for B-spline curve fitting using genetic algorithms.* Canberra, The 2003 Congress on Evolutionary Computation, 2003. CEC '03..

Lee, D.-U., Cheung, R., Luk, W. & Villasenor, J., 2008. Hardware Implementation Trade-Offs of Polynomial Approximations and Interpolations. *IEEE Transactions on Computers,* 57(5), pp. 686-701.

Lim, S. P. & Haron, H., 2013. *Performance comparison of Genetic Algorithm, Differential Evolution and Particle Swarm Optimization towards benchmark functions.* Kuching, 2013 IEEE Conference on Open Systems (ICOS).

Lim, S. P. & Haron, H., 2014. *Applying NURBS Surfaces Approximation with Different Parameterization Methods on CKSOM Model Closed Surfaces Data.* Cherbourg, Springer, Cham.

Lim, S. P., Hoon, H. & Ong, C. H., 2018. *Performance Comparison of Different Operation Techniques in Genetic Algorithm towards Benchmark Functions.* Penang, 2018 8th IEEE International Conference on Control System, Computing and Engineering (ICCSCE).

Lim, S. P., Hoon, H. & Ong, C. H., 2018. *Performance Comparison of Different Operation Techniques in Genetic Algorithm towards Benchmark Functions.* Penang, 2018 8th IEEE International Conference on Control System, Computing and Engineering (ICCSCE).

Li, Q., Li, J.-P. & Chen, L., 2018. *A Bezier Curve-Based Font Generation Algorithm for Character Fonts.* Exeter, 2018 IEEE 20th International Conference on High Performance Computing and Communications; IEEE 16th International Conference on Smart City; IEEE 4th International Conference on Data Science and Systems (HPCC/SmartCity/DSS).

Liu, Y., 2003. *Effect of Knot Vectors on B-Spline Curves and Surfaces,* New York: Department of Mechanical Engineering, State University of New York.

Liu, Z., Cohen, F. & Zhang, Z., 2014. *Fitting B-splines to scattered data new and old parameterization.* Marrakech, 2014 International Conference on Multimedia Computing and Systems (ICMCS).

Marinić-Kragić, I., Ćurković, M. & Vučina, D., 2018. Adaptive re-parameterization based on arbitrary scalar fields for shape optimization and surface fitting. *Engineering Applications of Artificial Intelligence,* Volume 67, pp. 39-51.

Mohanty, F., Satpathy, S. & Pattnaik, P. K., 2015. *Removal of fixed impulse noise from digital images using Bezier curve based interpolation.* Bhubaneswar, 2015 IEEE

International Conference on Computer Graphics, Vision and Information Security (CGVIS).

Mortenson, M. E., 2006. *Geometric Modeling.* 3rd ed. California: Industrial Press Inc..

Ostaszewski, M. & Kuzmierowski, T., 2015. *Frequency fitting algorithm of control signals based on Hermite curves.* Szilvasvarad, Proceedings of the 2015 16th International Carpathian Control Conference (ICCC).

Pandunata, P. & Shamsuddin, S. M. H., 2010. *Differential Evolution Optimization for Bezier Curve Fitting.* Sydney, 2010 Seventh International Conference on Computer Graphics, Imaging and Visualization.

Ravari, A. N. & Hamid D. Taghirad, 2016. Reconstruction of B-spline curves and surfaces by adaptive group testing. *Computer-Aided Design,* Volume 74, pp. 32-44.

Ravichandran, K. & Arulchelvan, S., 2017. *The Curve Fitting Model Analyzed the Survey of Crime News Awareness in India.* Tindivanam, 2017 Second International Conference on Recent Trends and Challenges in Computational Models (ICRTCCM).

Roslan, N. & Yahya, Z. R., 2016. *Parameterization method on cubic Bézier curve fitting using differential evolution.* s.l., AIP Publishing.

Shene, C.-K., 2014. *Knot Vector Generation.* [Online] Available at: https://pages.mtu.edu/~shene/COURSES/cs3621/NOTES/INT-APP/PARA-knot-generation.html
[Accessed 10 November 2019].

Sommerville, I., 2011. *Software Engineering.* 9th ed. USA: Pearson Education, Inc..

Song, L., Li, F. & Li, Z., 2015. *A Bezier curve based on path tracking in Computer Generated Force.* Zhangjiajie, 2015 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD).

Suszyński, Z. & Świta, R., 2017. *Image approximation using B-spline surfaces.* Warsaw, 2017 21st European Microelectronics and Packaging Conference (EMPC) & Exhibition.

Ueda, E. K., Tsuzuki, M. S. G. & Barari, A., 2018. *Piecewise Bézier Curve Fitting of a Point Cloud Boundary by Simulated Annealing.* São Paulo, 2018 13th IEEE International Conference on Industry Applications (INDUSCON).

Uhlmann, V., Fageot, J. & Unser, M., 2016. Hermite Snakes With Control of Tangents. *IEEE Transactions on Image Processing,* 25(6).

Wan, Y. & Yin, S., 2014. *Three-dimensional curve fitting based on cubic B-spline interpolation curve.* Dalian, 2014 7th International Congress on Image and Signal Processing.

Wu, Q. & Tao, S., 2009. *Shape Modification of B-Spline Curves via Constrained Optimization for Multi-target Points.* Los Angeles, 2009 WRI World Congress on Computer Science and Information Engineering.

Yang, C., Yu, J. & Zhu, M., 2017. *High-precision resistance calibrator error compensation method research based on fitting algorithm.* Siem Reap, 2017 12th IEEE Conference on Industrial Electronics and Applications (ICIEA).

Yang, J., Liu, F., Tao, X. & Wang, X., 2012. *The application of evolutionary algorithm in B-spline curve fitting.* Sichuan, 2012 9th International Conference on Fuzzy Systems and Knowledge Discovery.

Yoshimoto, F., Moriyama, M. & Harada, T., 1999. *Automatic knot placement by a genetic algorithm for data fitting with a spline.* Aizu-Wakamatsu, Proceedings Shape Modeling International '99. International Conference on Shape Modeling and Applications.

APPENDIX A: DATASET AND TABLES

Appendix A shows the datasets used in this research. Each of the data points in 3D which consists of x, y and z coordinates. Tables that present the error and performance results for each datasets with knot vector generation after optimization are shown.

Data Set 1

i	x	y	z
1	0.58072	2.08688	0
2	3.50755	2.05734	0
3	5.36585	3.24051	0
4	7.64228	6.74273	0
5	9.40767	7.79716	0
6	11.1963	7.94949	0
7	13.9837	6.73456	0
8	15.08595	4.83752	0
9	16.7247	2.71041	0
10	19.2799	2.03701	0

Data Set 2

i	x	y	z
1	1.83761	1.33271	0
2	0.726496	2.6504	0
3	4.82906	7.40482	0
4	5.12821	7.15185	0
5	8.4188	1.62069	0
6	8.84615	2.08095	0
7	9.44444	6.46124	0

Data Set 3

i	x	y	z
1	0	9	0
2	0	5	0
3	0	3	0
4	0	1	0
5	1	0	0
6	3	0	0
7	5	0	0
8	9	0	0

Data Set 4

i	x	y	z
1	32	14	0
2	34	10	0
3	36	9	0
4	38	12	0
5	38.2	11.7	0
6	42	14	0
7	44	12	0
8	46	13	0
9	48	7	0
10	50	6	0
11	52	10	0
12	54	10	0
13	56	10	0
14	58	10	0

Data Set 5

i	x	y	Z
1	1.40152	2.75472	0
2	4.84848	8.41509	0
3	4.84848	9.4	0
4	6.21212	7.0566	0
5	5	4.22642	0
6	9.01515	1.32075	0

Data Set 6

i	x	y	Z
1	-64.9885	-3.47E-15	10
2	-64.9885	1.17851	10
3	-64.9885	4.71172	10
4	-64.9885	8.23099	10
5	-64.9885	11.7234	10
6	-64.9885	15.177	10
7	-64.9885	18.5811	10
8	-64.9885	21.9267	10
9	-64.8629	25.1835	10
10	-64.3195	27.1986	10
11	-63.7437	28.4339	10
12	-63.0155	29.5658	10
13	-62.1502	30.5815	10
14	-61.1627	31.4684	10
15	-60.0687	32.2155	10
16	-58.8839	32.8123	10
17	-57.6241	33.2485	10
18	-56.3068	33.5147	10
19	-54.9499	33.602	10
20	-51.8667	33.602	10
21	-48.783	33.602	10
22	-45.6989	33.602	10
23	-42.6143	33.602	10
24	-39.5292	33.602	10
25	-36.4438	33.602	10
26	-33.3579	33.602	10
27	-30.2717	33.602	10
28	-27.1852	33.602	10
29	-24.0983	33.602	10
30	-21.0111	33.602	10
31	-17.9236	33.602	10
32	-14.8359	33.602	10
33	-11.7479	33.602	10
34	-8.65971	33.602	10
35	-5.57133	33.602	10
36	-2.48278	33.602	10
37	0.605909	33.602	10
38	3.69473	33.602	10
39	6.78365	33.602	10
40	9.87265	33.602	10
41	12.9617	33.602	10
42	16.0508	33.602	10
43	19.1399	33.602	10
44	22.229	33.602	10
45	25.3181	33.602	10
46	28.4071	33.602	10
47	31.496	33.602	10
48	34.5849	33.602	10
49	37.6736	33.602	10
50	40.7622	33.602	10
51	43.8506	33.602	10
52	46.9388	33.602	10
53	50.0218	33.5576	10
54	52.0115	33.1629	10
55	53.2678	32.6832	10
56	54.4462	32.0405	10
57	55.5291	31.2451	10
58	56.4991	30.3076	10
59	57.3395	29.2401	10
60	58.0337	28.0564	10
61	58.5653	26.7706	10
62	58.9177	25.3996	10
63	59.074	23.9625	10
64	59.0805	20.619	10
65	59.0805	17.2107	10
66	59.0805	13.7467	10
67	59.0805	10.2369	10
68	59.0805	6.69301	10
69	59.0805	3.12756	10

Continue at the right

Dataset 1

Dataset 1																
Experiment	Error															
	Uniform				Centripetal				Exponential				Chord Length			
	Equal Distance		Averaging		Equal Distance		Averaging		Equal Distance		Averaging		Equal Distance		Averaging	
	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE
1	25.88634	25.05133	16.61476	15.86615	28.92872	27.93540	16.89466	16.28444	30.23947	29.52103	17.82859	16.93893	31.34075	30.61123	18.93454	18.10486
2	25.88634	25.44138	16.59216	15.79603	28.92872	28.13460	16.92390	16.33995	30.23947	29.62969	17.82859	16.81152	31.39952	31.26664	18.94154	18.11691
3	25.85918	25.11895	16.59621	15.97457	28.34185	27.71719	16.92390	16.09146	30.07559	29.87813	17.81467	16.93776	31.39952	31.03175	18.94154	18.19397
4	25.85918	25.05824	16.60706	15.99960	28.34185	27.77625	16.92899	16.27653	30.07559	30.21313	17.81467	16.86707	31.36301	31.11709	18.97220	18.13998
5	26.23769	25.22286	16.60706	15.88535	27.94844	28.29606	16.92899	16.20854	30.07559	29.85140	17.83622	16.95622	31.36301	31.44586	18.97220	18.31550
6	26.23769	25.02175	16.58664	15.86524	27.94844	27.79060	16.92899	16.29968	30.23898	30.14987	17.83622	16.78276	31.39872	30.97183	18.97220	18.18887
7	25.87526	25.25101	16.59133	15.89069	28.00850	27.84091	16.91779	15.74361	30.23898	29.73804	17.80967	17.15998	31.39872	31.07989	18.95858	17.91901
8	25.87526	25.17097	16.61156	15.72403	28.00850	28.03383	16.91779	16.19993	30.06556	30.07092	17.80967	17.15845	31.39872	31.11531	18.95858	18.05351
9	25.86243	25.13912	16.61156	15.85366	27.97985	27.82772	16.94951	16.17727	30.06556	29.20080	17.80967	17.04781	31.42245	31.31154	18.94857	17.96981
10	25.86243	25.47506	16.62099	15.80969	27.97985	27.86581	16.94951	16.07733	30.05469	29.76410	17.83905	16.86994	31.42245	31.39971	18.94857	18.31103
Average	25.94418	25.19507	16.60393	15.86650	28.24147	27.92184	16.92640	16.16987	30.13695	29.80171	17.82270	16.95304	31.39069	31.13509	18.95485	18.13135
Minimum	25.85918	25.02175	16.58664	15.72403	27.94844	27.71719	16.89466	15.74361	30.05469	29.20080	17.80967	16.78276	31.34075	30.61123	18.93454	17.91901
Maximum	26.23769	25.47506	16.62099	15.99960	28.92872	28.29606	16.94951	16.33995	30.23947	30.21313	17.83905	17.15998	31.42245	31.44586	18.97220	18.31550

Experiment	CPU Time															
	Uniform				Centripetal				Exponential				Chord Length			
	Equal Distance		Averaging		Equal Distance		Averaging		Equal Distance		Averaging		Equal Distance		Averaging	
	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE
1	0.710	6.756	0.799	6.987	0.417	6.914	0.419	6.808	0.442	7.099	0.415	6.942	0.431	6.722	0.417	6.903
2	0.698	6.770	0.753	6.976	0.417	6.901	0.417	6.834	0.428	7.091	0.410	6.939	0.433	6.837	0.407	6.892
3	0.686	6.795	0.740	6.886	0.420	6.904	0.449	6.821	0.429	7.102	0.412	6.925	0.440	7.092	0.406	6.865
4	0.761	6.783	0.742	6.906	0.425	6.912	0.426	6.881	0.434	7.097	0.410	6.920	0.438	6.821	0.405	6.918
5	0.789	6.767	0.667	6.871	0.420	6.904	0.425	6.841	0.426	7.104	0.419	6.953	0.431	6.803	0.413	6.892
6	0.744	6.796	0.697	6.940	0.416	6.921	0.457	6.810	0.429	7.128	0.409	6.947	0.432	6.766	0.411	6.892
7	0.716	6.854	0.732	6.917	0.425	6.901	0.435	6.840	0.426	7.093	0.410	6.941	0.435	6.779	0.420	6.912
8	0.747	6.817	0.704	6.870	0.423	6.917	0.425	6.828	0.428	7.089	0.411	6.949	0.433	6.733	0.406	6.894
9	0.718	6.759	0.743	6.890	0.418	6.910	0.463	6.830	0.427	7.081	0.410	6.944	0.434	6.686	0.405	6.901
10	0.816	6.770	0.676	6.843	0.415	6.928	0.422	6.810	0.428	7.097	0.405	6.940	0.432	6.681	0.408	6.906
Average	0.739	6.787	0.725	6.909	0.420	6.911	0.434	6.830	0.430	7.098	0.411	6.940	0.434	6.792	0.410	6.898

Dataset 2

Dataset 2

Experiment	Error															
	Uniform				Centripetal				Exponential				Chord Length			
	Equal Distance		Averaging		Equal Distance		Averaging		Equal Distance		Averaging		Equal Distance		Averaging	
	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE
1	11.55143	11.13579	12.03508	11.54261	13.60008	13.32141	12.58382	12.3651	13.42579	13.18996	12.33213	12.03303	12.95767	12.60319	11.9791	11.72176
2	11.55143	11.24532	12.03508	11.64146	13.62171	13.22975	12.58382	12.26598	13.42579	13.06264	12.33213	12.04524	12.95947	12.8061	11.98375	11.70032
3	11.57796	11.1991	12.03508	11.65886	13.62171	13.29684	12.58389	12.19333	13.42579	13.16262	12.3191	12.07581	12.95947	12.6629	11.98375	11.78693
4	11.57796	11.12249	12.01452	11.57546	13.62171	13.2376	12.58389	12.33479	13.44219	13.01366	12.3191	12.11784	12.95947	12.68359	11.98375	11.75344
5	11.57796	11.16591	12.01452	11.65585	13.64027	13.32375	12.58389	12.24803	13.44219	13.06575	12.3191	12.11129	12.97407	12.58429	11.97736	11.57232
6	11.57098	11.22065	12.01452	11.45114	13.64027	13.15308	12.56556	12.28899	13.44219	13.11615	12.32758	11.97957	12.97407	12.73466	11.97736	11.77963
7	11.57098	11.09803	12.02184	11.72249	13.64027	13.20038	12.56556	12.26625	13.44871	12.80646	12.32758	12.05427	12.97407	12.61054	11.97736	11.7629
8	11.57098	11.14618	12.02184	11.603	13.62563	13.2621	12.56556	12.26093	13.44871	13.03493	12.32758	12.05487	12.95935	12.6847	11.98084	11.76273
9	11.5321	11.08013	12.02184	11.40901	13.62563	13.39198	12.56182	12.27373	13.44871	13.13429	12.33517	12.07033	12.95935	12.51317	11.98084	11.77268
10	11.5321	10.86546	12.0322	11.64865	13.62563	13.2842	12.56182	12.04118	13.45012	13.17835	12.33517	12.05847	12.95935	12.67724	11.98084	11.80505
Average	11.56139	11.12791	12.02465	11.59085	13.62629	13.27011	12.57396	12.25383	13.44002	13.07648	12.32746	12.06007	12.96363	12.65604	11.98050	11.74178
Minimum	11.53210	10.86546	12.01452	11.40901	13.60008	13.15308	12.56182	12.04118	13.42579	12.80646	12.31910	11.97957	12.95767	12.51317	11.97736	11.57232
Maximum	11.57796	11.24532	12.03508	11.72249	13.64027	13.39198	12.58389	12.36510	13.45012	13.18996	12.33517	12.11784	12.97407	12.80610	11.98375	11.80505

Experiment	CPU Time															
	Uniform				Centripetal				Exponential				Chord Length			
	Equal Distance		Averaging		Equal Distance		Averaging		Equal Distance		Averaging		Equal Distance		Averaging	
	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE
1	0.324	5.012	0.344	5.272	0.348	5.093	0.336	5.288	0.333	5.378	0.356	5.148	0.326	5.575	0.348	5.196
2	0.324	4.986	0.343	5.193	0.329	5.07	0.328	5.262	0.334	5.341	0.335	5.141	0.327	5.469	0.349	5.123
3	0.322	4.97	0.35	5.156	0.328	5.068	0.328	5.268	0.342	5.395	0.338	5.148	0.328	5.398	0.348	5.749
4	0.323	4.974	0.341	5.219	0.333	5.078	0.335	5.314	0.341	5.331	0.337	5.194	0.327	5.444	0.366	5.508
5	0.324	5.007	0.34	5.169	0.329	5.079	0.332	5.274	0.335	5.38	0.342	5.142	0.331	5.428	0.356	5.496
6	0.331	5.001	0.343	5.174	0.328	5.088	0.326	5.27	0.335	5.33	0.343	5.151	0.325	5.4	0.355	5.492
7	0.329	4.974	0.344	5.162	0.333	5.066	0.325	5.285	0.34	5.363	0.341	5.15	0.324	5.515	0.357	5.497
8	0.329	5.009	0.355	5.162	0.328	5.08	0.327	5.282	0.338	5.344	0.339	5.146	0.327	5.468	0.347	5.486
9	0.327	4.989	0.345	5.183	0.329	5.079	0.334	5.27	0.337	5.38	0.343	5.16	0.325	5.41	0.357	5.483
10	0.325	4.989	0.342	5.182	0.327	5.09	0.332	5.272	0.335	5.35	0.342	5.152	0.325	5.439	0.346	5.504
Average	0.326	4.991	0.345	5.187	0.331	5.079	0.330	5.279	0.337	5.359	0.342	5.153	0.327	5.455	0.353	5.453

Dataset 3

Experiment	CPU Time															
	Uniform				Centripetal				Exponential				Chord Length			
	Equal Distance		Averaging		Equal Distance		Averaging		Equal Distance		Averaging		Equal Distance		Averaging	
	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE
1	0.374	5.574	0.525	5.967	0.37	6.059	0.593	5.798	0.35	6.159	0.631	6.159	0.35	5.906	0.537	5.784
2	0.37	5.568	0.53	5.961	0.355	6.057	0.617	5.791	0.352	6.126	0.566	6.126	0.351	5.899	0.491	5.784
3	0.369	5.572	0.594	5.966	0.359	6.051	0.583	5.797	0.351	6.111	0.523	6.111	0.351	5.909	0.487	5.684
4	0.363	5.573	0.504	5.993	0.358	6.056	0.575	5.783	0.351	6.198	0.632	6.198	0.349	5.904	0.498	5.711
5	0.363	6.203	0.52	5.96	0.357	6.032	0.602	5.805	0.352	6.127	0.487	6.127	0.35	5.883	0.619	5.695
6	0.363	5.971	0.485	6.002	0.359	6.028	0.485	5.799	0.358	6.187	0.48	6.187	0.358	5.915	0.526	5.681
7	0.364	5.916	0.647	5.952	0.36	6.043	0.656	5.785	0.35	6.234	0.6	6.234	0.349	5.884	0.567	5.686
8	0.361	5.882	0.565	5.958	0.357	6.004	0.586	5.771	0.361	6.124	0.566	6.124	0.351	5.916	0.456	5.683
9	0.365	5.899	0.522	5.947	0.356	6.029	0.53	5.891	0.35	6.12	0.544	6.12	0.349	5.89	0.523	5.675
10	0.361	6.478	0.45	5.985	0.36	6.032	0.541	5.623	0.349	6.069	0.541	6.069	0.353	5.905	0.649	5.679
Average	0.365	5.864	0.534	5.969	0.359	6.039	0.577	5.784	0.352	6.146	0.557	6.146	0.351	5.901	0.535	5.706

Experiment	CPU Time															
	Uniform				Centripetal				Exponential				Chord Length			
	Equal Distance		Averaging		Equal Distance		Averaging		Equal Distance		Averaging		Equal Distance		Averaging	
	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE
1	0.374	5.574	0.525	5.967	0.37	6.059	0.593	5.798	0.35	6.159	0.631	6.159	0.35	5.906	0.537	5.784
2	0.37	5.568	0.53	5.961	0.355	6.057	0.617	5.791	0.352	6.126	0.566	6.126	0.351	5.899	0.491	5.784
3	0.369	5.572	0.594	5.966	0.359	6.051	0.583	5.797	0.351	6.111	0.523	6.111	0.351	5.909	0.487	5.684
4	0.363	5.573	0.504	5.993	0.358	6.056	0.575	5.783	0.351	6.198	0.632	6.198	0.349	5.904	0.498	5.711
5	0.363	6.203	0.52	5.96	0.357	6.032	0.602	5.805	0.352	6.127	0.487	6.127	0.35	5.883	0.619	5.695
6	0.363	5.971	0.485	6.002	0.359	6.028	0.485	5.799	0.358	6.187	0.48	6.187	0.358	5.915	0.526	5.681
7	0.364	5.916	0.647	5.952	0.36	6.043	0.656	5.785	0.35	6.234	0.6	6.234	0.349	5.884	0.567	5.686
8	0.361	5.882	0.565	5.958	0.357	6.004	0.586	5.771	0.361	6.124	0.566	6.124	0.351	5.916	0.456	5.683
9	0.365	5.899	0.522	5.947	0.356	6.029	0.53	5.891	0.35	6.12	0.544	6.12	0.349	5.89	0.523	5.675
10	0.361	6.478	0.45	5.985	0.36	6.032	0.541	5.623	0.349	6.069	0.541	6.069	0.353	5.905	0.649	5.679
Average	0.365	5.864	0.534	5.969	0.359	6.039	0.577	5.784	0.352	6.146	0.557	6.146	0.351	5.901	0.535	5.706

Dataset 4

Dataset 4																
Experiment	Error															
	Uniform				Centripetal				Exponential				Chord Length			
	Equal Distance		Averaging		Equal Distance		Averaging		Equal Distance		Averaging		Equal Distance		Averaging	
	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE
1	134.45484	132.56028	74.4717	73.64126	128.90182	127.43342	98.78587	97.70562	145.10064	143.66865	108.91495	108.83466	157.993	119.83926	157.22154	119.69207
2	134.48468	132.7246	74.46695	73.39042	128.92982	127.45215	98.78587	97.61752	145.16012	143.67681	108.81222	108.41697	157.993	119.83926	157.22154	119.66161
3	134.47861	132.71059	74.51972	73.52409	128.92982	127.67036	98.86496	97.40258	145.12248	143.88594	108.87862	108.71253	158.0552	119.8256	157.03691	119.90519
4	134.56881	132.7761	74.50315	73.27463	128.89012	127.40528	98.81745	97.79488	145.12248	143.804	108.79784	108.34404	158.0552	119.81452	156.85408	119.96292
5	134.56235	133.1024	74.46103	73.43802	128.87702	127.49148	98.85061	97.66775	145.11715	143.66361	109.7973	108.54877	158.1234	120.72296	156.93622	119.77075
6	134.56235	133.14345	74.46103	73.63088	128.87439	126.93663	98.80972	97.26775	145.0701	143.61297	109.7973	108.57713	158.1234	120.76848	157.23383	119.86565
7	134.50941	132.94745	74.51873	73.66285	128.91189	127.51799	98.80972	97.73376	145.0701	143.85067	108.75704	108.65348	158.046	120.76848	156.69535	120.02096
8	134.53382	132.96129	74.56178	73.56827	128.91189	127.45766	98.79632	97.54423	145.08926	143.89447	108.83952	108.63836	158.10411	119.83474	156.69535	119.74593
9	134.55341	132.66522	74.53195	73.41274	128.95039	127.4739	98.80318	97.44976	145.10452	143.74506	109.72884	108.73334	158.10411	119.78172	156.53638	119.75523
10	134.55341	133.04611	74.51439	73.66612	128.90563	127.64604	98.84599	97.50353	145.12735	143.74506	108.77599	108.73861	158.06525	120.71056	156.53638	120.04839
Average	134.52617	132.86375	74.50104	73.52093	128.90828	127.44849	98.81697	97.56874	145.10842	143.75472	109.10996	108.61979	158.06627	120.19056	156.89676	119.84287
Minimum	134.45484	132.56028	74.46103	73.27463	128.87439	126.93663	98.78587	97.26775	145.07010	143.61297	108.75704	108.34404	157.99300	119.78172	156.53638	119.66161
Maximum	134.56881	133.14345	74.56178	73.66612	128.95039	127.67036	98.86496	97.79488	145.16012	143.89447	109.79730	108.83466	158.12340	120.76848	157.23383	120.04839

Experiment	CPU Time															
	Uniform				Centripetal				Exponential				Chord Length			
	Equal Distance		Averaging		Equal Distance		Averaging		Equal Distance		Averaging		Equal Distance		Averaging	
	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE
1	0.792	10.994	0.903	10.045	0.874	9.884	0.768	9.801	0.935	10.305	0.895	10.305	0.762	10.068	0.827	10.005
2	0.952	10.505	0.905	9.913	0.831	10.018	0.845	9.83	0.861	10.351	0.867	10.351	0.788	10.013	0.823	9.997
3	0.851	9.973	0.889	9.858	0.82	9.901	0.774	9.819	0.826	10.588	0.802	10.588	0.816	10.013	0.784	9.997
4	0.743	9.819	0.839	9.892	0.895	9.922	0.817	9.829	0.855	10.088	0.802	10.088	0.778	10.029	0.775	10.076
5	0.741	9.845	0.838	9.87	0.938	9.887	0.801	9.808	0.756	10.065	0.777	10.065	0.896	10.014	0.767	10.018
6	0.734	9.821	0.79	9.877	0.785	9.886	0.806	9.808	0.806	10.027	0.879	10.027	0.84	10.003	0.738	9.971
7	0.811	9.85	0.885	9.907	0.711	9.894	0.774	9.797	0.817	10.027	1.465	10.027	0.839	10.003	0.836	9.987
8	0.726	9.858	0.717	9.866	0.678	9.888	0.718	9.793	0.812	10.009	1.42	10.009	0.976	10.691	0.732	10.027
9	0.784	9.833	0.839	9.858	0.888	9.884	0.841	9.806	0.682	10.027	1.404	10.027	0.84	10.691	0.768	10.006
10	0.768	9.816	0.807	9.902	0.7	9.896	0.643	9.797	0.83	10.033	1.235	10.033	0.905	10.438	0.748	10.001
Average	0.790	10.031	0.841	9.899	0.812	9.906	0.779	9.809	0.818	10.152	1.055	10.152	0.844	10.196	0.780	10.009

Dataset 5

Dataset 5

Experiment	Error															
	Uniform				Centripetal				Exponential				Chord Length			
	Equal Distance		Averaging		Equal Distance		Averaging		Equal Distance		Averaging		Equal Distance		Averaging	
	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE
1	8.8691	8.7691	11.3272	11.191	11.39184	12.87618	14.605	15.61788	14.33238	16.66883	15.46032	18.22266	16.67253	18.85438	18.31565	20.63361
2	8.7691	8.92835	11.32829	11.16991	11.39184	13.10189	14.605	15.5136	14.33238	16.97699	13.68953	18.61226	16.62055	18.68406	18.31565	20.4676
3	8.65328	8.72245	10.82206	10.87562	12.14269	13.62118	13.69879	15.47464	14.3406	16.79854	13.68953	18.57086	16.62055	18.57521	18.34691	20.57516
4	8.67945	8.85208	10.82206	10.96685	13.47278	13.64543	13.69879	15.46588	15.54999	16.68035	15.46064	18.63178	16.62055	18.85085	18.34691	20.57516
5	8.67945	8.87722	11.18076	11.14554	11.59689	13.51717	14.6375	15.46612	14.92358	16.6374	15.46064	18.47575	16.90835	18.91716	20.467	20.64151
6	8.96682	8.7996	11.18076	11.12677	11.59689	13.59212	14.28268	15.49664	14.92358	17.01958	15.27342	18.46918	16.90835	18.84312	20.467	20.45583
7	8.96682	8.98034	10.83504	11.06149	13.44365	13.55831	14.28268	15.58645	15.56804	16.91521	15.27342	18.70726	16.65432	18.91814	18.27929	20.68733
8	9.0066	8.88243	10.83504	11.06864	13.46503	13.63392	14.59593	15.52498	15.56804	16.92425	15.46178	18.72029	16.65432	18.74912	18.27929	20.24471
9	8.67585	8.69228	11.12257	11.15337	12.15178	13.38187	13.69181	15.25247	16.976	16.89559	15.46178	18.79947	16.65432	18.83547	18.16618	20.5666
10	8.67585	8.84417	11.12257	11.08857	11.54769	13.50334	13.69181	15.62217	16.976	16.90697	15.44438	18.66322	16.95249	18.93791	18.16618	20.76864
Average	8.79423	8.83480	11.05764	11.08478	12.22011	13.44314	14.17900	15.50208	15.34906	16.84237	15.06754	18.58727	16.72663	18.81654	18.71501	20.56162
Minimum	8.65328	8.69228	10.82206	10.87562	11.39184	12.87618	13.69181	15.25247	14.33238	16.63740	13.68953	18.22266	16.62055	18.57521	18.16618	20.24471
Maximum	9.00660	8.98034	11.32829	11.19100	13.47278	13.64543	14.63750	15.62217	16.97600	17.01958	15.46178	18.79947	16.95249	18.93791	20.46700	20.76864

Experiment	CPU Time															
	Uniform				Centripetal				Exponential				Chord Length			
	Equal Distance		Averaging		Equal Distance		Averaging		Equal Distance		Averaging		Equal Distance		Averaging	
	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE	GA	DE
1	0.715	4.717	0.651	4.789	0.67	4.715	0.5	4.645	0.771	4.615	0.587	4.527	0.808	4.974	0.521	4.582
2	0.83	4.71	0.767	4.696	0.682	4.666	0.61	4.786	0.759	4.609	0.531	4.508	0.756	4.992	0.526	4.571
3	0.845	4.755	0.844	4.696	0.682	4.659	0.608	4.722	0.785	4.615	0.545	4.558	1.011	4.689	0.515	4.578
4	0.695	4.708	0.866	4.696	0.753	4.679	0.569	4.722	0.768	4.619	0.584	4.517	0.819	4.636	0.515	4.565
5	0.765	4.7	0.663	4.683	0.828	4.642	0.549	4.614	0.712	4.61	0.458	4.516	0.689	4.55	0.461	4.571
6	0.749	4.735	0.739	4.663	0.781	4.66	0.536	4.613	0.749	4.625	0.569	4.51	0.723	4.53	0.535	4.571
7	0.808	4.715	0.836	4.656	0.752	4.684	0.585	4.642	0.754	4.664	0.588	4.522	0.817	4.557	0.535	4.573
8	0.686	4.715	0.759	4.652	0.815	4.675	0.525	4.618	0.682	4.64	0.533	4.516	0.762	4.531	0.566	4.57
9	0.774	4.707	0.799	4.65	0.711	4.677	0.623	4.593	0.768	4.626	0.545	4.528	0.717	4.573	0.602	4.583
10	0.745	4.74	0.877	4.667	0.771	4.674	0.594	4.64	0.763	4.608	0.586	4.512	0.752	4.545	0.629	4.574
Average	0.761	4.720	0.780	4.685	0.745	4.673	0.570	4.660	0.751	4.623	0.553	4.521	0.785	4.658	0.541	4.574

APPENDIX B: WEEKLY REPORT

Appendix B shows the weekly report of the research.

FINAL YEAR PROJECT WEEKLY REPORT

(~~Project I~~ / Project II)

Trimester, Year: Trimester 3, Year 3	**Study week no.:** 1
Student Name & ID: Kheng Jia Shen 16ACB03287	
Supervisor: Ts Dr Lim Seng Poh	
Project Title: B-Spline Curve Fitting with Different Parameterization Methods	

1. WORK DONE [Please write the details of the work done in the last fortnight.] Continue with the previous work. Calculation for basis function based on the input from previous parameters and knot vector.
2. WORK TO BE DONE Calculate the transpose, inverse function for the dataset.
3. PROBLEMS ENCOUNTERED Not familiar with the calculation.
4. SELF EVALUATION OF THE PROGRESS Moderate performance

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(~~Project I~~ / Project II)

Trimester, Year: Trimester 3, Year 3	**Study week no.:** 3
Student Name & ID: Kheng Jia Shen 16ACB03287	
Supervisor: Ts Dr Lim Seng Poh	
Project Title: B-Spline Curve Fitting with Different Parameterization Methods	

1. WORK DONE [Please write the details of the work done in the last fortnight.] Continue with the calculation to get control points. The transpose of the basis function is done. Some basic understanding about Gauss Jordan Elimination is studied to calculate the inverse function.
2. WORK TO BE DONE Calculate the error for the results
3. PROBLEMS ENCOUNTERED Less understanding about the inverse function.
4. SELF EVALUATION OF THE PROGRESS Moderate performance

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(~~Project I~~ / Project II)

Trimester, Year: Trimester 3, Year 3	**Study week no.:** 4
Student Name & ID: Kheng Jia Shen 16ACB03287	
Supervisor: Ts Dr Lim Seng Poh	
Project Title: B-Spline Curve Fitting with Different Parameterization Methods	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

Had some update on the previous inverse function. The previous results are incorrect and modification had been done so that the calculation can be completed.

2. WORK TO BE DONE

Calculate errors of the results and proceed to optimization.

3. PROBLEMS ENCOUNTERED

There is some logic of the flow that are confusing which will lead to incorrect results.

4. SELF EVALUATION OF THE PROGRESS

Moderate performance

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(~~Project I~~ / Project II)

Trimester, Year: Trimester 3, Year 3	**Study week no.:** 6
Student Name & ID: Kheng Jia Shen 16ACB03287	
Supervisor: Ts Dr Lim Seng Poh	
Project Title: B-Spline Curve Fitting with Different Parameterization Methods	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

The correctness of the errors is tested and record in table. Start off with studying the concept for GA optimization as this field is an unknown knowledge.

2. WORK TO BE DONE

Perform GA optimization

3. PROBLEMS ENCOUNTERED

Flow of the GA method and the operation.

4. SELF EVALUATION OF THE PROGRESS

Moderate performance

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(~~Project I~~ / Project II)

Trimester, Year: Trimester 3, Year 3	**Study week no.:** 7
Student Name & ID: Kheng Jia Shen 16ACB03287	
Supervisor: Ts Dr Lim Seng Poh	
Project Title: B-Spline Curve Fitting with Different Parameterization Methods	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

Start to work on the coding for GA. Initialization of the population and selection operation are done. With the input control points, the chromosome in population are initialize and selection is done.

2. WORK TO BE DONE

Continue with other operation in GA

3. PROBLEMS ENCOUNTERED

None

4. SELF EVALUATION OF THE PROGRESS

Moderate performance

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(~~Project I~~ / Project II)

Trimester, Year: Trimester 3, Year 3	**Study week no.:** 8
Student Name & ID: Kheng Jia Shen 16ACB03287	
Supervisor: Ts Dr Lim Seng Poh	
Project Title: B-Spline Curve Fitting with Different Parameterization Methods	

1. WORK DONE [Please write the details of the work done in the last fortnight.] Continue mutation, crossover and replacement operation in GA. After the operation, checking of the correctness of the results is done. Furthermore, some basic concept and the flow of DE is studied for understanding.
2. WORK TO BE DONE Perform DE optimisaton.
3. PROBLEMS ENCOUNTERED None
4. SELF EVALUATION OF THE PROGRESS Moderate performance

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(~~Project I~~ / Project II)

Trimester, Year: Trimester 3, Year 3	**Study week no.:** 9
Student Name & ID: Kheng Jia Shen 16ACB03287	
Supervisor: Ts Dr Lim Seng Poh	
Project Title: B-Spline Curve Fitting with Different Parameterization Methods	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

Further reviewed on the operation in DE. Some coding is done on some operation in DE such as crossover, selection and mutation.

2. WORK TO BE DONE

Continue with the other operation in DE. Check the correctness of the results.

3. PROBLEMS ENCOUNTERED

None

4. SELF EVALUATION OF THE PROGRESS

Moderate performance

Supervisor's signature

Student's signature

POSTER

B-SPLINE CURVE FITTING WITH DIFFERENT PARAMETERIZATION METHODS

SUPERVISOR: DR LIM SENG POH

Introduction

B-spline curve is important in the geometric modelling field and CAD in visualization and curve modelling. B-spline is flexible and could provide a better behaviour and local control. However, there is an issue in many fields which is to construct an optimal curve with the given data points. Therefore, in this research, a comparison is made between the parameterization methods to determine the optimal method for the B-spline curve fitting. Uniformly spaced and averaging knot vector generations are used and analysed. After applying the parameterization method, genetic algorithm and differential evolution optimization are used to optimise the accuracy of the curve fitting.

Research Methodologies

Analysis Literature Review and Problem Definition → Data Collection and Definition → Curve Fitting Process → Optimization → Testing → Documentation

Insert data points of dataset → Perform parameterization → Generate knot vector → Perform basis function calculation → Perform control points calculation → Perform curves data calculation → Perform curves error calculation

Problem Statement

- Optimization problems
- Accuracy could not be guaranteed
- Parameterization methods could not handle all types of data points

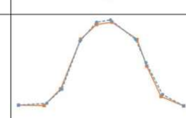
Project Objective

- To analyse different parameterization methods for B-spline curve.
- To optimize the accuracy of B-spline curve

Project Scope

- C++ language
- B-spline curve
- Four Parameterization Methods
- Genetic Algorithm (GA) and Differential Evolution (DE)

Analysis

Dataset	Graph	Explanation
1		DE with uniform parameterisation method and averaging knot vector generation

Parameterisation Methods	Knot Vector Generation	
	Uniformly Spaced	Averaging
Uniform	26.44794	16.78655
Centripetal	29.17968	17.14180
Exponential	31.23465	18.04786
Chord Length	32.48294	19.14408

Conclusion

This research is about the B-spline curve fitting with different parametrization methods. By using the input of dataset given, four parameterizations method were done to obtain the parameter values. After that, uniformly spaced knot vector and averaging knot vector generation were calculated with the use of parameters obtained previously. GA and DE were used for optimisation. Analysis had been done to compare the four parameterization methods which are uniform, chord length, centripetal and exponential methods. Both the knot vector generation and optimisation were analysed to identify the differences between them.

UTAR
UNIVERSITI TUNKU ABDUL RAHMAN

Proposed by: Kheng Jia Shen 16ACB03287

Bachelor of Computer Science (Hons)
Faculty of Information and Communication Technology (Kampar Campus), UTAR.

PLAGIARISM CHECK RESULT

Document Viewer

Turnitin Originality Report

Processed on: 23-Apr-2020 18:25 +08
ID: 1305335389
Word Count: 13713
Submitted: 2

B-Spline Curve Fitting with Different Paramet... By Jia Shen Kheng

Similarity Index	Similarity by Source	
13%	Internet Sources:	2%
	Publications:	8%
	Student Papers:	8%

include quoted include bibliography excluding matches < 8 words mode: quickview (classic) report Change mode print download

2% match (publications)
Lecture Notes in Computer Science, 2013.

1% match (student papers from 21-Mar-2013)
Submitted to University of Warwick on 2013-03-21

1% match (student papers from 29-Aug-2014)
Submitted to University of Bath on 2014-08-29

1% match (student papers from 08-Jan-2012)
Submitted to Universiti Teknologi Malaysia on 2012-01-08

1% match (publications)
H. Haron, A. Rehman, D. I. S. Adi, S. P. Lim, T. Saba. "Parameterization Method on B-Spline Curve", Mathematical Problems in Engineering, 2012

<1% match (student papers from 14-May-2013)
Submitted to Universiti Teknologi Malaysia on 2013-05-14

<1% match (student papers from 20-Nov-2013)
Submitted to INTI International University on 2013-11-20

<1% match (publications)
Lim, Seng Poh, and Habibollah Haron. "Performance comparison of Genetic Algorithm, Differential Evolution and Particle Swarm Optimization towards benchmark functions", 2013 IEEE Conference on Open Systems (ICOS), 2013.

<1% match (student papers from 02-May-2008)
Submitted to The University of Manchester on 2008-05-02

<1% match (publications)
"Computer Aided Engineering Design", Springer Science and Business Media LLC, 2005

<1% match (Internet from 18-Oct-2010)
http://cadcam.eng.sunysb.edu

<1% match (publications)
Mehmet B. Ercan, Jonathan L. Goodall. "Design and implementation of a general software library for using NSGA-II with SWAT for multi-objective model calibration", Environmental Modelling & Software, 2016

ORIGINALITY REPORT

13%	2%	8%	8%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Lecture Notes in Computer Science, 2013. Publication	2%
2	Submitted to Universiti Teknologi Malaysia Student Paper	1%
3	Submitted to University of Warwick Student Paper	1%
4	Submitted to University of Bath Student Paper	1%
5	H. Haron, A. Rehman, D. I. S. Adi, S. P. Lim, T. Saba. "Parameterization Method on B-Spline Curve", Mathematical Problems in Engineering, 2012 Publication	1%
6	Submitted to INTI International University Student Paper	<1%
7	Lim, Seng Poh, and Habibollah Haron. "Performance comparison of Genetic Algorithm, Differential Evolution and Particle Swarm	<1%

Universiti Tunku Abdul Rahman			
Form Title : Supervisor's Comments on Originality Report Generated by Turnitin for Submission of Final Year Project Report (for Undergraduate Programmes)			
Form Number: FM-IAD-005	Rev No.: 0	Effective Date: 01/10/2013	Page No.: 1of 1

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

Full Name(s) of Candidate(s)	Kheng Jia Shen
ID Number(s)	16ACB03287
Programme / Course	Bachelor of Computer Science (HONS)
Title of Final Year Project	B-Spline Curve Fitting With Different Parameterization Methods

Similarity	**Supervisor's Comments (Compulsory if parameters of originality exceeds the limits approved by UTAR)**
Overall similarity index: 13 % **Similarity by source** Internet Sources: 2% Publications: 8% Student Papers: 8%	Ok.
Number of individual sources listed of more than 3% similarity: 0	
Parameters of originality required and limits approved by UTAR are as Follows: **(i) Overall similarity index is 20% and below, and** **(ii) Matching of individual sources listed must be less than 3% each, and** **(iii) Matching texts in continuous block must not exceed 8 words** ***Note: Parameters (i) – (ii) shall exclude quotes, bibliography and text matches which are less than 8 words.***	

Note Supervisor/Candidate(s) is/are required to provide softcopy of full set of the originality report to Faculty/Institute

Based on the above results, I hereby declare that I am satisfied with the originality of the Final Year Project Report submitted by my student(s) as named above.

______________________________ ______________________________

Signature of Supervisor Signature of Co-Supervisor

Name: Ts. Dr. Lim Seng Poh

Date: 23/4/2020

Name: ___________________________

Date: ____________________________

UNIVERSITI TUNKU ABDUL RAHMAN

FACULTY OF INFORMATION & COMMUNICATION TECHNOLOGY (KAMPAR CAMPUS)

CHECKLIST FOR FYP2 THESIS SUBMISSION

Student Id	16ACB03287
Student Name	KHENG JIA SHEN
Supervisor Name	TS DR LIM SENG POH

TICK (√)	**DOCUMENT ITEMS** Your report must include all the items below. Put a tick on the left column after you have checked your report with respect to the corresponding item.
/	Front Cover
/	Signed Report Status Declaration Form
/	Title Page
/	Signed form of the Declaration of Originality
/	Acknowledgement
/	Abstract
/	Table of Contents
/	List of Figures (if applicable)
/	List of Tables (if applicable)
	List of Symbols (if applicable)
/	List of Abbreviations (if applicable)
/	Chapters / Content
/	Bibliography (or References)
/	All references in bibliography are cited in the thesis, especially in the chapter of literature review
/	Appendices (if applicable)
/	Poster
/	Signed Turnitin Report (Plagiarism Check Result - Form Number: FM-IAD-005)

*Include this form (checklist) in the thesis (Bind together as the last page)

I, the author, have checked and confirmed all the items listed in the table are included in my report. (Signature of Student) Date: 23/04/2020	Supervisor verification. Report with incorrect format can get 5 mark (1 grade) reduction. ______________________ (Signature of Supervisor) Date: 23/4/2020