Shape-preserving Interpolation with Biarcs and NURBS

by

Unmesh Anant

A Thesis submitted to the Faculty of Graduate Studies of
The University of Manitoba
in partial fulfillment of the requirements of the degree of

Master of Science

Department of Computer Science
University of Manitoba
Winnipeg, Manitoba, Canada

Thesis advisor

Dr. Desmond J. Walton

Author

Unmesh Anant

Shape-preserving Interpolation with Biarcs and NURBS

Abstract

Non-Uniform Rational B-Splines (NURBS) curve has acquired great significance in the field of Computer Aided Design and Machining due to their ability to draw a large variety of shapes in an interactive computer graphics environment. A biarc curve is a composition of two circular arcs such that they are tangent continuous at the point of join. Biarcs have replaced traditionally used line segments in approximating curves and surfaces for generating tool paths of Computerised cutting machines called CNC (Computerised Numerical Controlled) machines. This is due to their ability to be at a greater proximity to the original curve with fewer number of segments. Since most of the machining tools can move only in straight lines and circular arcs, it is desirable that the tool paths be composed of biarcs and/or straight line segments. Shape preserving interpolation is a technique of drawing a curve through a set of points such that the shape represented by the data points are preserved. Both NURBS and biarc curves are not essentially shape preserving curves; however, if certain constraints are imposed on them, they are able to preserve the shape represented by the data points. This work proposes a technique that incorporates both NURBS and biarcs to perform the interpolation. The advantages are twofold; it acts as a common platform for the two techniques to operate together, which is novel, and the fitted NURBS curve can

be approximated by biarcs, which has applications in the machining industry.

Contents

Abstract ii
Table of Contents iv
List of Figures vi
List of Tables viii
Acknowledgments ix
Dedication x

1 Introduction **1**
1.1 Shape-preserving interpolation 5
1.2 Problem Description 6
1.3 Proposed Solution: Biarc Fitting and NURBS Curve Fitting 7
1.3.1 Biarc Fitting 8
1.3.2 NURBS Curve Fitting 11

2 Background **14**
2.1 Types of Curves 14
2.2 Desirable Properties of Curves 15
2.3 Curvature 15
2.4 Cusps 16
2.5 Inflection Points 16
2.6 Two-point G^1 Hermite data 17
2.7 Shape-preserving Interpolation 17
2.8 Biarc 18
2.9 The Bézier Curve 18
2.10 B-splines 20
2.11 Non Uniform Rational B-spline Curves (NURBS) 22
2.12 Tolerance Limit (ϵ) 24

3 Related Work **25**
3.1 Biarcs 25
3.1.1 Approximation of discrete data points 26

3.1.2 Family of biarcs 27
3.1.3 Approximation of parametric curves 27
3.2 Shape preserving 29

4 Biarc Fitting **32**
4.1 Biarc Fitting: Mathematical Model 33
4.2 Implementation Algorithm 43

5 NURBS Curve Fitting **45**
5.1 Constructing the quadratic NURBS Curve 45
5.1.1 Equation of the first quadratic $\overline{C}_L(u)$ 46
5.1.2 Equation of the second quadratic $\overline{C}_R(u)$ 46
5.2 Linear Blending 48
5.3 Implementation Algorithm 51
5.4 Negative weights 51

6 Examples **53**
6.1 Biarc Fitting 53
6.1.1 Example 1 54
6.1.2 Example 2 54
6.2 NURBS Curve Fitting 55
6.2.1 Cases with NURBS curve as circular arcs 56
6.2.2 Cases with NURBS curve in general 56
6.2.3 Cases with negative weights 56

7 Approximating NURBS Curve with Biarcs **64**
7.1 Calculating points at angular intervals 65
7.2 Determining the distance between the curve and the biarc 67
7.3 Parametric subdivision of cubic NURBS curve 68
7.4 Implementation Algorithm 72

8 Applications **74**
8.1 Applications of NURBS Curve 74
8.2 Applications of biarcs 76
8.3 Results 79

9 Conclusion **84**

Bibliography **87**

List of Figures

1.1 A unique biarc. 9
1.2 Quadratic NURBS curve drawn on data points. 12
1.3 NURBS curve Fitting. 13

2.1 A cubic NURBS curve with an inflection point. 16
2.2 Two-point G^1 Hermite data. 17
2.3 A biarc. 19
2.4 Effect of control points on a quadratic NURBS curve. 23
2.5 Effect of change in weight on a quadratic NURBS curve. 23

4.1 Biarc with h, k, W and θ. 34
4.2 Biarc when $r_1 > r_0$. 42

5.1 Data set with $\overline{C}_L(u)$ 47
5.2 Data set with $\overline{C}_L(u)$ and $\overline{C}_R(u)$. 48
5.3 A set of quadratic NURBS curves. 49

6.1 Biarc when $W = 60°$, $\psi = 30°$ for $\theta = 5°, 10°, 20°, 30°, 40°$ and $50°$. . . 54
6.2 Biarc when $W = 80°$, $\psi = 25°$ for $\theta = 1°, 10°, 20°, 30°, 40°$ and $45°$. . . 55
6.3 Final NURBS curve when $\alpha = 40°$, $\beta = 320°$ 57
6.4 Final NURBS curve when $\alpha = 105°$, $\beta = 255°$ 58
6.5 Final NURBS curve when $\alpha = 43°$, $\beta = 38°$ 59
6.6 Final NURBS curve when $\alpha = 81°$, $\beta = 22°$ 60
6.7 Final NURBS curve when $\alpha = 35°$, $\beta = 114°$ 61
6.8 Final NURBS curve when $\alpha = 58°$, $\beta = 141°$ 62
6.9 Final NURBS curve when $\alpha = 76°$, $\beta = 164°$ 63

7.1 Distance between a NURBS curve and a biarc. 65
7.2 Parametric subdivision of cubic NURBS. 68

8.1 Camshaft using NURBS curve fitting. 77
8.2 Ping-Pong Paddle using NURBS curve fitting. 78

8.3 A biarc fitted camshaft: tolerance limit = 10^{-5}, biarcs = 70. 80
8.4 A biarc fitted ping-pong paddle: tolerance limit = 10^{-5}, biarcs = 196. 82
8.5 Camshaft interpolation 82
8.6 Ping-Pong paddle interpolation 83

List of Tables

8.1 Dataset for the camshaft model 76
8.2 Dataset for the ping-pong paddle 76
8.3 Biarc Fitting: Camshaft 81
8.4 Biarc Fitting: Ping-Pong paddle 81

Acknowledgments

I would like to thank my supervisor Dr. Desmond J. Walton for his patience, constant support and numerous feedbacks through the course of this research. I would also like to thank my examination committee for their constructive comments.

I am grateful (as ever) to my parents, my 'little' sister for constant moral and emotional support all the time and, all my friends who ever wished that I finish my work and pushed me through it.

Thanks a lot everyone!

This thesis is dedicated to my folks and my sister. You are my inspiration!

Chapter 1

Introduction

Interpolation refers to the construction of a curve through a given set of data points. When the curve preserves the shape of the data, it is known as shape-preserving interpolation. Given a set of data points $\overline{I}_i$, $i = 0, 1, 2 \ldots, N$, and unit tangent vectors $\overline{T}_i$ at those points, a polynomial curve may have to be constructed which exactly fits these data points and unit tangent vectors. This is known as G^1 Hermite interpolation. A curve drawn through a set of points assists in better understanding the shape and properties of the data which gave rise to those points. Some of the applications of the interpolation techniques are in civil engineering for designing highways and railroads, in mechanical engineering for designing machines or other parts, in computer vision for shape completion and shape partitioning, and in the design of rollercoasters and bicycle racing tracks [1]. NURBS (Non Uniform Rational B-Splines) curves and biarc are curves that are not shape preserving in general. Certain constraints have to be imposed on these curves when they are used for interpolation so that they preserve the shape of the data points. NURBS curves and

biarcs (see section 2.8 and 2.11) have the desirable properties and flexibility to act as a significant tool for shape interpolation in Computer Aided Design and Computer Aided Manufacturing (CAD/CAM). This involves use of a computer to design or (and) manufacture of machine parts. CAD is a technique used for interactive graphical design of intermediate or end consumer goods, tools and machinery which could be then used for manufacture of other components.

CAM involves the use of computers to manufacture parts. A Computerized Numerical Controlled (CNC) machine is capable of machining parts automatically, once it receives instructions from its processing unit. A design is created in a CAD system and after proper analysis, instructions are fed to a CNC machine, which creates tool paths for the cutter to machine the part. The design involves the use of curves and surfaces. These include straight lines, circular arcs, Béziers, B-splines in 2D and freeform surfaces in 3D.

CAM machines, however, have their limitations. Most machining tools currently in use have controllers only for straight line or circular paths. Thus, the design has to be redrawn or approximated in terms of straight lines and circular arcs. Several schemes have been introduced, over the years, to convert the design curves and surfaces into straight lines and circular arcs to accommodate the limitations of the machine. These schemes are known as *curve or surface approximation*. An inexact or an inaccurate but useful representation of an object is called an approximation. In terms of CAD or CAM, the approximation of curves and surfaces uses straight lines and circular arcs.

Traditionally, CNC tool paths have been generated using straight lines because early cutters could not move along circular arcs. There are still many of these cutters

in use. These cutters have two major drawbacks. Large numbers of linear move commands produced during the tool path generation overwhelm the size of the NC program storage and may cause buffer overflow of the CNC machine [2]. The tool would then have to stop and wait for the next instruction to load, resulting in a 'stop and go' movement of the tool. Furthermore, the scheme produces a tangentially discontinuous tool path resulting in frequent sudden changes in the direction of the tool. The above factors are a major concern for the CAM industry as they not only affect the finished quality of the part, but may also cause wear and tear of expensive CNC machines [3].

With the introduction of circular interpolation controllers for these machines, circular arcs could also be used for the approximation. Biarc approximation schemes were introduced in the late 70's to design approximation curves using circular arcs. A biarc, introduced by Bolton [4], is a curve formed by joining two arcs such that the unit tangent is continuous at the point of join (also called G^1 continuity). The two arcs could be circular arcs or an arc and a straight line segment. Curves composed of arcs have fewer segments than those composed of straight line segments, so there is little or no buffer overflow. The tool path created is smooth and, hence, causes less wear and tear on the machine.

Over the years, biarcs have been used to approximate various curves and surfaces [5, 6, 7, 8, 9, 10, 11]. A brief review of these works is provided in chapter 3.

Another application of the biarc is in shape completion. It is the process of filling in a missing or hidden part of an object. This scenario is termed as occlusion. Kimia et al. [1] uses shape completion in addressing the problem of *occlusion* in computer

vision applications. Here, a pleasing curve is desired to fill in the gap such that the object maintains G^1 continuity. According to the authors, biarcs and cubic splines are typically used for these problems due to their ability to interpolate curves, given two points and tangent directions at those points. They are also used in tasks that involve filling in or what is known as *in-painting*. In-painting involves a complex blending of functions to create images or impressions, e.g., illusionary impressions.

Some of the other major applications of biarcs are in mechanical engineering, for the completion of ship and aircraft parts, and in computer typography, for designing letters and symbols.

Meek and Walton [12] address the use of biarcs in shape completion. A set of two points with respective tangent directions at those points are referred to as two-point G^1 Hermite data. Meek and Walton state that given two-point G^1 Hermite data, a family of biarcs is possible through those points. The authors illustrate an example of a violin represented by points and use their method to draw an optimal biarc, through the set of points, to obtain a smooth shape.

The purpose for this thesis is to develop an interpolation scheme that makes use of both biarcs and NURBS curves. The motivation for this work is to treat biarc fitting and shape preserving interpolation as an integrated problem. This would not only have application in the field of curve and surface approximation, but also in shape preservation or shape completion. The goal is to develop techniques suitable for user interaction using computer graphics.

1.1 Shape-preserving interpolation

Interpolation is defined by Farin [13] as finding a curve or surface that satisfies some imposed constraint exactly. The most common constraint is the requirement of passing through a set of given points. Shape-preserving interpolation however, is a requirement that the shape of the given data sets be preserved.

An important challenge in the CAD/CAM industry is to define an interpolating curves or surfaces that can be efficiently used in the manufacturing industry such as design of airplanes, ships and machine parts. For many other applications such as the paths of vehicles, particles, etc., one may wish to construct an interpolating curve that would reflect the information of the data, or in other words, 'preserve the shape of the data'. Thus, the need for shape-preserving interpolation schemes comes into the picture. Interpolating curves in general are not shape-preserving; however, certain constraints may be imposed on them to make them shape-preserving.

The following are the basic properties that are desired of a shape preserving interpolation curve. These are general properties and may differ in terms of their specifications, depending on the scheme.

- Local convexity preserving i.e. if the curve preserves the convexity for each set of data points.
- Smoothness or minimal strain energy.
- Ease of generating closed or open curves. A closed curve is a curve with no endpoints and completely encloses an area, otherwise it is an open curve.

- G^1 (tangent continuity) or G^2 (curvature continuity) should be preserved, depending on the application.
- Stability i.e. if the input data is changed slightly, the resulting curve is changed only slightly.
- Invariance under rotation or scaling.

1.2 Problem Description

This work addresses the problem of shape-preserving interpolation using NURBS curve together with biarc approximation. Previous work addressed shape-preserving interpolation separate from biarc approximation. However, the nature of the problems in terms of how the data sets have to be approximated using circular arcs or curves, are very closely related. The problem is stated as follows.

Given G^1 Hermite data, i.e., two points $\overline{A}$ and $\overline{B}$ with associated unit tangent vectors $\overline{T}_A$ and $\overline{T}_B$ respectively, find a simple curve joining $\overline{A}$ to $\overline{B}$ that satisfies the following three criteria:

1. If the data are consistent with an S-shaped or a C-shaped curve joining $\overline{A}$ to $\overline{B}$, the curve should be S-shaped or C-shaped, respectively.
2. If the data arose from a circular arc, the curve should reduce to that circular arc.
3. If the data arose from a known curve, or is presumed to have arisen from a known curve, the new curve should be a good approximation to the known

curve.

1.3 Proposed Solution: Biarc Fitting and NURBS Curve Fitting

In this thesis, the word 'tangent' will be taken to mean 'unit tangent vector'. Given the two points $\overline{A}$, $\overline{B}$ with tangents $\overline{T}_A$, $\overline{T}_B$ at those points, this work addresses the above problem by suggesting two different approaches to obtain a curve starting from $\overline{A}$ and ending at $\overline{B}$.

The first approach uses a biarc to match the data. Biarcs consist of two circular arcs connected in a tangent continuous (G^1) fashion. If the data arose from a circular arc, each arc of the biarc will be a segment of the original arc. We refer to this approach as *biarc fitting*.

In the second approach, a NURBS curve will be used to match G^1 Hermite data in a way that if the data points $\overline{A}$, $\overline{B}$ belong to a circular arc, it gives back (or reconstructs) the circular arc. If the data is not from a circular arc, a cubic NURBS curve will be obtained. The advantage of using a NURBS curve over a polynomial Bézier curve is that a NURBS as well as a Rational Bézier curve precisely draw conics and circular arcs, while a polynomial Bézier can only approximate them. We will refer this scheme as *NURBS curve fitting*.

However, in both the cases, the curve will pass through the data points in a shape-preserving manner.

1.3.1 Biarc Fitting

This section introduces the concept of a biarc, its properties and how it can be used as a shape-preserving interpolation tool. Given two points $\overline{A}$, $\overline{B}$ and tangents $\overline{T}_A$, $\overline{T}_B$ at those points, we need to construct a biarc through those points. The reason for using biarcs is that a single circular arc cannot always be constructed between the two points and satisfy the tangents at the start and end points. G^1 Hermite data have six constraints, namely; the start point (2), the end point (2), the start tangent (1) and the end tangent (1) while, a circular arc has five, the center (2), the radius (1), the beginning angle (1) and the span angle(1). Thus, one (more) degree of freedom than a circular arc has, is required in order to construct a curve that matches G^1 Hermite data.

A biarc, being composed of two circular arcs, has 10 degrees of freedom. Point $\overline{P}_0$ and tangent $\overline{T}_0$ impose three constraints. Point $\overline{P_1}$ and tangent $\overline{T}_1$ impose 3 more constraints. Furthermore, the point of join $\overline{J}$ and continuity of tangent at it, also impose three constraints. Therefore, out of 10, one degree of freedom is left over. This enables a family of biarcs to interpolate G^1 Hermite data. An additional constraint is required to obtain a unique G^1 Hermite interpolating biarc.

Illustration: Biarc Fitting

Consider Figure 1.1. Here, $\overline{P}_0 = \overline{A}$, $\overline{T}_0 = \overline{T}_A$, $\overline{P}_1 = \overline{B}$ and $\overline{T}_1 = \overline{T}_B$. Point $\overline{P}$ is the intersection of the two tangents. The two circular arcs of the biarc intersect at the point $\overline{J}$ and $\overline{EF}$ is the tangent at the point of join. Let $\overline{D} = \overline{P}_1 - \overline{P}_0$. Furthermore, let α be the angle from $\overline{D}$ to $\overline{T}_0$, β the angle from $\overline{T_1}$ to $\overline{D}$, ϕ the angle from $\overline{JP_0}$ to

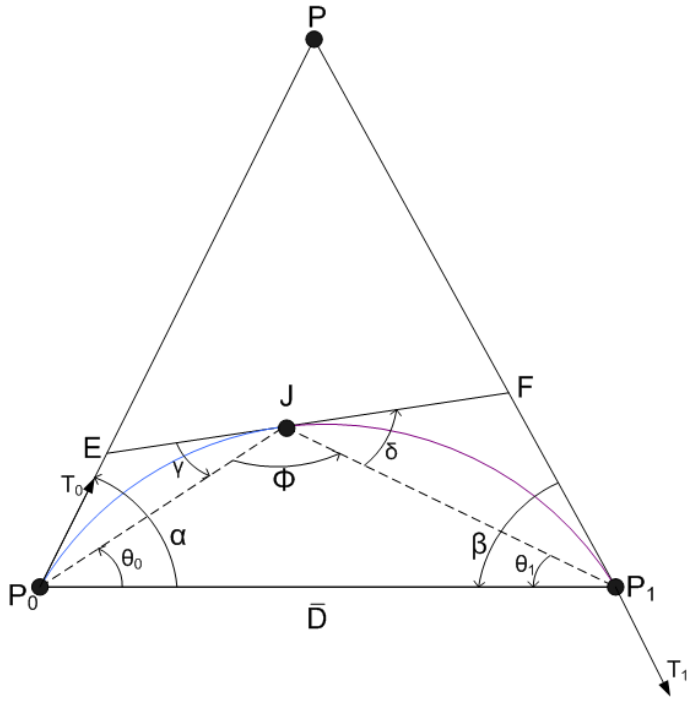

Figure 1.1: A unique biarc.

$\overline{JP}_1$, θ_0 the angle from $\overline{D}$ to $\overline{P_0J}$ and θ_1 the angle from $\overline{JP}_1$ to $\overline{D}$. Let δ be the angle from vector $\overline{JP}_1$ to vector $\overline{JF}$ and γ be the angle from $\overline{JE}$ to $\overline{JP}_0$. The triangles ΔP_0JE and ΔP_1FJ are isosceles, so

$$\gamma = \angle EP_0J = \alpha - \theta_0 \tag{1.1}$$

and

$$\delta = \angle JP_1F = \beta - \theta_1. \tag{1.2}$$

In the straight angle EJF, $\pi = \gamma + \phi + \delta$. Using (1.1) and (1.2) and substituting for γ and δ in terms of θ_0 and θ_1,

$$\pi = (\alpha - \theta_0) + \phi + (\beta - \theta_1) = \alpha + \beta + \phi - (\theta_0 + \theta_1) \tag{1.3}$$

In $\Delta P_0 P_1 \mathrm{J}$, using the angle sum property of a triangle,

$$\theta_0 + \theta_1 = \pi - \phi \tag{1.4}$$

Using equation (1.4) to substitute $(\theta_0 + \theta_1)$ in equation (1.3), it follows that

$$2\pi = \alpha + \beta + 2\phi$$

or,

$$\phi = \pi - \frac{(\alpha + \beta)}{2}.$$

Hence, the angle subtended at $\overline{J}$ by $\overline{D}$ is constant. From the geometric property that the angle subtended by a chord on the circumference of a circle is constant, it is concluded that point $\overline{J}$ moves on the circumference of a circle as established by Sabin [14]. Hence, with given G^1 Hermite data, a family of biarcs is defined. This freedom in choosing the point $\overline{J}$ on the circumference of a circle represents the additional degree of freedom, mentioned earlier.

Several existing constraints have been discussed in the literature to determine a unique biarc. Some of them are,

1. Sabin [14] and Bolton [4], choose $\overline{J}$ to be the incentre of $\Delta P_0 \mathrm{P} P_1$.

2. Bolton [4], Su and Liu [15], Meek and Walton [16], minimize the difference of the two radii $|r_1 - r_0|$ of the two arcs of the biarc.

3. Bolton [4], Sabin [14], Su and Liu [15], Parkinson and Moreton [17], minimize the difference of the ratio of the radii of the two arcs to unity $\left|1 - \frac{r_1}{r_0}\right|$.

4. Bolton [4], Nutbourne and Martin [18], Su and Liu [15], Piegl and Tiller [19], minimize the difference in curvature $|\kappa_1 - \kappa_0|$ of the two arcs of the biarc.

Recently, Kimia et al. [1] also made use of the difference in curvature $|\kappa_1 - \kappa_0|$ of the biarc.

1.3.2 NURBS Curve Fitting

Except where otherwise mentioned, a restricted form of the NURBS curve, often called the Bézier rational form, will be used in this thesis. As stated earlier in section 1.3 (as proposed solution), the purpose here is to find a G^1 Hermite interpolating curve between two points such that if the G^1 Hermite data originate from a circular arc, the curve defaults to a circular arc. The user here does not know if the data comes from a circular arc or not.

Mathematically, a quadratic NURBS curve used in this thesis is represented by:

$$C(u) = \frac{(1-u)^2\overline{P}_0 w_0 + 2u(1-u)\overline{P}_1 w_1 + u^2\overline{P}_2 w_2}{(1-u)^2 w_0 + 2u(1-u)w_1 + u^2 w_2} \tag{1.5}$$

where $\overline{P}_i$, $i = 0, 1, 2$, are the control points and w_i, $i = 0, 1, 2$ are the weights at the respective control points. This form can be derived from a general NURBS curve with the knot vectors $\overline{U}$ (explained in Section 2.11).

For the solution strategy, a quadratic NURBS curve(say, $\overline{C}_L$), constrained to be a circular arc, is first drawn from $\overline{A}$, with initial tangent as $\overline{T}_A$, to point $\overline{B}$. Similarly a NURBS curve ($\overline{C}_R$) is constructed, from $\overline{A}$ to $\overline{B}$ with final tangent as $\overline{T}_B$. In general, blending the two curves $\overline{C}_L$ and $\overline{C}_R$, will yield a cubic NURBS curve as the two curves are distinct. However, if the G^1 Hermite data being interpolated belong to a circular arc, then $\overline{C}_L$ and $\overline{C}_R$ both will be that circular arc, and so will the blend of $\overline{C}_L$ and $\overline{C}_R$. A more detailed illustration of the process of linear blending is provided in section 5.2. When the degree of a polynomial or rational curve is increased without

changing the curve geometrically, the resulting curve is said to be a degree-raised or degree-elevated curve. For e.g. linear blending of two quadratic NURBS curves result in a degree-raised quadratic or cubic NURBS curve. Similarly, blending of two cubics would result in degree-raised cubic or a quartic NURBS curve. In rest of this thesis, term 'degree-raised quadratic NURBS' implies to a 'cubic NURBS curve'.

Illustration: NURBS Curve Fitting

Again, for the purpose of illustration $\overline{P}_0 = \overline{A}$, $\overline{T}_0 = \overline{T}_A$, $\overline{P}_1 = \overline{B}$ and $\overline{T}_1 = \overline{T}_B$. First draw two quadratic NURBS curve(Figure 1.2), $\overline{C}_L(u)$ say, from $\overline{P}_0$ to

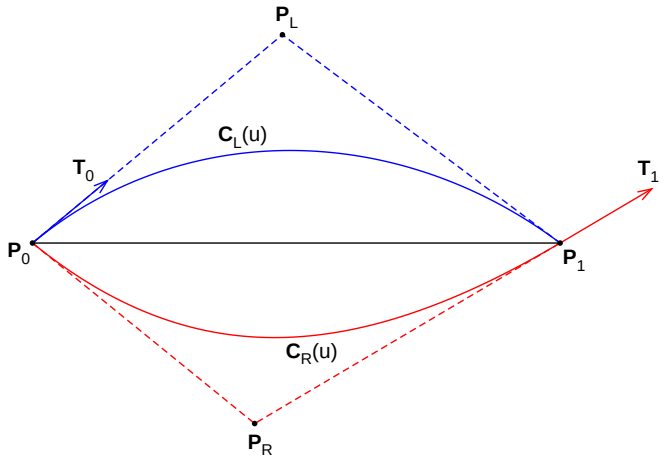

Figure 1.2: Quadratic NURBS curve drawn on data points.

$\overline{P}_1$ satisfying the tangent $\overline{T}_0$ and $\overline{C}_R(u)$ from $\overline{P}_0$ to $\overline{P}_1$ satisfying the tangent $\overline{T}_1$. Assuming that the quadratics are circular arcs, the other control points ($\overline{P}_L$ of $\overline{C}_L(u)$ and $\overline{P}_R$ of $\overline{C_R}(u)$) can be determined, in order to construct the NURBS curve. Then the linear blending $\overline{C}(u)$ of the two quadratics is performed. If the data belong to a circular arc, then a degree-raised quadratic ($\overline{C}_1(u)$) is obtained, otherwise a cubic

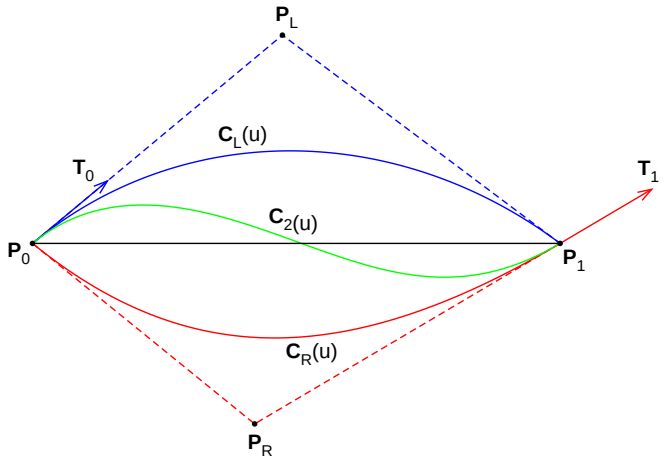

Figure 1.3: NURBS curve Fitting.

NURBS curve($\overline{C}_2(u)$) is obtained (see Figure 1.3).

Chapter 2

Background

2.1 Types of Curves

Curves are a major tool for drawing, with applications in architecture, engineering, industrial design and other CAD-oriented fields. They are important because of their ability to be modified, scaled or transformed using simple mathematical operations. Their ability to be represented mathematically makes them one of the most powerful computer-aided drawing tools.

Curves can be represented in three basic categories, based on the way they are expressed mathematically, as follows:

Implicit: Any point on the curve (x, y), satisfies a function $F(x, y) = 0$.

Parametric: The coordinates of points on the curve are represented in terms of functions of a variable or a parameter (say u), i.e., $C(u) = (x(u), y(u))$.

Non-parametric explicit: Coordinates of points on the curve are related by $y =$

$f(x)$ or $x = g(y)$. More details about the curves and their categories can be found in Marsh [20].

2.2 Desirable Properties of Curves

The properties listed below are desirable for the curves used in CAD/CAM. Biarcs, Béziers, B-Splines and NURBS curves are defined or constructed such that they have these properties, with the exceptions that Bézier curves are not locally controllable and not all biarcs satisfy the convex hull property.

- *Affine Invariance:* they are invariant under affine transformations applied to the control points.
- *Convex Hull:* curve $\overline{C}(u)$ lies within the convex hull of the control polygon.
- *Local Control:* if the control point $\overline{P}_i$ or the weight w_i is changed, only a part of the curve is affected.
- *Coincident with start and end point:* $\overline{C}(0) = \overline{P}_0$ and $\overline{C}(1) = \overline{P}_n$, where $\overline{P}_0$ is the first control point and $\overline{P}_n$ the last.

2.3 Curvature

The curvature κ of a curve $\overline{C}(u) = (X(u), Y(u))$ is defined as the rate at which the unit tangent vector changes direction and expressed mathematically as,

$$\kappa = \frac{\dot{X}(u)\ddot{Y}(u) - \ddot{X}(u)\dot{Y}(u)}{[\dot{X}^2(u) + \dot{Y}^2(u)]^{\frac{3}{2}}}.$$

2.4 Cusps

A cusp on a curve is defined as a point on the curve where the first derivative vector vanishes causing the curvature to be undefined, and the tangent changes direction by π. For a given parametric curve $C(u)$, a cusp on the curve at the parameter $u = u_0$ exists iff,

$$\lim_{u \to u_0^-} \frac{\dot{C}(u)}{|\dot{C}(u)|} = - \lim_{u \to u_0^+} \frac{\dot{C}(u)}{|\dot{C}(u)|}.$$

2.5 Inflection Points

An inflection point on a curve is defined as the point at which the curve does not have a cusp and the tangent at that point passes through the curve Piegl and Tiller [21]. Inflection point can also be defined simply as a point where a curve crosses

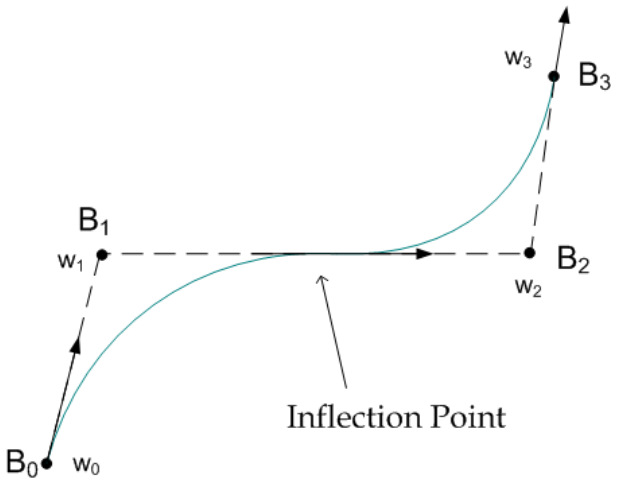

Figure 2.1: A cubic NURBS curve with an inflection point.

from one side to the other side of its tangent line at that point, as shown in Figure 2.1.

2.6 Two-point G^1 Hermite data

As mentioned earlier (refer to section 1.3), the term 'tangent' will be taken to mean 'unit tangent vector' throughout this thesis. A tangent specifies a direction. A set of two points with tangents on each of them is defined as two-point G^1 Hermite data. Figure 2.2 shows points $\overline{A}$ and $\overline{B}$ with tangents $\overline{T}_A$ and $\overline{T}_B$ respectively. Tangent $\overline{T}_A$ is aligned with x-axis while, W is the angle that tangent $\overline{T}_B$ makes with the x-axis.

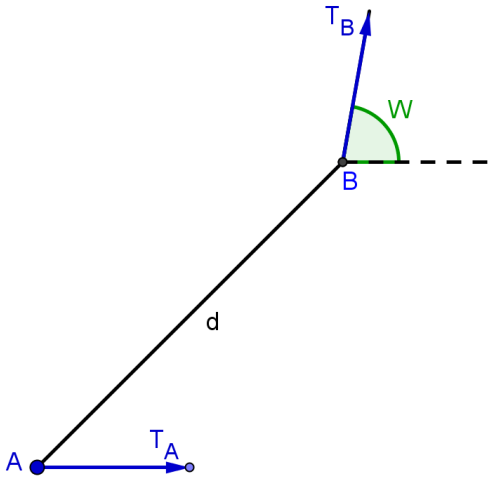

Figure 2.2: Two-point G^1 Hermite data.

2.7 Shape-preserving Interpolation

The technique of constructing a curve through a given set of data points such that the shape of the data set is preserved, is called shape-preserving interpolation. The interpolating curve has to pass through each point of the data set. This technique has its application in the field of Computer Aided Design (CAD) for the design of

various shapes and, data analysis wherein, interpolation is used to summarize relation between two or more variables. The examples that will be used to illustrate the application (Chapter 8) of the proposed work in this thesis, are shape-preserving.

2.8 Biarc

Consider Figure 2.3. Given two points $\overline{A}$ and $\overline{B}$ with associated tangents $\overline{T}_A$ and $\overline{T}_B$ respectively, a G^1 biarc that matches these G^1 Hermite data must satisfy the following three conditions:

1. It must have $\overline{A}$ as the start point and $\overline{B}$ as the end point.
2. It must match the corresponding tangents $\overline{T}_A$ and $\overline{T}_B$ at those points.
3. The join point $\overline{J}$ of the two arcs should maintain tangent continuity (also called G^1 continuity).

In order to construct a biarc, radii and centres of the two arcs have to be determined. A curve that consists of two or more circular arcs or line segments is called an arc spline. An arc spline is usually constructed using biarcs. Several methods have been introduced over the years to determine an arc spline through a given set of points [3, 4, 16, 17, 19, 22].

2.9 The Bézier Curve

Introduced first by P. Bézier, a Bézier curve is defined as follows [20]:

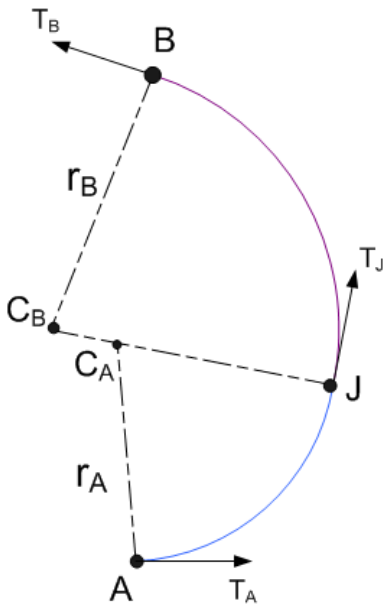

Figure 2.3: A biarc.

Given $n+1$ *control points* $\overline{P}_0,\overline{P}_1,\ldots,\overline{P}_n$, the Bézier curve of degree n is defined as

$$\overline{C}(u) = \sum_{i=0}^{n} B_{i,n}(u)\overline{P}_i \quad \text{where } 0 \le u \le 1,$$

and

$$B_{i,n}(u) = \begin{cases} \dfrac{n!}{(n-i)!i!}(1-u)^{n-i}u^i & \text{if } 0 \le i \le n; \\ 0 & \text{otherwise} \end{cases}$$

are the *Bernstein polynomials* of degree n.

The *control polygon* is the polygon with the control points as its vertices. Control points determine the shape of the curve; manipulation of even a single point would alter its shape.

A linear Bézier curve has degree $n = 1$. It is a line segment joining the two control

points $\overline{P}_0$ and $\overline{P}_1$ and defined by the linear equation:

$$\overline{C}(u) = (1-u)\overline{P}_0 + u\overline{P}_1, \text{ for } u \in [0,1].$$

A quadratic Bézier curve has degree $n = 2$ with the control points $\overline{P}_0$, $\overline{P}_1$ and $\overline{P}_2$:

$$\overline{C}(u) = (1-u)^2\overline{P}_0 + 2(1-u)u\overline{P}_1 + u^2\overline{P}_2, \text{ for } u \in [0,1].$$

A cubic Bézier curve with degree $n = 3$ and $\overline{P}_0$, $\overline{P}_1$, $\overline{P}_2$ and $\overline{P}_3$ as the control points is defined as,

$$\overline{C}(u) = (1-u)^3\overline{P}_0 + 3(1-u)^2u\overline{P}_1 + 3(1-u)u^2\overline{P}_2 + u^3\overline{P}_3 \text{ for } u \in [0,1].$$

From the above equations, it can be observed that as the degree of the Bézier increases, the complexity of the equations increase. This is the reason why Bézier curves with low degrees are preferred for curve design.

2.10 B-splines

A B-spline is a piecewise polynomial curve. A B-spline of degree d (or order $d+1$) with control points $\overline{P}_0,\ldots,\overline{P}_n$ and knot set $u = \{u_0,\ldots,u_m\}$ is defined to include the interval $[a,b] = [u_d, u_{m-d}]$ as

$$\overline{C}(u) = \sum_{i=0}^{n} \overline{P}_i N_{i,d}(u) \tag{2.1}$$

where $N_{i,d}(u)$ are the B-spline basis functions.

The B-spline basis function $N_{i,d}(u)$ of degree d are defined recursively in terms of the knot vectors $u_0 \leq u_1 \leq u_2 \leq \ldots \leq u_m$ by the Cox-de Boor algorithm [20],

$$N_{i,0}(u) = \left\{ \begin{array}{ll} 1 & \text{if } u \in [u_i, u_{i+1}) \\ 0 & \text{otherwise,} \end{array} \right\},$$

$$N_{i,d}(u) = \frac{u - u_i}{u_{i+d} - u_i} N_{i,d-1}(u) + \frac{u_{i+d+1} - u}{u_{i+d+1} - u_{i+1}} N_{i+1,d-1}(u)$$

for $i = 0, 1, 2, \ldots, n$ and $d \geq 1$. The B-spline basis functions $N_{i,d}(u)$ have the interesting properties that all $N_{i,d}(u) > 0$ (positivity) and $N_{i,d}(u) > 0$ elsewhere (local support); all are piecewise polynomial functions of degree d and, all the functions sum up to unity in a given interval.

These properties cause B-spline curves, which use B-spline functions as blending functions, to have the following which are useful for curve design.

- *Local Control:* Each curve segment is defined by $d + 1$ control points and can be evaluated individually.
- *Convex Hull:* are defined within the convex hull of the control polygon.
- *Affine Transformation:* are invariant under affine transformation.

The basis functions, are related to the Bernstein polynomials of Bézier curves. The basis functions of the B-spline of order $d + 1$ defined over the knot vector U defined by

$$u_0 = u_1 \ldots = u_d = 0, u_{d+1} = u_{d+2} \ldots = u_{2d+1} = 1.$$

are exactly equal to the Bernstein polynomials of degree d. For instance, suppose a cubic B-spline curve of degree 3 is defined with a knot vector U,

$$U = (0, 0, 0, 0, 1, 1, 1, 1).$$

The four basis functions $N_{0,4}$, $N_{1,4}$, $N_{2,4}$ and $N_{3,4}$ are the same as the Bernstein polynomials $B_{0,3}$, $B_{1,3}$, $B_{2,3}$ and $B_{3,3}$ of a Bézier curve with degree 3. The Cox-de

Boor algorithm yields

$$N_{0,4}(u) = (1-u)^3,$$

$$N_{1,4}(u) = 3u(1-u)^2,$$

$$N_{2,4}(u) = 3u^2(1-u),$$

$$N_{3,4}(u) = u^3.$$

This suggests that B-splines could be constrained to form a Bézier curve or in other words, B-spline functions are the generalization of Bézier curves. A more detailed illustration of transformations between the Bernstein polynomials and B-spline functions can be found in Bartels et al. [23]

2.11 Non Uniform Rational B-spline Curves (NURBS)

A rational form of a B-spline curve is defined as a Non Uniform Rational B-spline curve. Piegl [24] defines a NURBS curve of degree d as,

$$\overline{C}(u) = \frac{\sum_{i=0}^{n} N_{i,d}(u) w_i \overline{P}_i}{\sum_{i=0}^{n} N_{i,d}(u) w_i}, \quad a \leq u \leq b. \tag{2.2}$$

where $\overline{P}_i$ are the control points, w_i are the weights at those points and $N_{i,d}(u)$ are the B-spline basis functions of degree d defined in Section 2.10. These curves are referred to as being *non uniform* because of the non uniform distribution of knots. Figure 2.4 illustrate the affect in the shape of a quadratic NURBS curve with the change in position of a control point ($\overline{B}_1$). Figure 2.5 (similar to Figure 7.12 in Piegl and Tiller [21]) demonstrates the affect of the change in the weight of control point B_1, on the shape of a quadratic NURBS curves. A cubic NURBS curve with an inflection point is illustrated in Figure 2.1 .

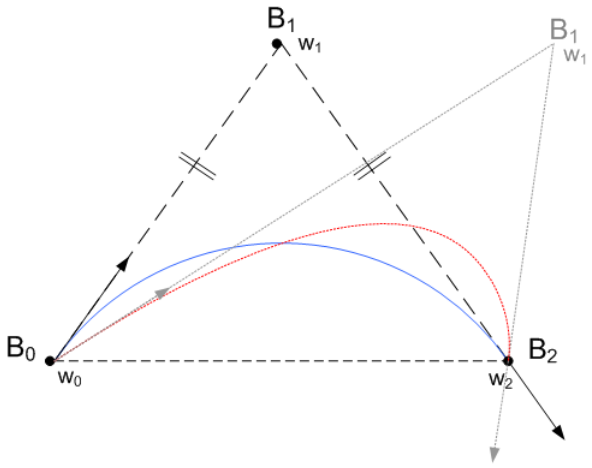

Figure 2.4: Effect of control points on a quadratic NURBS curve.

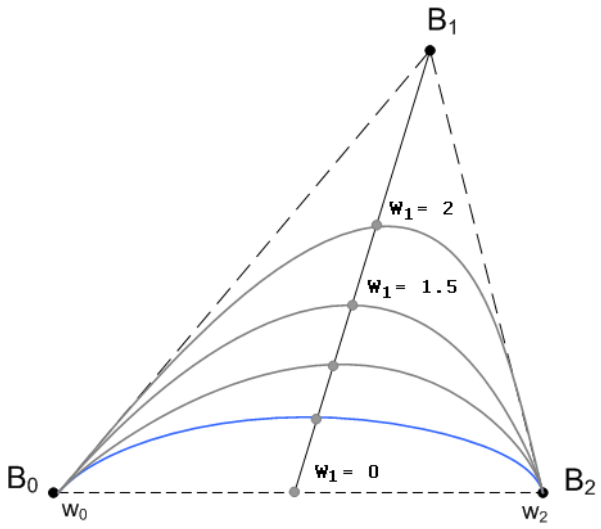

Figure 2.5: Effect of change in weight on a quadratic NURBS curve.

The NURBS curve and B-splines curves discussed above share a relationship. Given a NURBS curve defined by equation (2.2); if the weights w_i on the control points P_i are all reduced to unity i.e. $w_1 = w_2 = w_3 \ldots = w_n = 1$ then, the

denominator of (2.2) is

$$\sum_{i=0}^{n} N_{i,d}(u)w_i = \sum_{i=0}^{n} N_{i,d}(u).$$

It is a known identity that

$$\sum_{i=0}^{n} N_{i,d}(u) = 1 \text{ (Piegl [21])}.$$

Hence, the NURBS (2.2) curve would reduce to,

$$\overline{C}(u) = \sum_{i=0}^{n} N_{i,d}(u)\overline{P}_i, \quad a \leq u \leq b \tag{2.3}$$

The above equation, when compared to the equation (2.1) of B-splines, is found to be exactly the same.

Hence, a NURBS curve can be constrained to yield a B-spline while, a B-spline reduces to a Bézier curve under certain restrictions. A brief survey on NURBS curve, its properties and its use in curves and surfaces can be found in Piegl [24]. Throughout this thesis, the term NURBS will refer to the following limited cases of the curve,

$$\overline{C}(u) = \frac{(1-u)^2\overline{P}_0 w_0 + 2u(1-u)\overline{P}_1 w_1 + u^2\overline{P}_2 w_2}{(1-u)^2 w_0 + 2u(1-u)w_1 + u^2 w_2}$$

or,

$$\overline{C}(u) = \frac{(1-u)^3\overline{P^*}_0 w_0^* + 3u(1-u)^2\overline{P^*}_1 w_1^* + 3u^2(1-u)\overline{P^*}_2 w_2^* + u^3\overline{P^*}_3 w_3^*}{(1-u)^3 w_0^* + 3u(1-u)^2 w_1^* + 3u^2(1-u)w_2^* + u^3 w_3^*} \text{ for } u \in [0,1].$$

2.12 Tolerance Limit (ϵ)

Tolerance limit is defined as the maximum desirable distance of an approximating spline from a given curve. The tolerance limit is provided by the user as an input parameter. In this thesis, it is denoted by the symbol ϵ.

Chapter 3

Related Work

This section discusses the work that has been done in approximations using biarcs and shape-preserving interpolation schemes.

3.1 Biarcs

In all the approximation schemes mentioned in this section, the term tolerance limit is used often. During the approximation of curves with arc splines, the arc splines are usually not the same as the original curve, instead, they are at a certain distance from the curve. These approximation schemes are then tuned to generate arc splines at the best proximity to the original curve. The tolerance limit (section 2.12) is provided by the user as an input parameter, before approximating the curve.

3.1.1 Approximation of discrete data points

Piegl [25] introduced one of the early methods for the approximation of data points using circular arcs and straight lines. The scheme tried to find a piecewise circular curve closest to the data set using a minimum number of segments. This scheme is based on simple trigonometric calculations and was able to do good approximations. Piegl implemented it on a shoe-sole data set and the accuracy of his method was found to be acceptable.

Piegl and Tiller [19], [11] use the parametric model of a biarc for the approximation of data points. A parametric model defines a control polygon for each arc of the biarc. Since, the two arcs share a common point (the point of join) the biarc curve is defined by $3+3-1=5$ control points. This model is coordinate independent. The method is tested for the approximation of NURBS curve. In order to approximate, parametric polygonal decomposition of the NURBS curve is performed. It then creates an offset of the polygon by half the tolerance so as to obtain a fat polygon. A G^1 biarc is then constructed which lies within the fat polygon. Based on the results shown, the scheme seems to perform well for a very small tolerance limit. Yang and Chen [26] define a similar model however, the authors further derive the formulae for the angles subtended by the two arcs of the biarc. This is done to achieve a mathematical relation between the curvatures of the biarc and the gouging interval. Gouging occurs when a tool overcuts a part due to geometric mismatch. A gouging interval is required to avoid such a situation. The method generates gouging free biarcs as compared to previous methods which did not consider gouging.

Meek and Walton [16] followed a geometrical approach to the problem of con-

structing a biarc for two-point G^1 Hermite data as in section 2.6. Let W be given as the angle $\overline{T}_B$ makes with the x-axis, θ as the angle subtended at the centre of the first circular arc, then $W - \theta$ is the angle at the centre of the second arc. Geometrical properties were used to obtain the length of the two arcs in terms of θ and $W - \theta$. Furthermore, the signs of these angles determine whether the biarc is a C-shaped or an S-shaped curve. Based on this scheme, two arc-spline construction algorithms were introduced and were tested on data sets representing a circle and a shoe-sole. The results were compared to previous methods for efficiency and accuracy.

3.1.2 Family of biarcs

Yang and Chen [26] defined the whole family of biarcs matching two-point G^1 Hermite data in terms of a parameter 'k'. The value of 'k' was defined in terms of the control points of the biarc and the given data points. The authors used these values to further derive the equations for the centres and the radii of the two arcs of the biarc. Meek and Walton [12] expanded on their geometrical model to mathematically define a family of biarcs for two-point G^1 Hermite data.

3.1.3 Approximation of parametric curves

Walton and Meek [5] approximate a quadratic Bézier curve by arc splines. Let $\overline{P}_0$, $\overline{P}_1$ and $\overline{P}_2$ be the control points of a quadratic Bézier curve, then beginning and ending points and tangents of a biarc are chosen to coincide with the beginning and ending points of the Bézier curve. To constrain the choice to a unique biarc of the family of biarcs possible through the given points, they assume that the tangent at

the joint is parallel to the vector $\overline{P}_2 - \overline{P}_0$. This also ensures that the biarc is C-shaped and within the convex hull of the control polygon. The authors prove mathematically that a unique biarc is in fact possible. The method is coordinate independent.

Meek and Walton [6] approximate NURBS curves by arc splines. Given the points $\overline{A}$ and $\overline{B}$ and the tangents $\overline{T}_A$ and $\overline{T}_B$ at those points, bounding curves C_A and C_B can be defined as follows. Bounding curve C_A is a circular arc that passes from $\overline{A}$ to $\overline{B}$ and matches direction $\overline{T}_A$ at $\overline{A}$, and C_B as an arc from $\overline{A}$ to $\overline{B}$ matching $\overline{T}_B$ at $\overline{B}$. So, the family of biarcs that interpolate to G^1 Hermite data are within the region defined by the bounding curves C_A and C_B of those points. For α and β defined as the angles from $\overline{T}_A$ to the vector $\overline{B} - \overline{A}$ and from $\overline{B} - \overline{A}$ to $\overline{T_B}$, a set of conditions are defined to determine the shape of the resulting curve, i.e. C-shaped, S-shaped or a straight line. In order to create an arc spline that approximates a NURBS curve to given accuracy, the curve is divided such that the distance between the bounding curves is within a predefined tolerance limit. A unique biarc is chosen on the basis of whichever gives a best C-shaped arc (here, α and β have the same sign). For the numerical tests, the method was used for the approximation of quadratic NURBS (parabola, hyperbola and ellipse) curve. The approximate distance between the arc spline and NURBS curve, and the number of biarcs used for the approximation, are then determined for a given distance between the bounding arcs (or tolerance limit).

A planar cubic Bézier spiral is approximated by circular arcs in Walton and Meek [8]. The deviation distance is the distance between the arc spline and the curve to be approximated. It is calculated along the radial direction at the point of join of the two arcs of the biarc. This approach is based on calculating the deviation

distance of the biarc from the cubic Bézier spiral. Biarc approximation is performed by recursive subdivision of the spiral and approximating it with the biarc. The subdivision is performed until the biarc is within the deviation (or tolerance) of the Bézier segment.

3.2 Shape preserving

In the above work, the biarc approximation is considered separately from the shape preserving problem. The research in this thesis merges the two problems and additionally fits a NURBS curve such that it naturally defaults to a circular arc if data from a circular arc are specified.

Goodman and Unsworth [27] did early work on shape preserving interpolation of data points using continuous-curvature parametric curves. The scheme defines a tangent $\overline{T}_i$ and curvature κ_i at each point, and then derives curve segments $\overline{Q}(u)$ for each pair of neighbouring points. Goodman and Unsworth then discuss a set of properties for the interpolating curve, which according to the authors, are desired of the scheme. They are: the curve should have a minimum number of inflection points; the scheme should be local, i.e., a change in one or more points should only have an effect in a small portion of the curve, and the curve should be invariant under affine transformations. The method can also be modified to interpolate points arising from a function. A condition was also considered where the given data sets had collinear points. This helped in better generalizing the scheme to interpolate an arbitrary set of data. Foley et al. [28] extend the work of Goodman and Unsworth [27] by providing a scheme that obtains the curvature value at the point of interpolation automatically

or user initiated (if required). The method also provides a lower bound on the value of the curvature that can be used.

Seymour and Unsworth [29] made use of rational cubic splines for interpolation. The curve satisfies the same set of properties as defined in [28]. However, the authors define a different set of properties for interactive shape modification and interpolation. They are as follows. The curve should preserve G^2 continuity; it should be local convexity preserving; the effect on the shape should be intuitive; and the changes to the shape of the curve should be local. In order to interactively change the shape of the interpolating curve, method adjusts Bézier control points $\overline{P}_i$ and weights w_i on them together with curvatures κ_i defined on the data points. The scheme also proves to be effective tool for observing the behaviour of Bézier with the change in the control points.

Duan et al. [30] make use of rational splines to interactively change the shape of the interpolating curve without affecting the given data sets. The authors further discuss two significant properties that are desirable of these curves. The first is to be able to control the convexity of the curve and the second, to control the smoothness of the curve. The calculations involved in defining the interpolating curve are reduced by choosing rational splines with linear denominators. Further, the two significant properties of the curve (mentioned above) depend on the second order derivative, which is easier to calculate for a linear denominator as compared to cubic or quadratic denominator. Duan et al. then derive a set of conditions (or inequalities) in order to alter the convexity of rational cubic splines mathematically.

Spirals are defined as curves with a monotonely increasing or decreasing curvature.

Goodman and Meek [31] introduce a scheme of planar interpolation using a pair of rational spirals. The spiral segments considered are having zero curvatures at one of the ends. This work addresses two situations where the two-point G^2 Hermite interpolation problem can be solved with a pair of rational spiral segments. A C-shaped pair of spiral segments are formed when they meet at the endpoints having non-zero curvature, with G^2 continuity. An S-shaped curvature is formed when the two spirals of opposite signed curvature meet with the same continuity. The method has its applications in creating C or S-shaped transition curves that require G^2 continuity.

Lavery [32] uses first-derivative based parametric and non parametric cubic L_1 spline curves for shape-preserving interpolation. Lavery introduces various types of parametric L_1 and L_2 splines based on first-derivative and second-derivative expressions. A regularization parameter ϵ is introduced to establish the uniqueness of the spline and its closeness to the data set being interpolated. These splines are then compared with each other. The investigation revealed that the first-derivative based L_1 spline curves best preserve the shape of an irregular data sets.

Chapter 4

Biarc Fitting

There is a one-parameter family of biarcs that interpolates to given two-point G^1 Hermite data. In this thesis, the parameter θ, which is the angle subtended by the first arc at its centre of the biarc, is used as that parameter. For biarc G^1 Hermite data, $\overline{B}_0$, $\overline{T}_0$, $\overline{B}_2$, $\overline{T}_2$ will be used instead of $\overline{A}$, $\overline{T}_A$, $\overline{B}$, $\overline{T}_B$ of sections 1.2, 1.3, 2.8, 2.6 and 3.1. The shape of the biarc depends on the tangents at the two points and the vectors connecting the two points. If $\overline{D} = \overline{B}_2 - \overline{B}_0$ then, the shape of the final biarc depends on the signs of the scalars $\overline{T}_0 \times \overline{D}$ and $\overline{D} \times \overline{T}_2$. If $(\overline{T}_0 \times \overline{D}) \cdot (\overline{D} \times \overline{T}_2) > 0$ then a C-shaped arc is expected, else an S-shaped biarc is obtained. The emphasis is laid on the C-shaped curves as the quadratic NURBS curves (discussed in next chapter) being used in NURBS curve fitting are C-shaped.

The rest of the chapter explains in detail the biarc fitting model introduced in this work and provides the algorithm used to implement this scheme.

4.1 Biarc Fitting: Mathematical Model

Consider Figure 4.1. The point $\overline{B}_1$ is defined as the point of intersection of the line through $\overline{B}_0$ parallel to the tangent $\overline{T}_0$ and the line through $\overline{B}_2$ parallel to the tangent $\overline{T}_2$. In order to construct the biarc, the following parameters are defined: h is the length of the vector $\overline{B}_1 - \overline{B}_0$, k is the length of the vector $\overline{B}_2 - \overline{B}_1$, θ is the angle from vector $\overline{B_0B_1}$ to the tangent $\overline{T}_J$ at the point of join $\overline{J}$, and W is the angle from vector $\overline{B_0B_1}$ to vector $\overline{B_1B_2}$. The tangent at $\overline{J}$ intersect $\overline{B_0B_1}$ at $\overline{X}$ and $\overline{B_1B_2}$ at $\overline{Y}$. Angle ψ is defined as the angle between vectors $\overline{B_0B_1}$ and $\overline{B_0B_2}$. Let θ_0 and r_0 be the span angle and radius of the first circular arc and, θ_1 and r_1 of that of the second circular arc respectively. First the angles θ_0 and θ_1, then the radii r_0 and r_1, are determined.

Now, as $\overline{B_0X}$ and $\overline{XJ}$ are tangent to the same circular arc B_0J, $\overline{C_0B_0}$ is perpendicular to $\overline{B_0X}$, as the tangent at a given point on a circular arc is perpendicular to its radius at that point. Similarly, $\overline{C_0J}$ is perpendicular to $\overline{JX}$. Therefore,

$$\angle C_0B_0X = \angle XJC_0 = \frac{\pi}{2}. \tag{4.1}$$

By the geometrical property of a polygon, the sum of the interior angles of a closed polygon with n sides is given as $(n-2)\pi$. In polygon C_0B_0XJ,

$$\angle C_0B_0X + \angle B_0XJ + \angle XJC_0 + \angle JC_0B_0 = 2\pi,$$

$$\text{Using equation (4.1)}, \frac{\pi}{2} + (\pi - \theta) + \frac{\pi}{2} + \theta_0 = 2\pi,$$

$$\text{Therefore, } \theta_0 = \theta. \tag{4.2}$$

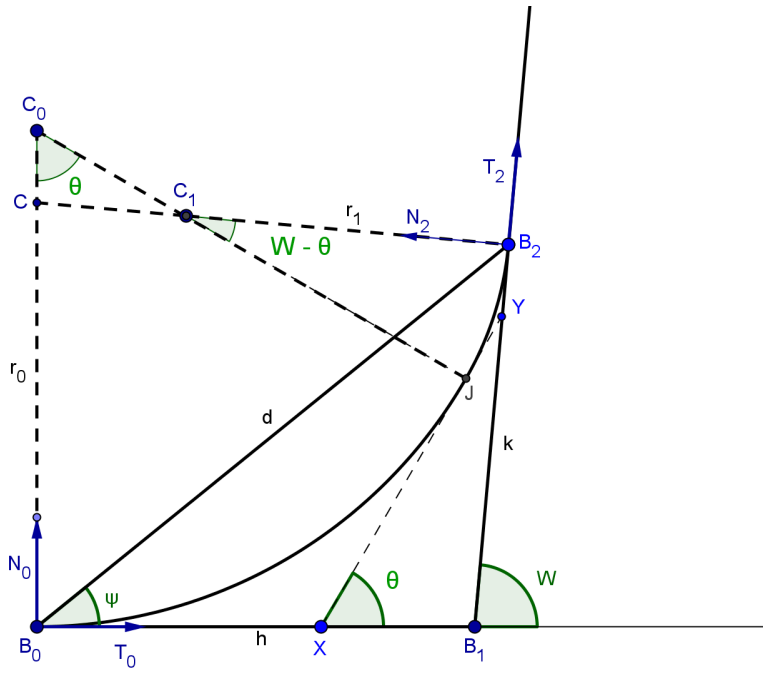

Figure 4.1: Biarc with h, k, W and θ.

Similarly, in polygon C_1JYB_2, since $\angle JYB_1 = W - \theta$ it follows that

$$\theta_1 = W - \theta. \tag{4.3}$$

In order to construct the biarc between the two points $\overline{B}_0$ and $\overline{B}_2$, it is desirable to find expressions for r_0 and r_1 in terms of the given parameters h, k, θ and W.

Consider $\triangle CC_1C_0$ in Figure 4.1, $|\overline{C_1C_0}| = r_0 - r_1$, $\angle C_1C_0C = \theta$ and $\angle CC_1C_0 = W - \theta$. In $\triangle CC_1C_0$, $\theta + (W - \theta) + \angle C_0CC_1 = \pi$, therefore $\angle C_0CC_1 = \pi - W$, then using the sine law,

$$\frac{r_0 - r_1}{\sin(\pi - W)} = \frac{|\overline{CC}_1|}{\sin\theta},$$

$$\text{or, } |\overline{CC}_1| = \frac{\sin\theta}{\sin W} \cdot (r_0 - r_1). \tag{4.4}$$

$$\text{Similarly, } |\overline{C_0C}| = \frac{\sin(W-\theta)}{\sin W} \cdot (r_0 - r_1). \tag{4.5}$$

$$\begin{aligned} \text{Furthermore, } |\overline{CB_2}| &= |\overline{CC_1}| + r_1, \\ &= \left(\frac{\sin\theta}{\sin W} \cdot (r_0 - r_1) + r_1\right). \end{aligned} \tag{4.6}$$

$$\begin{aligned} |\overline{CB_0}| &= |\overline{C_0B_0}| - |\overline{C_0C}|, \\ &= r_0 - \left(\frac{\sin(W-\theta)}{\sin W} \cdot (r_0 - r_1)\right). \end{aligned} \tag{4.7}$$

In $\triangle CB_0B_2$, let $|\overline{B_2B_0}| = d$, let, $\overline{N}_0$ be the unit normal vector, perpendicular to the tangent $\overline{T}_0$, from point $\overline{B}_0$ towards the centre $\overline{C}_0$ and let $\overline{N}_2$ be the unit normal at $\overline{B}_2$ in a similar fashion. Using the vector-sum property, in $\triangle CB_0B_2$,

$$\begin{aligned} \overline{B}_2 - \overline{B}_0 &= (|\overline{CB_0}|)\overline{N}_0 + (|\overline{CB_2}|)(-\overline{N}_2), \\ &= (|\overline{CB_0}|)\overline{N}_0 - (|\overline{CB_2}|)\overline{N}_2. \end{aligned}$$

Putting values from equations (4.6) and (4.7) yields,

$$\begin{aligned} \overline{B_2} - \overline{B}_0 = &\left[r_0 - \frac{\sin(W-\theta)}{\sin W} \cdot (r_0 - r_1)\right] \overline{N}_0 \\ &- \left[r_1 + \frac{\sin\theta}{\sin W} \cdot (r_0 - r_1)\right] \overline{N}_2. \end{aligned}$$

Expanding,

$$\begin{aligned} \overline{B}_2 - \overline{B}_0 = r_0\overline{N}_0 - r_1\overline{N}_2 &- \left(\frac{\sin(W-\theta)}{\sin W} \cdot (r_0 - r_1)\right) \overline{N}_0 \\ &- \left(\frac{\sin\theta}{\sin W} \cdot (r_0 - r_1)\right) \overline{N}_2. \end{aligned} \tag{4.8}$$

In $\triangle CC_1C_0$, using the vector-sum property, it can be derived that,

$$|\overline{CC_1}|\overline{N}_2 + |\overline{C_0C}|\overline{N}_0 = (r_0 - r_1)\overline{N}. \tag{4.9}$$

where, $\overline{N}$ is the unit normal vector $\perp$ to the tangent $\overline{T}_J$ at the point of join $\overline{J}$, of the two circular arcs of the biarc. Putting values from equations (4.4) and (4.5) in equation (4.9),

$$\left(\frac{\sin\theta}{\sin W}\cdot(r_0-r_1)\right)\overline{N}_2+\left(\frac{\sin(W-\theta)}{\sin W}\cdot(r_0-r_1)\right)\overline{N}_0=(r_0-r_1)\overline{N}. \tag{4.10}$$

Replacing values from equation (4.10) into equation (4.8), a simpler expression is obtained,

$$\overline{B}_2-\overline{B}_0=r_0\overline{N}_0-r_1\overline{N}_2+(r_1-r_0)\overline{N}. \tag{4.11}$$

Similarly, applying the vector sum property to $\triangle B_0B_1B_2$,

$$\overline{B}_2-\overline{B}_0=h\overline{T}_0+k\overline{T}_2. \tag{4.12}$$

Comparing equations (4.11) and (4.12) we obtain;

$$r_0\overline{N}_0-r_1\overline{N}_2+(r_1-r_0)\overline{N}=h\overline{T}_0+k\overline{T}_2. \tag{4.13}$$

The dot product of equation (4.13) with $\overline{N}_0$ yields

$$r_0(\overline{N}_0\cdot\overline{N}_0)-r_1(\overline{N}_2\cdot\overline{N}_0)+(r_1-r_0)(\overline{N}\cdot\overline{N}_0)=h(\overline{T}_0\cdot\overline{N}_0)+k(\overline{T}_2\cdot\overline{N}_0). \tag{4.14}$$

The dot product of equation (4.13) with $\overline{T}_0$ yields

$$r_0(\overline{N}_0\cdot\overline{T}_0)-r_1(\overline{N}_2\cdot\overline{T}_0)+(r_1-r_0)(\overline{N}\cdot\overline{T}_0)=h(\overline{T}_0\cdot\overline{T}_0)+k(\overline{T}_2\cdot\overline{T}_0). \tag{4.15}$$

To solve equations (4.14) and (4.15), the following values of dot products are used:

$$\begin{aligned}
\overline{N}_0 \cdot \overline{N}_2 &= \cos W, \\
\overline{T}_2 \cdot \overline{N}_0 &= \cos\left(\frac{\pi}{2} - W\right) = \sin W, \\
\overline{N} \cdot \overline{N}_0 &= \cos\theta, \\
\overline{N}_2 \cdot \overline{T}_0 &= \cos\left(\frac{\pi}{2} + W\right) = -\sin W, \\
\overline{N} \cdot \overline{T}_0 &= \cos\left(\frac{\pi}{2} + \theta\right) = -\sin\theta, \\
\overline{T}_2 \cdot \overline{T}_0 &= \cos W.
\end{aligned}$$

Putting the above values in equation (4.14):

$$r_0 - r_1 \cos W + (r_1 - r_0)\cos\theta = k \sin W.$$

$$\therefore\ r_0(1 - \cos\theta) + r_1(\cos\theta - \cos W) = k \sin W. \tag{4.16}$$

Similarly for equation (4.15):

$$r_1 \sin W - (r_1 - r_0)\sin\theta = h + k\cos W,$$

$$\therefore\ r_0 \sin\theta + r_1(\sin W - \sin\theta) = h + k\cos W. \tag{4.17}$$

(4.16) and (4.17) are two linear equations for the two unknowns r_0 and r_1. The

determinant of the system is,

$$\begin{vmatrix} 1-\cos\theta & \cos\theta-\cos W \\ \sin\theta & \sin W-\sin\theta \end{vmatrix}$$

$$= \begin{vmatrix} 2\sin^2\frac{\theta}{2} & 2\sin\frac{W+\theta}{2}\sin\frac{W-\theta}{2} \\ 2\sin\frac{\theta}{2}\cos\frac{\theta}{2} & 2\cos\frac{W+\theta}{2}\sin\frac{W-\theta}{2} \end{vmatrix}$$

$$= 4\sin\frac{\theta}{2}\sin\frac{W-\theta}{2}\left[\sin\frac{\theta}{2}\cos\frac{W+\theta}{2}-\cos\frac{\theta}{2}\sin\frac{W+\theta}{2}\right],$$

$$= -4\sin\frac{W}{2}\sin\frac{\theta}{2}\sin\frac{W-\theta}{2}. \tag{4.18}$$

A non-zero determinant ensures a unique solution of the linear system. The restrictions $0<\theta<2\pi$, $0<W<2\pi$, $\theta<W$ and $W-\theta<2\pi$ ensure a unique solution for r_0 and r_1. Solving for r_0, r_1

$$r_0 = \frac{\begin{vmatrix} k\sin W & \cos\theta-\cos W \\ h+k\cos W & \sin W-\sin\theta \end{vmatrix}}{-4\sin\frac{W}{2}\sin\frac{\theta}{2}\sin\frac{W-\theta}{2}} = \frac{\begin{vmatrix} k\sin W & \sin\frac{W+\theta}{2} \\ h+k\cos W & \cos\frac{W+\theta}{2} \end{vmatrix}}{-2\sin\frac{W}{2}\sin\frac{\theta}{2}},$$

$$= \frac{k\sin\frac{W-\theta}{2}-h\sin\frac{W+\theta}{2}}{-2\sin\frac{W}{2}\sin\frac{\theta}{2}}. \tag{4.19}$$

$$r_1 = \frac{\begin{vmatrix} 1-\cos\theta & k\sin W \\ \sin\theta & h+k\cos W \end{vmatrix}}{-4\sin\frac{W}{2}\sin\frac{\theta}{2}\sin\frac{W-\theta}{2}} = \frac{\begin{vmatrix} \sin\frac{\theta}{2} & k\sin W \\ \cos\frac{\theta}{2} & h+k\cos W \end{vmatrix}}{-2\sin\frac{W}{2}\sin\frac{W-\theta}{2}},$$

$$= \frac{h\sin\frac{\theta}{2} - k\sin\left(W-\frac{\theta}{2}\right)}{-2\sin\frac{W}{2}\sin\frac{W-\theta}{2}}. \tag{4.20}$$

In order to determine the acceptable range of parameter θ, equations similar to the work by Meek and Walton [16] are derived. Recall that ψ is the angle from tangent $\overline{T}_0$ to the vector $\overline{B_0B_2}$, and $d = |\overline{B_2B_0}|$. The parameters h and k will now be expressed in terms of ψ and d. In $\triangle B_0B_1B_2$,

$$d\sin\psi = k\sin W,$$

or,

$$k = \frac{\sin\psi}{\sin W}d. \tag{4.21}$$

As,

$$d\cos\psi = h + k\cos W,$$

solving with substitution of k in terms of ψ from (4.21),

$$h = d\left[\cos\psi - \frac{\sin\psi\cos W}{\sin W}\right],$$

or,

$$h = \frac{\sin(W-\psi)}{\sin W}d. \tag{4.22}$$

Substituting the values of h and k from equations (4.22) and (4.21) respectively into (4.19), following equation of r_0 is obtained;

$$r_0 = \frac{\sin(W-\psi)\sin\left(\frac{W+\theta}{2}\right) - \sin\psi\sin\left(\frac{W-\theta}{2}\right)}{2\sin\frac{\theta}{2}\sin\frac{W}{2}\sin W} d,$$

Simplifying the numerator,

$$\frac{1}{2}\left[\cos\left(\frac{W}{2}-\frac{\theta}{2}-\psi\right) - \cos\left(\frac{3}{2}W+\frac{\theta}{2}-\psi\right)\right] - \frac{1}{2}\left[\cos\left(-\frac{W}{2}+\frac{\theta}{2}+\psi\right) - \cos\left(\frac{W}{2}-\frac{\theta}{2}+\psi\right)\right],$$

$$= \frac{1}{2}\left[\cos\left(\frac{W}{2}-\frac{\theta}{2}+\psi\right) - \cos\left(\frac{3}{2}W+\frac{\theta}{2}-\psi\right)\right],$$

$$= -\frac{2}{2}\sin W\sin\left(-\frac{W+\theta}{2}+\psi\right),$$

$$= \sin W\sin\left(\frac{W+\theta}{2}-\psi\right).$$

Hence,

$$r_0 = \frac{\sin W\sin\left(\frac{W+\theta}{2}-\psi\right)}{2\sin\frac{\theta}{2}\sin\frac{W}{2}\sin W} d,$$

$$= \frac{\sin\left(\frac{W+\theta}{2}-\psi\right)}{2\sin\frac{\theta}{2}\sin\frac{W}{2}} d. \tag{4.23}$$

Similarly, r_1 is obtained from equation (4.20) as

$$r_1 = \frac{\sin\left(\psi - \frac{\theta}{2}\right)}{2\sin\left(\frac{W-\theta}{2}\right)\sin\frac{W}{2}}d. \tag{4.24}$$

For $r_0 > 0$, consider equation (4.23). For,

$$\sin\left(\frac{W+\theta}{2} - \psi\right) > 0,$$
$$\frac{W+\theta}{2} - \psi > 0,$$

or,

$$\theta > 2\psi - W.$$

Similarly for $r_1 > 0$ in equation (4.24),

$$\sin\left(\psi - \frac{\theta}{2}\right) > 0,$$

or,

$$\psi - \frac{\theta}{2} > 0,$$

Hence,

$$\theta < 2\psi.$$

Thus, θ has a range of values within which the biarc may be defined. Therefore, in order to have both r_0 and r_1 positive, $2\psi - W < \theta < 2\psi$. For convenience, the mid-point of the range is considered for the value of θ i.e.,

$$\theta = \frac{4\psi - W}{2}, \tag{4.25}$$

which corresponds to minimizing the difference in curvature of the two arcs of the biarc [16].

The biarc fitting scheme is also valid for the case where the value of r_1 is greater than r_0 i.e. $r_1 > r_0$. Figure 4.2 illustrates the biarc when $r_1 > r_0$. Given the scenario,

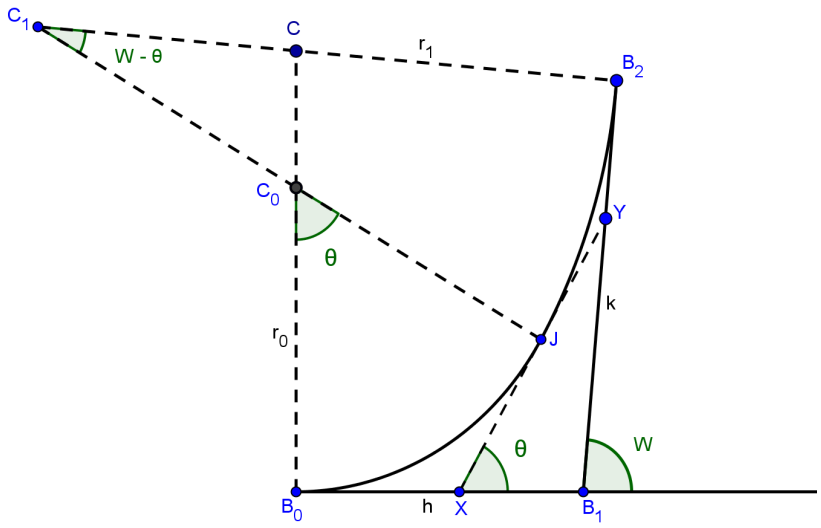

Figure 4.2: Biarc when $r_1 > r_0$.

$\overline{B}_2$ would be expressed in terms of $\overline{B}_0$ as:

$$\overline{B}_2 = \overline{B}_0 + r_0\overline{N}_0 + (r_1 - r_0)\overline{N} - r_1\overline{N}_2,$$

$$\text{or, } \overline{B}_2 - \overline{B}_0 = r_0\overline{N}_0 + (r_1 - r_0)\overline{N} - r_1\overline{N}_2$$

which is the same as equation (4.11) for $r_1 < r_0$. The rest of the calculations are the same and yield exactly the same equations for r_0 and r_1. Thus, the scheme remains unaffected by the relative sizes of the radii.

The mathematical equations derived originally in this work (4.19) and (4.20) will be used to calculate r_0 and r_1. Using these equations, biarc can be constructed

when the input parameters are h, k, W and θ.

4.2 Implementation Algorithm

1. Given the G^1 Hermite data consisting of the input points $\overline{B}_0$ and $\overline{B}_2$ with the respective tangents $\overline{T}_0$ and $\overline{T}_2$ at those points.

2. The point of intersection of the straight lines parallel to $\overline{T}_0$ and $\overline{T}_2$ through $\overline{B}_0$ and $\overline{B}_2$ is at the point $\overline{B}_1$. The angle from tangent $\overline{T}_0$ to $\overline{T}_2$ is defined as W. The angle W will have be restricted between $0° < W < 180°$ such that, an intersection point for the two tangent directions at $\overline{B}_1$ is always obtained and the biarc is C-shaped.

3. The length of the vectors, are measured as follows

$$|\overline{B_0B_1}| = h = |\overline{B}_1 - \overline{B}_0|;$$

$$|\overline{B_1B_2}| = k = |\overline{B}_2 - \overline{B}_1|;$$

$$|\overline{B_0B_2}| = d = |\overline{B}_2 - \overline{B}_0|;$$

4. The angle W is

$$W = \sin^{-1}(\overline{T}_0 \times \overline{T}_2),$$

and angle ψ is

$$\psi = \sin^{-1}\left(\frac{\overline{T}_0 \times (\overline{B}_2 - \overline{B}_0)}{|\overline{B}_2 - \overline{B}_0|}\right).$$

5. The value of parameter θ is chosen depending on the value of W and ψ. Good biarcs are obtained as long as $2\psi - W < \theta < 2\psi$.

6. The radii r_0 and r_1 of the biarc are obtained from the equations (4.19) and (4.20).

7. The initial and final points and respective tangents together with the radii now determine the centres $\overline{C}_0$ and $\overline{C}_2$ of the two arcs of the biarc.

8. The biarc is constructed. The first circular arc is drawn centered at $\overline{C}_0$ having radius r_0 and angle θ. Similarly, the second arc centered $\overline{C}_1$ with radius r_1 and angle $W - \theta$ is drawn. As the method forces the tangents to the arcs to match at the joint, a tangent continuous biarc is obtained.

Chapter 5

NURBS Curve Fitting

In this section, the NURBS curve fitting scheme used for shape preserving interpolation is discussed in detail. As discussed in section 1.3.2, the quadratic NURBS curve have been used to generate circular arcs. The user does not need to know whether the data points belong to a circular arc or not.

Consider two-point G^1 Hermite data (section 2.6). The proposed scheme is explained in more detail below.

5.1 Constructing the quadratic NURBS Curve

In order to define the quadratic NURBS curve three control points with corresponding weights are required. For the first quadratic NURBS curve $\overline{C}_L(u)$, control points $\overline{P}_0 = \overline{A}$, $\overline{P}_L$ and $\overline{P}_2 = \overline{B}$ with corresponding weights $w_0 = 1$, w_L and $w_2 = 1$ respectively, are declared. Similarly, for $\overline{C}_R(u)$, $\overline{P}_0 = \overline{A}$, $\overline{P}_R$ and $\overline{P}_2 = \overline{B}$ with weights $w_0 = 1$, w_R and $w_2 = 1$ respectively. The weights at the first and the third control

points have been chosen as unity for simplicity and is a standard practice in Piegl [24]. The control points $\overline{P}_L$, $\overline{P}_R$ with weights w_L and w_R are to be determined.

In order to construct the quadratic NURBS curve P_L needs to be determined. For NURBS curve to be a circular arc, P_L will merge with the point of intersection of the tangent directions at the end points such that, $|\overline{P_0P_L}| = |\overline{P_LP_2}|$. Piegl and Tiller [21] established that, for the quadratic to be a circular arc, the weight at the control point $\overline{P}_L$ (refer Figure 5.1) should be $w_L = \cos\alpha$, where α is the angle from the chord $\overline{P_0P_2}$ to $\overline{P_0P_L}$ to the tangent direction $\overline{T}_0$.

5.1.1 Equation of the first quadratic $\overline{C}_L(u)$

Consider Figure 5.1. Given the points $\overline{P_0}$, $\overline{P_2}$. Let length $d = |\overline{P_0P_2}|$, then the perpendicular distance from the point $\overline{P_L}$ to the vector $\overline{P_0P_2}$ is $\frac{1}{2}d\tan\alpha$ i.e.

$$\overline{P}_L = \overline{P}_0 + (\frac{1}{2}d, \frac{1}{2}d\tan\alpha). \tag{5.1}$$

Hence, the equation of the first quadratic NURBS curve is expressed as;

$$\overline{C}_L(u) = \frac{(1-u)^2\overline{P}_0 + 2u(1-u)\overline{P}_L\cos\alpha + u^2\overline{P}_2}{(1-u)^2 + 2u(1-u)\cos\alpha + u^2} \tag{5.2}$$

5.1.2 Equation of the second quadratic $\overline{C}_R(u)$

Similarly, for the second quadratic $\overline{C}_R$ (refer Figure 5.2) to be constructed, the weight w_R at the control point $\overline{P}_R$ is $w_R = \cos\beta$ where, β is the angle from $\overline{P_0P_2}$ to $\overline{T}_B$. So,

$$\overline{P}_R = \overline{P}_2 - (\frac{1}{2}d, \frac{1}{2}d\tan\beta) \tag{5.3}$$

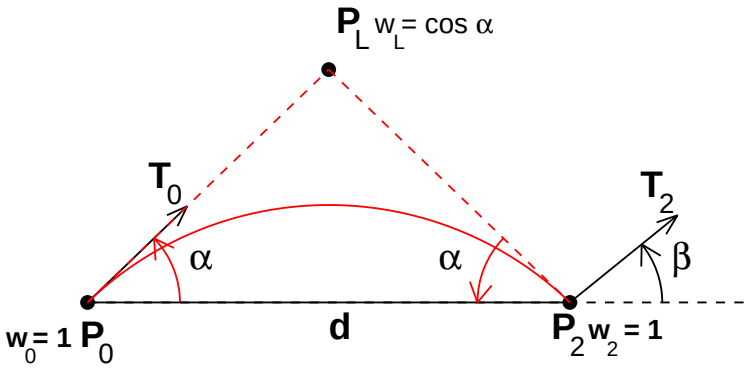

Figure 5.1: Data set with $\overline{C}_L(u)$

Therefore, the second quadratic is;

$$\overline{C}_R(u) = \frac{(1-u)^2\overline{P}_2 + 2u(1-u)\overline{P}_R\cos\beta + u^2\overline{P}_0}{(1-u)^2 + 2u(1-u)\cos\beta + u^2}. \tag{5.4}$$

A cubic NURBS curve is obtained, after the two quadratics given by equations (5.2) and (5.4) are blended together linearly. If the two quadratics belong to the same circular arc, a degree-raised quadratic representing a circular arc is obtained (refer to section 6.2.1) or otherwise, a more general cubic NURBS curve is obtained. The various examples of the curve fitting scheme are shown in section 6.2. The next section

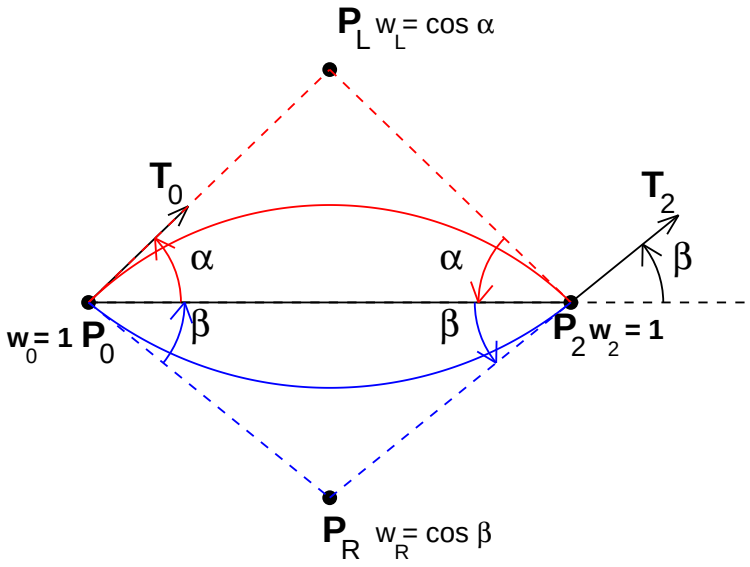

Figure 5.2: Data set with $\overline{C}_L(u)$ and $\overline{C}_R(u)$.

illustrates in detail the process of linear blending.

5.2 Linear Blending

Linear blending of two quadratic NURBS curve may be expressed as a linear combination of their control points and the weights on those control points. Consider Figure 5.3. The first quadratic NURBS curve (say, the left curve) is defined by control points $\overline{B}_{L,0}$, $\overline{B}_{L,1}$ and $\overline{B}_{L,2}$ with the weights $w_{L,0} = 1$, $w_{L,1}$ and $w_{L,2} = 1$ respectively.

The quadratic NURBS curve on the right is given by the control points $\overline{B}_{R,0}$, $\overline{B}_{R,1}$ and $\overline{B}_{R,2}$ and the weights $w_{R,0} = 1$, $w_{R,1}$, $w_{R,2} = 1$ respectively. In order to linearly blend the two curves to form a cubic NURBS curve, linear blending functions have to be determined. The linear blend $\overline{Q}(t)$ of the two quadratics is,

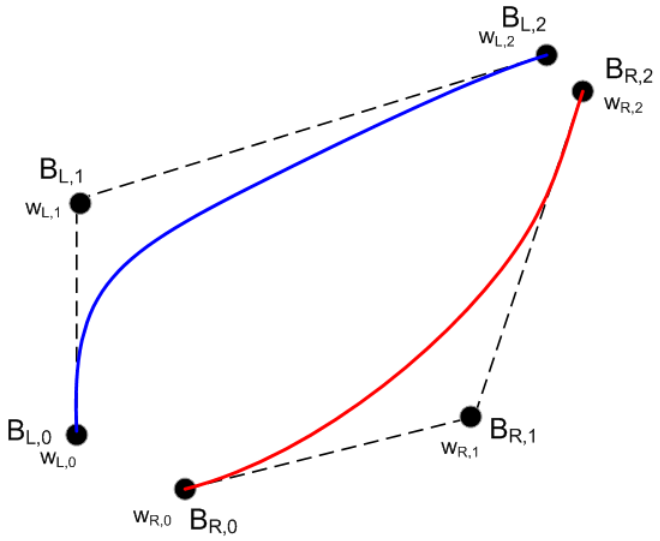

Figure 5.3: A set of quadratic NURBS curves.

$$\overline{Q}(t) = \frac{\overline{N}}{D} = \frac{(1-t)^2\overline{B}_0(t)w_0(t) + 2t(1-t)\overline{B}_1(t)w_1(t) + t^2\overline{B}_2(t)w_2(t)}{(1-t)^2w_0(t) + 2t(1-t)w_1(t) + t^2w_2(t)}. \tag{5.5}$$

with control points $\overline{B}_0$, $\overline{B}_1$ and $\overline{B}_2$ defined in terms of linear blending functions of the control points and weights for the left and the right curve.

$$\begin{aligned}
\overline{B}_0(t) &= (1-t)\overline{B}_{L,0} + t\overline{B}_{R,0}, \\
\overline{B}_1(t) &= \frac{(1-t)\overline{B}_{L,1}w_{L,1} + t\overline{B}_{R,1}w_{R,1}}{(1-t)w_{L,1} + tw_{R,1}}, \\
\overline{B}_2(t) &= (1-t)\overline{B}_{L,2} + t\overline{B}_{R,2}.
\end{aligned} \tag{5.6}$$

The weights w_0, w_1 and w_2 are the linear combination of the weights of the original quadratic NURBS curves:

$$\begin{aligned} w_0(t) &= (1-t) + t = 1, \\ w_1(t) &= (1-t)w_{L,1} + tw_{R,1}, \\ w_2(t) &= (1-t) + t = 1, \end{aligned} \tag{5.7}$$

with $0 \leq t \leq 1$. Replacing the values from equations (5.6) and (5.7) into equation (5.5) leads to the following expression for the numerator and denominator:

$$\begin{aligned} \overline{N} &= (1-t)^3\overline{B}_{L,0} + 3t(1-t)^2\left(\frac{\overline{B}_{R,0} + 2\overline{B}_{L,1}w_{L,1}}{3}\right) \\ &\quad + 3t^2(1-t)\left(\frac{2\overline{B}_{R,1}w_{R,1} + \overline{B}_{L,2}}{3}\right) + t^3\overline{B}_{R,2}, \end{aligned} \tag{5.8}$$

$$\begin{aligned} D &= (1-t)^3 + 3t(1-t)^2\left(\frac{1+2w_{L,1}}{3}\right) + 3t^2(1-t)\left(\frac{2w_{R,1}+1}{3}\right) \\ &\quad + t^3. \end{aligned} \tag{5.9}$$

Hence,

$$\frac{\overline{N}}{D} = \overline{Q}(t) = \frac{(1-t)^3\overline{B^*}_0 w_0^* + 3t(1-t)^2\overline{B^*}_1 w_1^* + 3t^2(1-t)\overline{B^*}_2 w_2^* + t^3\overline{B^*}_3 w_3^*}{(1-t)^3 w_0^* + 3t(1-t)^2 w_1^* + 3t^2(1-t)w_2^* + t^3 w_3^*}. \tag{5.10}$$

For the present application, the control points and weights of the first quadratic are $\overline{P}_0$, $\overline{P}_L$, $\overline{P}_2$ and 1, $\cos\alpha$, 1; the control points and weights for the second quadratic are $\overline{P}_0$, $\overline{P}_R$, $\overline{P}_2$ and 1, $\cos\beta$, 1. The curve $\overline{Q}(t)$ can be rewritten in the form of equation (5.10) using equations (5.8) and (5.9) as

$$\overline{Q}(t) = \frac{(1-t)^3\overline{P}_0 + 3t(1-t)^2\left(\frac{2\overline{P}_L\cos\alpha + \overline{P}_R}{3}\right) + 3t^2(1-t)\left(\frac{2\overline{P}_R\cos\beta + \overline{P}_L}{3}\right) + t^3\overline{P}_2}{(1-t)^3 + 3t(1-t)^2\left[\frac{1+2\cos\alpha}{3}\right] + 3t^2(1-t)\left[\frac{2\cos\beta+1}{3}\right] + t^3}. \tag{5.11}$$

5.3 Implementation Algorithm

The following algorithm is based on the results of the previous section (5.2).

1. Input the points $\overline{A}$ and $\overline{B}$ from the user.

2. Input tangents $\overline{T}_A$ and $\overline{T}_B$ at the points $\overline{A}$ and $\overline{B}$.

3. Determine the angles α and β from the points and tangents.

4. Determine the value $d = |\overline{B} - \overline{A}|$.

5. Construct the first quadratic NURBS curve($\overline{C}_L(u)$). Assume that, $\overline{P}_0 = \overline{A}$, tangent $\overline{T}_A$, $\overline{P}_2 = \overline{B}$. Point $\overline{P}_L$ is obtained by equation (5.1). The quadratic NURBS curve is defined using equation (5.2).

6. Construct the second quadratic NURBS curve($\overline{C}_R(u)$) with $\overline{P}_0 = \overline{A}$, tangent $\overline{T}_B$, $\overline{P}_2 = \overline{B}$. Equation (5.3) defines the point $\overline{P}_R$ while the equation of the quadratic is defined by equation (5.4).

7. For blending the two quadratic NURBS curves $\overline{C}_L$ and $\overline{C}_R$, to yield the required cubic NURBS curve, equation (5.8) and (5.9) are used.

8. The cubic NURBS curve $\overline{Q}(t)$ is constructed using equation (5.11).

5.4 Negative weights

There are occasions during the construction of the NURBS curve when the curve is obtained outside the convex hull of the control polygon. Also known as the complementary arc, it is obtained due to the negative weights defined for the curve. As

a solution Piegl [24] suggests inserting a knot vector at half of the parametric value which would create a new control polygon that contains the arc in the convex hull. Similar observation of 'negative weights' is also mentioned in the recent work by Farin [33] where, subdivision of the curve is suggested. Piegl [24] mentions some of these restrictions in terms their ability to generate a semi-circle or an ellipse. Some of the examples with negative weights are shown in section 6.2.3.

Chapter 6

Examples

This chapter describes some of the examples of the two curve fitting approaches introduced in this work. The purpose is to observe the way the scheme generates output for different input values.

6.1 Biarc Fitting

For experimentation, the value of W was kept constant for a set of values of h and k, while the value of θ is varied. The effect of the value of the parameter θ on the final biarc is then observed. In the biarc fitting examples used below (Figure 6.1 and 6.2) red color has been used for the first circular arc and blue for the second arc of the biarc.

6.1.1 Example 1

The input parameters are as follows; point $P_0 = (1.0, 0, 0)$, $P_2 = (5.5, 2.6, 0)$ i.e. $h = k = 3$, and $W = 60°$. The value of angle ψ is found to be $30°$. For $W = 60°$, $\psi = 30°$, the range of θ should be within $2\psi - W < \theta < 2\psi$ i.e. $0° < \theta < 60°$, which is consistent with the inequality. The affect on the shape of the biarc can be observed in Figure 6.1. The radii and centres in the figure have not been marked as their values are too large to appear with the biarc. In order to redraw them, size of the biarcs will have to be decreased considerably which will render the figure unviewable.

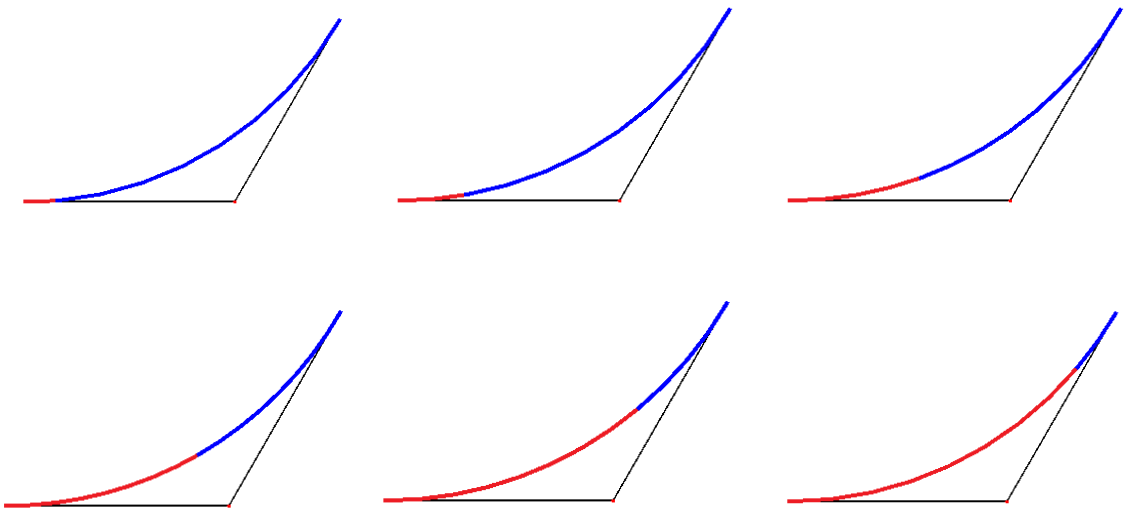

Figure 6.1: Biarc when $W = 60°$, $\psi = 30°$ for $\theta = 5°, 10°, 20°, 30°, 40°$ and $50°$.

6.1.2 Example 2

The input parameters are as follows; point $P_0 = (1.0, 0, 0)$, $P_2 = (4.3, 2.5, 0)$ i.e. $h = 3$, $k = 1.5$ and $W = 80°$. The angle ψ is found to be $25°$ through the observations.

For $W = 80°$, $\psi = 25°$, the range of θ should be within $2\psi - W < \theta < 2\psi$ i.e. $-30° < \theta < 50°$ to be consistent with the inequality. As $0° < \theta$ hence, $0° < \theta < 50°$ which is observed in the figure below. Figure 6.2 illustrates the change in the shape of the biarc as the parameter value θ is changed.

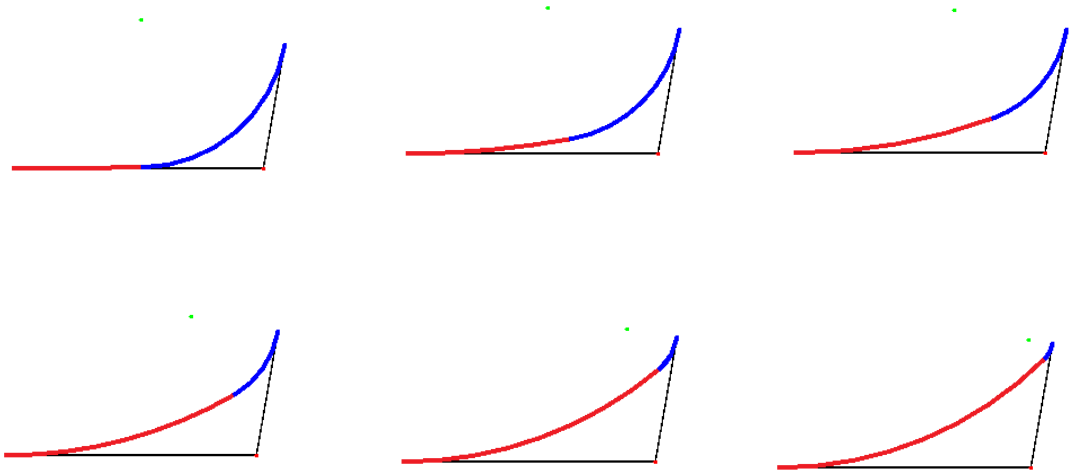

Figure 6.2: Biarc when $W = 80°$, $\psi = 25°$ for $\theta = 1°, 10°, 20°, 30°, 40°$ and $45°$.

6.2 NURBS Curve Fitting

For the examples in this section, the first quadratic is drawn in blue, the second qudratic in red while the final cubic curve is represented in green.

6.2.1 Cases with NURBS curve as circular arcs

It is known from elementary geometry that circular arcs match the G^1 Hermite data when the angle that the tangents make with the line segment connecting the two points in G^1 Hermite data are equal. Hence, if α and β are such angles then,

$$\alpha = -\beta \text{ or, vice-versa}$$

Figures 6.3 and 6.4 illustrate the case when the points originate form a circular arc. Since, the two quadratic NURBS correspond to the same circular arc, linear blending of the two produces the arc as degree-raised quadratic NURBS.

6.2.2 Cases with NURBS curve in general

This case illustrates the examples when the given data points do not belong to a circular arc and none of the weights are negative. Figure 6.5 shows the curve formed with $\alpha = 43°$, $\beta = 38°$ and $d = 2.60$. As the final output curve depends on the two tangent angles, an S- shaped curve is obtained satisfying the initial and final tangent directions. In Figure 6.6, the cubic NURBS curve is obtained for α close to $90°$ and $\beta = 22°$.

6.2.3 Cases with negative weights

The cases with the negative weights are shown in Figure 6.7 with $\alpha = 35°$, $\beta = 114°$. When the value of β is increased, the shape of the final curve changes and is illustrated in Figure 6.7. Figure 6.8 and 6.9 show two more such cases.

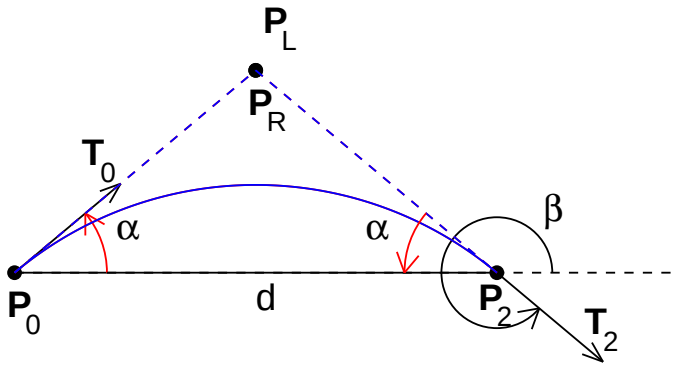

Figure 6.3: Final NURBS curve when $\alpha = 40°$, $\beta = 320°$

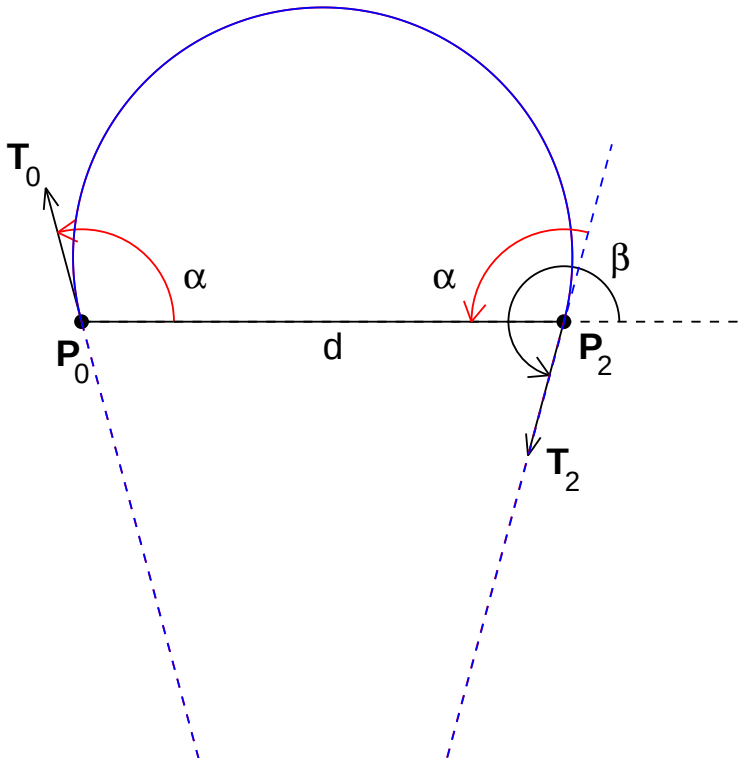

Figure 6.4: Final NURBS curve when $\alpha = 105°$, $\beta = 255°$

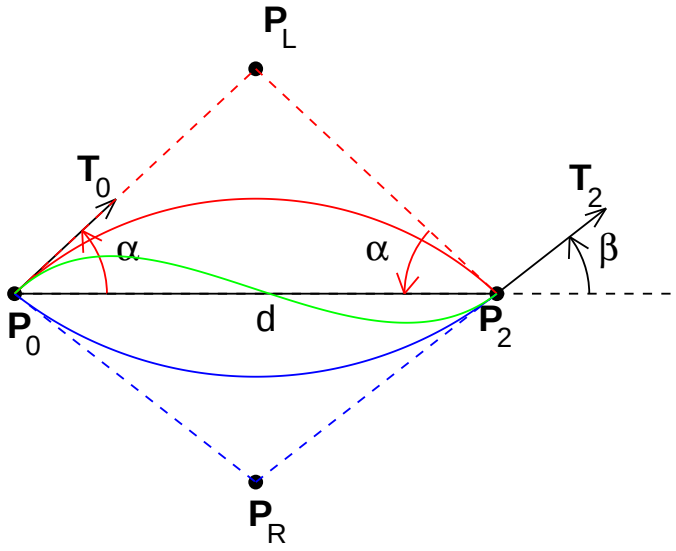

Figure 6.5: Final NURBS curve when $\alpha = 43°$, $\beta = 38°$

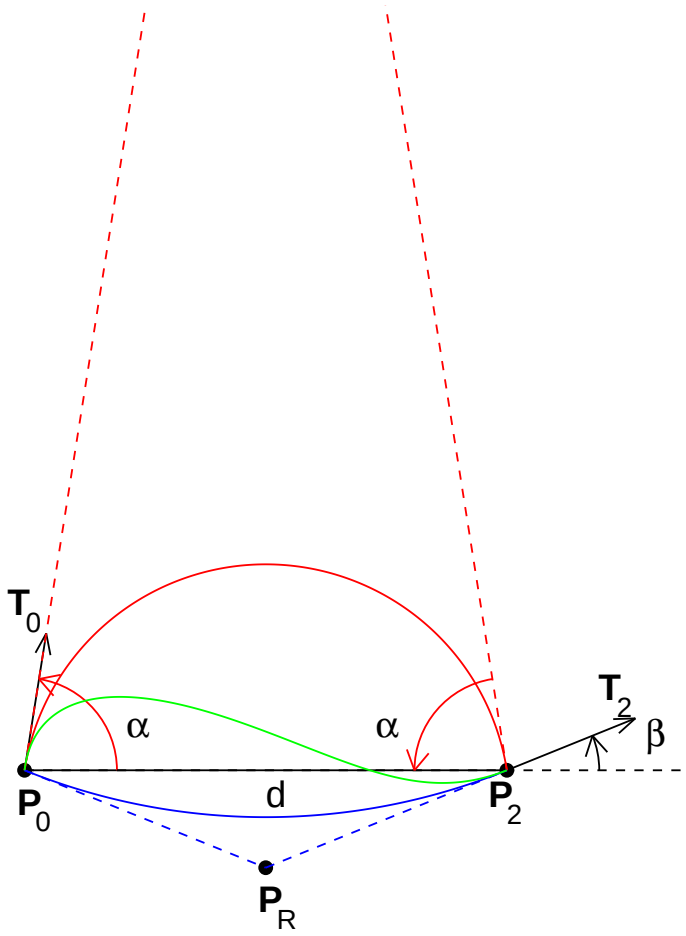

Figure 6.6: Final NURBS curve when $\alpha = 81°$, $\beta = 22°$

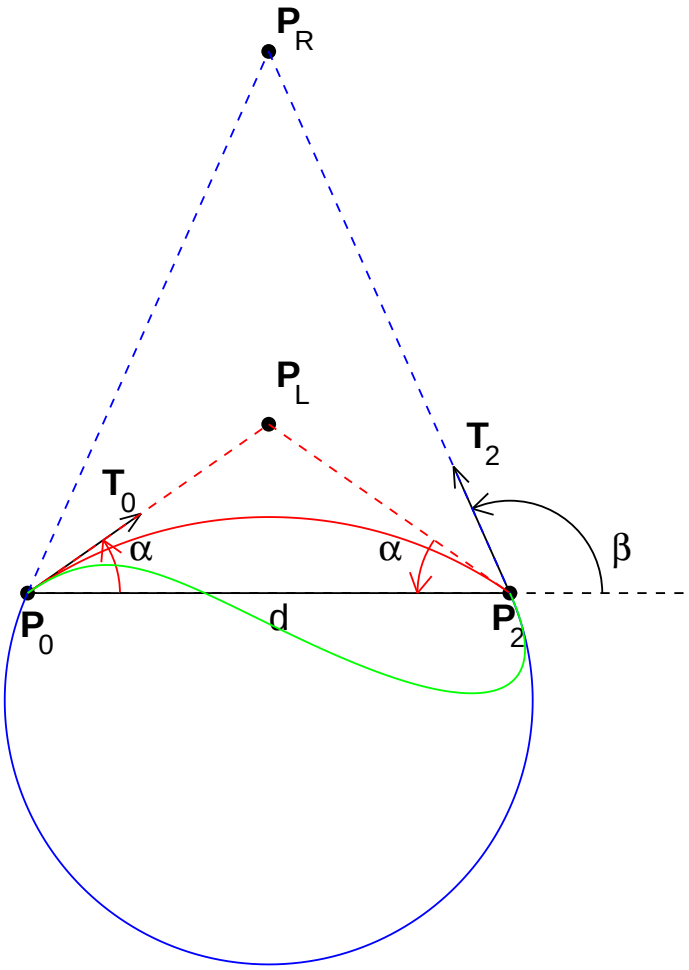

Figure 6.7: Final NURBS curve when $\alpha = 35°$, $\beta = 114°$

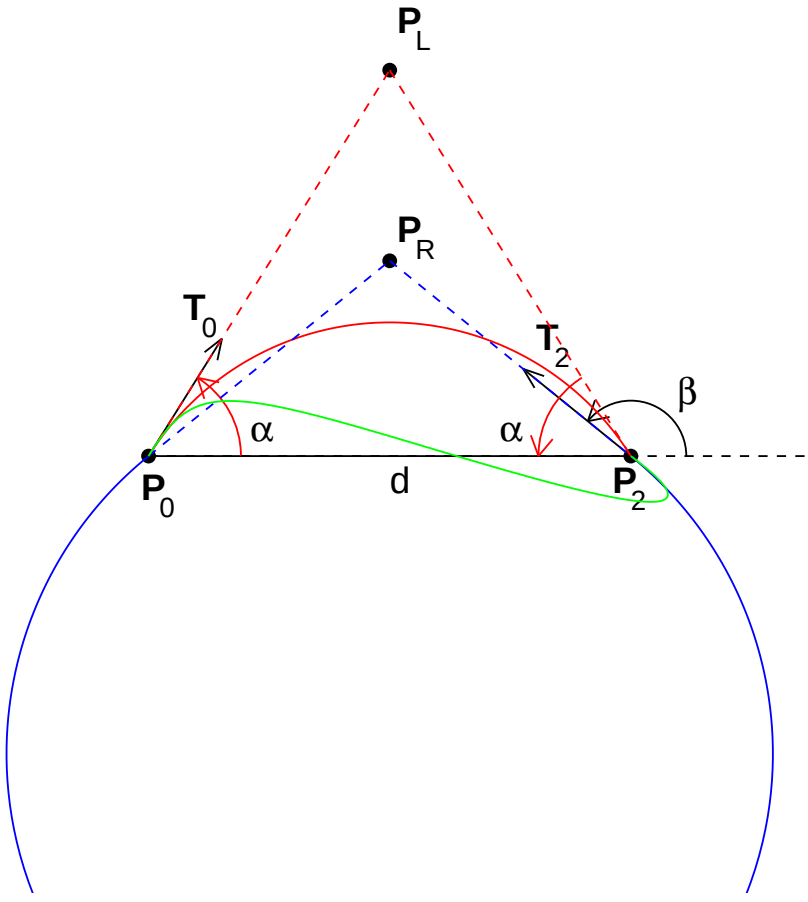

Figure 6.8: Final NURBS curve when $\alpha = 58°$, $\beta = 141°$

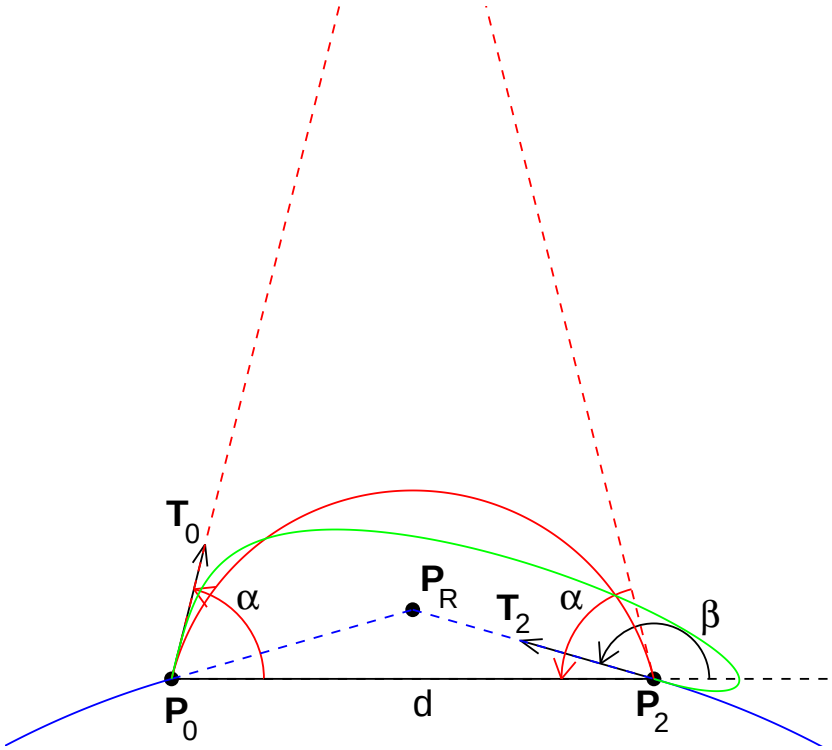

Figure 6.9: Final NURBS curve when $\alpha = 76°$, $\beta = 164°$

Chapter 7

Approximating NURBS Curve with Biarcs

This chapter describes an algorithm to approximate the cubic NURBS curve with biarcs. The idea is to successively subdivide the curve into several pieces until each piece of the curve can be approximated by a biarc within a given tolerance limit(ϵ). Consider Figure 7.1. For a given pair of points $\overline{V}_i$, $\overline{V}_{i+1}$ and corresponding tangents $\overline{H}_i$, $\overline{H}_{i+1}$ at them, a cubic NURBS curve (denoted in green) $\overline{C}_i(t)$ is first constructed and then approximated by the biarc $\overline{B}_i$ (denoted in red and blue). A biarc consists of two circular arcs. The calculation of the distance from the NURBS curve to the first arc will be described in detail. The distance from the NURBS curve to the other arc can be treated in a similar manner. The centre of the first arc of the biarc is assumed to be at $\overline{M}_i$ and the other at $\overline{M}_{i+1}$. A line is drawn from the centre $\overline{M}_i$ in the radial direction at an angle (ϕ) from the x-axis, intersecting the biarc $\overline{B}_i$ at $\overline{R}_b$ and the curve $\overline{C}_i$ at $\overline{R}_c$. Let, the distance $|\overline{M_iR_c}| = l$ and the distance $|\overline{M_iR_b}| = r$.

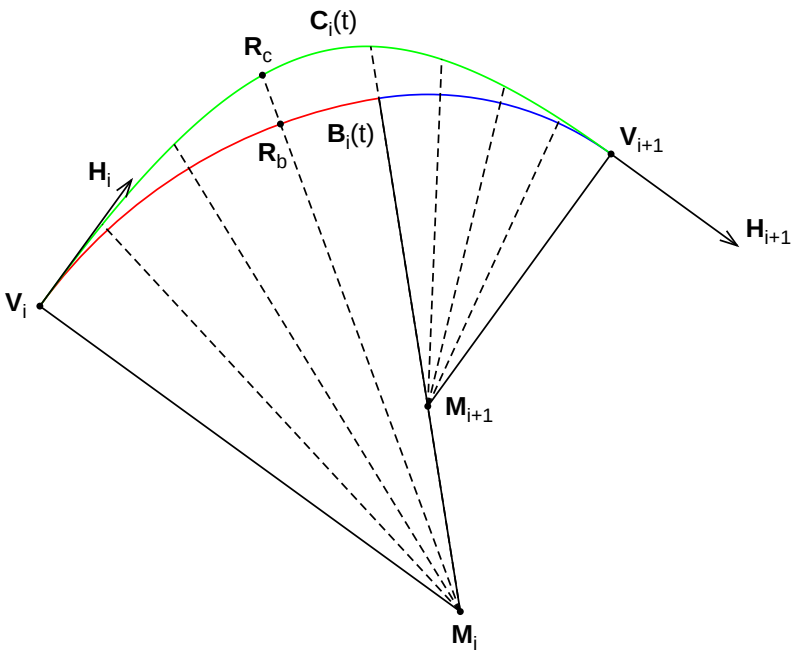

Figure 7.1: Distance between a NURBS curve and a biarc.

The distance d between the point $\bar{R}_c$ on the NURBS curve and the point $\overline{R}_b$ on the biarc is expressed as;

$$d = |l - r|.$$

7.1 Calculating points at angular intervals

In order to find the value of d, the point $\overline{R}_c$ on the curve has to be determined. Given the center $\overline{M}_i$, point the $\overline{R}_c$ is expressed in terms of polar coordinates as;

$$\overline{R}_c = \overline{M}_i + l(\cos\phi, \sin\phi).$$

Since, $\overline{R}_c$ is also on the cubic rational curve $\overline{C}_i$, it satisfies the equation for a NURBS curve at a parametric value (say, t_R). Hence,

$$\overline{M}_i + l(\cos\phi, \sin\phi) = \frac{(1-t)^3\overline{B}_0 w_0 + 3t(1-t)^2\overline{B}_1 w_1 + 3t^2(1-t)\overline{B}_2 w_2 + t^3\overline{B}_3 w_3}{(1-t)^3 w_0 + 3t(1-t)^2 w_1 + 3t^2(1-t) w_2 + t^3 w_3}.$$

Expressing the x and y components separately,

$$M_{i,x} + l(\cos\phi) = \frac{(1-t)^3 B_{0,x} w_0 + 3t(1-t)^2 B_{1,x} w_1 + 3t^2(1-t) B_{2,x} w_2 + t^3 B_{3,x} w_3}{(1-t)^3 w_0 + 3t(1-t)^2 w_1 + 3t^2(1-t) w_2 + t^3 w_3},$$

$$\text{and, } M_{i,y} + l(\sin\phi) = \frac{(1-t)^3 B_{0,y} w_0 + 3t(1-t)^2 B_{1,y} w_1 + 3t^2(1-t) B_{2,y} w_2 + t^3 B_{3,y} w_3}{(1-t)^3 w_0 + 3t(1-t)^2 w_1 + 3t^2(1-t) w_2 + t^3 w_3}.$$

The above set of equations is reduced to a single equation in terms of one unknown, i.e. the parameter t, by eliminating the second unknown l. The process of elimination yields the following equation.

$$(1-t)^3 A_0 + 3t(1-t)^2 A_1 + 3t^2(1-t) A_2 + t^3 A_3 w_3 = 0$$

where,

$$A_0 = w_0(B_{0,x}\sin\phi - B_{0,y}\cos\phi - M_{i,x}\sin\phi + M_{i,y}\cos\phi),$$

$$A_1 = w_1(B_{1,x}\sin\phi - B_{1,y}\cos\phi - M_{i,x}\sin\phi + M_{i,y}\cos\phi),$$

$$A_2 = w_2(B_{2,x}\sin\phi - B_{2,y}\cos\phi - M_{i,x}\sin\phi + M_{i,y}\cos\phi),$$

$$A_3 = w_3(B_{3,x}\sin\phi - B_{3,y}\cos\phi - M_{i,x}\sin\phi + M_{i,y}\cos\phi).$$

Given the control points $\overline{B}_0$, $\overline{B}_1$, $\overline{B}_2$ and $\overline{B}_3$, and the weights w_0, w_1, w_2 and w_3, the above equation is solved using the bisection method to obtain the parametric value t_R. This is the value which defines the point $\overline{R}_c$ on the curve $\overline{C}_i$. Thus, putting the value of t_R in the cubic equation would yield $\overline{R}_c$;

$$\overline{R}_c = \frac{(1-t_R)^3\overline{B}_0 w_0 + 3t_R(1-t_R)^2\overline{B}_1 w_1 + 3t_R^2(1-t_R)\overline{B}_2 w_2 + t_R^3\overline{B}_3 w_3}{(1-t_R)^3 w_0 + 3t_R(1-t_R)^2 w_1 + 3t_R^2(1-t_R) w_2 + t_R^3 w_3}. \qquad (7.1)$$

The problem of finding the distance between a curve and circular arc is mathematically complicated. In order to find a simple solution, it seems convenient to calculate the distance between the biarc and the curve at regular angular intervals. The number of intervals, n, could be increased for better accuracy of the results. In this thesis, for the example in the previous chapter, $n = 50$ was used. Although this method is not the most accurate, it works well as a very high precision is not required for these experiments.

7.2 Determining the distance between the curve and the biarc

The value of l is obtained from $\overline{R}_c$, given by equation (7.1), as follows:

$$\overline{R}_c = \overline{M}_i + l(\cos\phi, \sin\phi),$$

$$\therefore l = \left(\left[\frac{\bar{R}_{c,x} - \overline{M}_{i,x}}{\cos\phi} \right]^2 + \left[\frac{\overline{R}_{c,y} - \overline{M}_{i,y}}{\sin\phi} \right]^2 \right)^{\frac{1}{2}}.$$

As the value of the radius of the biarc r is already known, d is determined from $d = |l - r|$. Several values of $\bar{R}_b$ and $\overline{R}_c$ are taken and the distance d is calculated.

In this example (Figure 7.1), several values of $\overline{R}_b$, $\overline{R}_c$ for different values of ϕ are obtained between the points $\overline{V}_i$ and $\bar{V}_{i+1}$, for $n = 0$ (at $\overline{V}_i$), 5 and 6 (at $\overline{V}_{i+1}$). The first point on the biarc $\overline{R}_b$ would be at $\overline{V}_i$ and, value of ϕ is subsequently decreased (in a clockwise manner) n times until;

1. $d > \epsilon$, where ϵ is defined in Chapter 2 section 2.12; then the search for next $\overline{R}_c$ is terminated and the biarc is considered to be outside the tolerance limit of the

curve $C_i(u)$.

2. $\overline{R}_c = \overline{R}_b = \overline{V}_{i+1}$, in which case, the end of the biarc as well as the curve has been reached and the search is terminated. The biarc is considered to be within the tolerance limit of the curve.

7.3 Parametric subdivision of cubic NURBS curve

Parametric subdivision for the cubic curve is performed in the following manner; Consider Figure 7.2. Given the equation of a cubic NURBS curve,

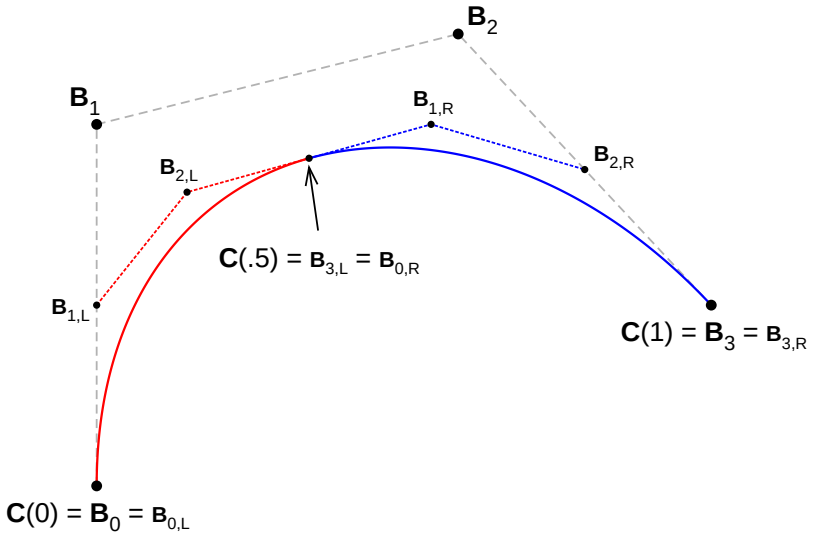

Figure 7.2: Parametric subdivision of cubic NURBS.

$$\overline{C}(t) = \frac{(1-t)^3\overline{B}_0 w_0 + 3t(1-t)^2\overline{B}_1 w_1 + 3t^2(1-t)\overline{B}_2 w_2 + t^3\overline{B}_3 w_3}{(1-t)^3 w_0 + 3t(1-t)^2 w_1 + 3t^2(1-t) w_2 + t^3 w_3} \text{ for } 0 \leq t \leq 1$$

The left half of the curve $\overline{C}_L(t)$ would be given as;

$$\overline{C}_L(t) = \frac{(1-t)^3\overline{B}_0 w_0 + 3t(1-t)^2\overline{B}_1 w_1 + 3t^2(1-t)\overline{B}_2 w_2 + t^3\overline{B}_3 w_3}{(1-t)^3 w_0 + 3t(1-t)^2 w_1 + 3t^2(1-t) w_2 + t^3 w_3} \text{ for } 0 \leq t \leq \frac{1}{2}$$

As $0 \leq t \leq \frac{1}{2}$ or, $0 \leq 2t \leq 1$, let $v = 2t$, then $0 \leq v \leq 1$ and $t = \frac{1}{2}v$. Hence,

$$\overline{C}_L(v) = \frac{\left(1-\frac{1}{2}v\right)^3 \overline{B}_0 w_0 + 3\left(\frac{v}{2}\right)\left(1-\frac{1}{2}v\right)^2 \overline{B}_1 w_1 + 3\left(\frac{v}{2}\right)^2\left(1-\frac{1}{2}v\right)\overline{B}_2 w_2 + \left(\frac{1}{2}v\right)^3 \overline{B}_3 w_3}{\left(1-\frac{1}{2}v\right)^3 w_0 + 3\left(\frac{v}{2}\right)\left(1-\frac{1}{2}v\right)^2 w_1 + 3\left(\frac{v}{2}\right)^2\left(1-\frac{1}{2}v\right) w_2 + \left(\frac{1}{2}v\right)^3 w_3}$$

$$= \frac{\left(1-\frac{3}{2}v+\frac{3}{4}v^2-\frac{1}{8}v^3\right)\overline{B}_0 w_0 + \left(\frac{3}{2}v-\frac{3}{2}v^2+\frac{3}{8}v^3\right)\overline{B}_1 w_1 + \left(\frac{3}{4}v^2-\frac{3}{8}v^3\right)\overline{B}_2 w_2 + \frac{v^3}{8}\overline{B}_3 w_3}{\left(1-\frac{3}{2}v+\frac{3}{4}v^2-\frac{1}{8}v^3\right) w_0 + \left(\frac{3}{2}v-\frac{3}{2}v^2+\frac{3}{8}v^3\right) w_1 + \left(\frac{3}{4}v^2-\frac{3}{8}v^3\right) w_2 + \frac{v^3}{8} w_3}$$

For the purpose of re-grouping, considering the numerator of the above equation and rewriting the parametric terms, the numerator is expanded as

$$\left(1 - 3v + 3v^2 - v^3 + \frac{3}{2}v - 3v^2 + \frac{3}{2}v^3 + \frac{3}{4}v^2 - \frac{3}{4}v^3 + \frac{1}{8}v^3\right)\overline{B}_0 w_0$$
$$+ \left(\frac{3}{2}v - 3v^2 + \frac{3}{2}v^3 + \frac{3}{2}v^2 - \frac{3}{2}v^3 + \frac{3}{8}v^3\right)\overline{B}_1 w_1 + \left(\frac{3}{4}v^2 - \frac{3}{4}v^3 + \frac{3}{8}v^3\right)\overline{B}_2 w_2 + \left(\frac{1}{8}v^3\right)\overline{B}_3 w_3.$$

$$= \left((1-v)^3 + \frac{3}{2}v(1-v)^2 + \frac{3}{4}v^2(1-v) + \frac{1}{8}v^3\right)\overline{B}_0 w_0 + \left(\frac{3}{2}v(1-v)^2 + \frac{3}{2}v^2(1-v) + \frac{3}{8}v^3\right)\overline{B}_1 w_1$$
$$+ \left(\frac{3}{4}v^2(1-v) + \frac{3}{8}v^3\right)\overline{B}_2 w_2 + \left(\frac{1}{8}v^3\right)\overline{B}_3 w_3.$$

Similarly for the denominator is expressed as

$$\left((1-v)^3 + \frac{3}{2}v(1-v)^2 + \frac{3}{4}v^2(1-v) + \frac{1}{8}v^3\right) w_0 + \left(\frac{3}{2}v(1-v)^2 + \frac{3}{2}v^2(1-v) + \frac{3}{8}v^3\right) w_1$$
$$+ \left(\frac{3}{4}v^2(1-v) + \frac{3}{8}v^3\right) w_2 + \left(\frac{1}{8}v^3\right) w_3.$$

Hence the equation for $\overline{C}_L(v)$ can be rewritten as,

$$\bar{C}_L(v) = \frac{(1-v)^3\overline{B}_0 w_0 + 3v(1-v)^2\left(\frac{1}{2}\overline{B}_0 w_0 + \frac{1}{2}\overline{B}_1 w_1\right) + 3v^2(1-v)\left(\frac{1}{4}\overline{B}_0 w_0 + \frac{1}{2}\overline{B}_1 w_1 + \frac{1}{4}\overline{B}_2 w_2\right) + v^3\left(\frac{1}{8}\overline{B}_0 w_0 + \frac{3}{8}\overline{B}_1 w_1 + \frac{3}{8}\overline{B}_2 w_2 + \frac{1}{8}\overline{B}_3 w_3\right)}{(1-v)^3 w_0 + 3v(1-v)^2\left(\frac{1}{2}w_0 + \frac{1}{2}w_1\right) + 3v^2(1-v)\left(\frac{1}{4}w_0 + \frac{1}{2}w_1 + \frac{1}{4}w_2\right) + v^3\left(\frac{1}{8}w_0 + \frac{3}{8}w_1 + \frac{3}{8}w_2 + \frac{1}{8}w_3\right)}$$

The new weights $w_{L,0}$, $i = 0, \cdots, 3$ for $\overline{C}_L(v)$ in terms of w_0, w_1, w_2, w_3 are as follows

$$\begin{aligned}
w_{L,0} &= w_0, \\
w_{L,1} &= \frac{1}{2}w_0 + \frac{1}{2}w_1, \\
w_{L,2} &= \frac{1}{4}w_0 + \frac{1}{2}w_1 + \frac{1}{4}w_2, \\
w_{L,3} &= \frac{1}{8}w_0 + \frac{3}{8}w_1 + \frac{3}{8}w_2 + \frac{1}{8}w_3.
\end{aligned}$$

The new control points are, as in [23]:

$$\begin{aligned}
\overline{B}_{L,0} &= (\overline{B}_0 w_0)/w_{L,0} = \overline{B}_0, \\
\overline{B}_{L,1} &= \left(\frac{1}{2}\overline{B}_0 w_0 + \frac{1}{2}\overline{B}_1 w_1\right)/w_{L,1}, \\
\overline{B}_{L,2} &= \left(\frac{1}{4}\overline{B}_0 w_0 + \frac{1}{2}\overline{B}_1 w_1 + \frac{1}{4}\overline{B}_2 w_2\right)/w_{L,2}, \\
\overline{B}_{L,3} &= \left(\frac{1}{8}\overline{B}_0 w_0 + \frac{3}{8}\overline{B}_1 w_1 + \frac{3}{8}\overline{B}_2 w_2 + \frac{1}{8}\overline{B}_3 w_3\right)/w_{L,3}.
\end{aligned}$$

Similar steps repeated on the equation of $\overline{C}(t)$ for $\frac{1}{2} \leq t \leq 1$, give new weights and control points for $\overline{C}_R(u)$i.e.,

$$
\begin{aligned}
w_{R,0} &= \frac{1}{8}w_0 + \frac{3}{8}w_1 + \frac{3}{8}w_2 + \frac{1}{8}w_3, \\
w_{R,1} &= \frac{1}{4}w_1 + \frac{1}{2}w_2 + \frac{1}{4}w_3, \\
w_{R,2} &= \frac{1}{2}w_2 + \frac{1}{2}w_3, \\
w_{R,3} &= w_3. \\
\overline{B}_{R,0} &= \left(\frac{1}{8}\overline{B}_0 w_0 + \frac{3}{8}\overline{B}_1 w_1 + \frac{3}{8}\overline{B}_2 w_2 + \frac{1}{8}\overline{B}_3 w_3\right)/w_{R,0}, \\
\overline{B}_{R,1} &= \left(\frac{1}{4}\overline{B}_1 w_1 + \frac{1}{2}\overline{B}_2 w_2 + \frac{1}{4}\overline{B}_3 w_3\right)/w_{R,1}, \\
\overline{B}_{R,2} &= \left(\frac{1}{2}\overline{B}_2 w_2 + \frac{1}{2}\overline{B}_3 w_3\right)/w_{R,2}, \\
\overline{B_{R,3}} &= (\overline{B}_3 w_3)/w_{R,3} = \overline{B}_3.
\end{aligned}
$$

Calculating Points at Parametric Intervals

Another simple method to obtain the distance between the cubic NURBS curve and the biarc would be using parametric interval. This would involve obtaining points on the cubic at a regular parametric intervals .i.e. at $C(0.0)$, $C(0.1)$, $C(0.2)$ etc. The distance of those points to the centre of the biarc could then be calculated. The subdivision of the NURBS curve, if the biarc is not within the tolerance limit, would not be required instead, the parametric values of the curve could be stored and the curve between those parameters could be considered. However, this method largely depends on the parametric distribution of the points on the curve. A more accurate method would be to obtain the points on the curve at equal angular intervals, i.e.,

at $C(0)$, $C(t_1)$, $C(t_2)$ etc, which results in equiangular distribution of the points. Although obtaining points at parametric intervals is simple, the equiangular method seems to be more accurate in terms of sampling (or selecting) points on the curve as it is independent of the parametric distribution of the points on the curve.

7.4 Implementation Algorithm

The proposed scheme has been implemented in two steps. For the first step, the given set of Hermite data is interpolated by the NURBS curve. In the second step, the NURBS curve obtained is approximated by the biarcs. It is ensured that the biarc fitting is performed within a tolerance limit of the original NURBS curve. As explained in section 4.1, the extra degree of freedom of the biarc, expressed as θ, has been chosen to be $\theta = \dfrac{4\psi - W}{2}$ (equation (4.25)), for this implementation. The *Recursive_Biarc_Approx* method defined in the algorithm is recursive. If a biarc is not within the tolerance limit of the cubic NURBS, the NURBS curve is subdivided into two curve segments; each segment is further subdivided if it is till not within the tolerance. This process is repeated recursively until each segment of the NURBS curve is approximated by a biarc within the tolerance limit. The steps involved are as follows:

1. Obtain the required set of control points $\overline{V}_0, \overline{V}_1, \cdots, \overline{V}_n$ and respective tangents $\overline{H}_0, \overline{H}_1, \cdots, \overline{H}_n$ on them. These points together with the tangents are the Hermite data that are used for constructing the cubic NURBS curve.

2. While $i = 0 \rightarrow n - 1$ do

(a) Given each pair of Hermite data $\overline{V}_i$, $\overline{H}_i$ and $\overline{V}_{i+1}$, $\overline{H}_{i+1}$, construct cubic NURBS curve $\overline{C}_i$, using algorithm in section 5.3.

(b) *Recursive_Biarc_Approx*($\overline{C}_i$)

i. Construct biarc B_i, using algorithm in section 4.2.

ii. Calculate distance d_i, of B_i with curve $\overline{C}_i$, using method in section 7.2.

iii. if $d_i > \epsilon$ then

A. Decompose $\overline{C}_i$ to yield $\overline{C}_{i,L}$ and $\overline{C}_{i,R}$, using method in section 7.3.

B. *Recursive_Biarc_Approx*($\overline{C}_{i,L}$).

C. *Recursive_Biarc_Approx*($\overline{C}_{i,R}$).

Chapter 8

Applications

In this chapter, the applications of the NURBS curve and the biarc curve are discussed further. Examples are provided to illustrate their applicability.

8.1 Applications of NURBS Curve

NURBS curves have been used in Computer Aided Design and Computer Aided Geometric Design, for many years. Their applications vary from the design of airplane, automobile bodies and engines, rail-road tracks, bicycle tracks, to a shoe sole, sport equipment, bottle etc. NURBS curve have found an increasing use in graphic design due to its ability to provide a convenient design environment. They can be modified in a stable and a non-complex manner. These curves can also be transformed to create surfaces of desired shapes. Many examples exist in the literature that illustrate the usability of the NURBS curve ([6], [13], [24], [27] and [21]).

The examples that are used in this chapter have been kept simple. They include

data sampled from circular arcs. This assists in evaluating the accuracy of the proposed NURBS curve fitting scheme. The two shapes that have been used as examples here are a camshaft model and a ping-pong paddle.

Initially, a set of points and tangents is defined such that they describe the desired shape. For example, the set of data points defined for a camshaft (refer to Table 8.1) consists of points $\overline{V}_0$, $\overline{V}_1$, $\overline{V}_2$, $\overline{V}_3$ with corresponding tangents $\overline{H}_0$, $\overline{H}_1$, $\overline{H}_2$ and $\overline{H}_3$. The data sets for the camshaft model and the ping-pong paddle are given Tables 8.1 and 8.2 respectively. For both examples, the last curve segment to be drawn is from the last point in the data set to the first point. For example in the camshaft model, the last curve segment is drawn from ($\overline{V}_3$,$\overline{H}_3$) to ($\overline{V}_0$,$\overline{H}_0$) and, for ping-pong paddle, it is from ($\overline{V}_5$,$\overline{H}_5$) to ($\overline{V}_0$,$\overline{H}_0$).

The NURBS curve fitting scheme works as follows. For a pair of Hermite data {($\overline{V}_0$,$\overline{H}_0$) and ($\overline{V}_1$,$\overline{H}_1$) }, the scheme first tries to fit a NURBS curve on it. For instance, in Table 8.1, the first pair of points belongs to a circular arc thus, the scheme regenerates a circular arc. It then considers the second pair of points i.e. {($\overline{V}_1$,$\overline{H}_1$) and ($\overline{V}_2$,$\overline{H}_2$)}. The second set does not belong to a circular arc and hence, a cubic NURBS curve is interpolated by the scheme. Figure 8.1 labels each of the curve segments as a circular arc or a non-circular cubic. It is noticeable that the curve fitting scheme ensures that the shape of the camshaft is preserved. The method moves to the next pair of data points for interpolation until the end of the data set. Figure 8.2 shows the shape of a ping-pong paddle with the data points and the segments of the curve that are circular arcs. These circular arcs, after being regenerated, are in cubic NURBS curve or a degree-raised quadratic rational NURBS curve.

Table 8.1: Dataset for the camshaft model

Points	Coordinates	Tangents	Coordinates
$\overline{V}_0$	0.0, 3.0	$\overline{H}_0$	0.0, 1.0
$\overline{V}_1$	1.7, 4.7	$\overline{H}_1$	1.0, 0.0
$\overline{V}_2$	4.3, 3.0	$\overline{H}_2$	0.0, -1.0
$\overline{V}_3$	1.7, 1.3	$\overline{H}_3$	-1.0, 0.0

Table 8.2: Dataset for the ping-pong paddle

Points	Coordinates	Tangents	Coordinates
$\overline{V}_0$	0.0, 3.0	$\overline{H}_0$	0.0, 1.0
$\overline{V}_1$	1.25, 3.25	$\overline{H}_1$	1.0, 0.0
$\overline{V}_2$	3.75, 4.50	$\overline{H}_2$	1.0, 0.0
$\overline{V}_3$	5.25, 3.0	$\overline{H}_3$	0.0, -1.0
$\overline{V}_4$	3.75, 1.50	$\overline{H}_4$	-1.0, 0.0
$\overline{V}_5$	1.25, 2.75	$\overline{H}_5$	-1.0, 0.0

8.2 Applications of biarcs

This section discusses some applications of the biarc fitting scheme. Although most of the design in Computer Aided Design (CAD) is done using NURBS curve, biarcs are used when the designs are to be machined in Computer Aided Machining (CAM). CAM involves the use of CNC machines, the toolpaths for which usually consists of only circular arcs and/or straight lines (Bolton [4] and Walton & Meek [16]). Therefore, the designs to be machined need to be approximated with circular arcs

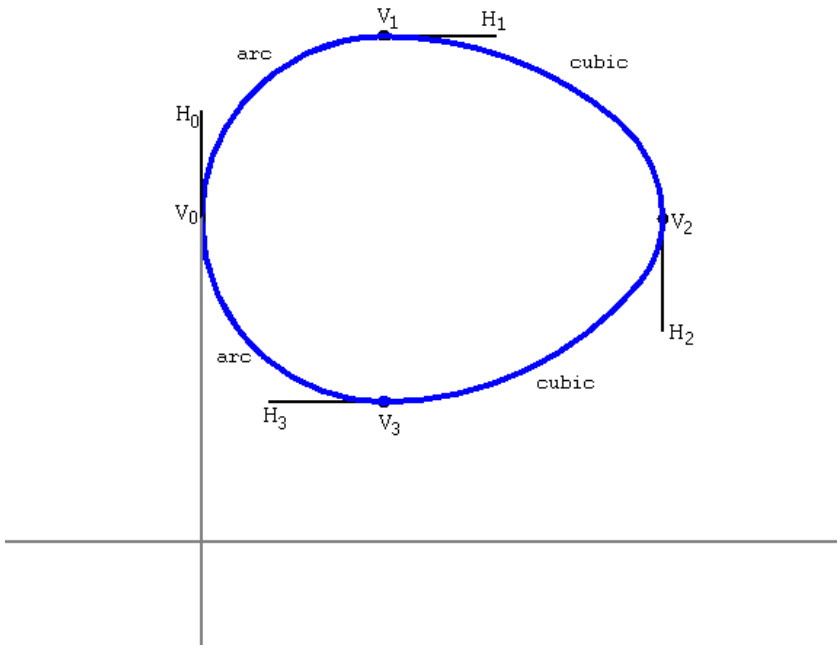

Figure 8.1: Camshaft using NURBS curve fitting.

and or straight line segments before being machined. The biarc approximation of a NURBS curve is much more efficient and accurate than the traditional straight line approximation of a NURBS curve.

The goal here is to illustrate that the scheme has the ability to approximate 2D shapes using biarcs effectively and efficiently. 'Effectively' is used in terms of the proximity of the biarc constructed, to the original curve. The measure of this closeness of the biarc to the original curve, is called the tolerance limit. The tolerance limit typically has a value ranging between 10^{-1} to 10^{-7} inches; however, it depends on the requirement of the system. 'Efficiency' of the scheme (or any biarc fitting scheme), is defined as the number of biarcs that the method produces to approximate the original curve, within the predefined tolerance limit. For experimental purposes, the tolerance

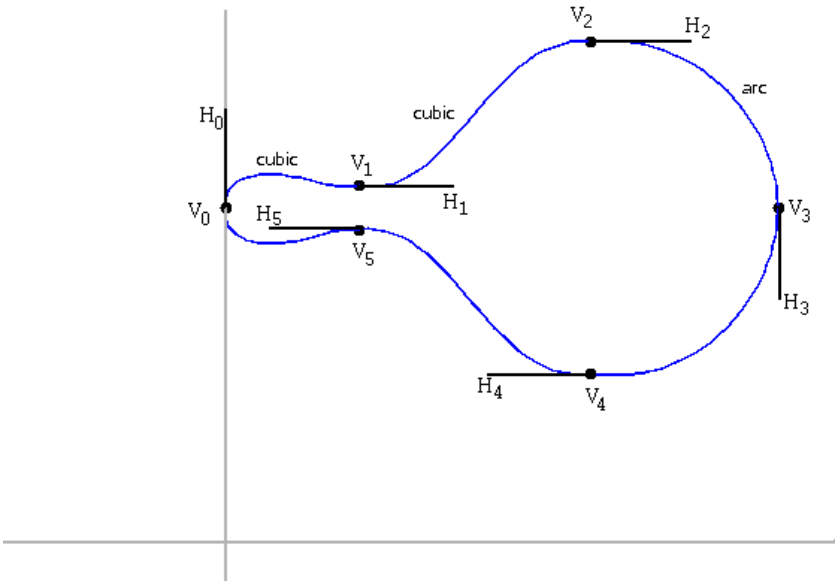

Figure 8.2: Ping-Pong Paddle using NURBS curve fitting.

limit values are then subsequently reduced and the number of biarcs recalculated. It is expected that the number of biarcs should increase as the tolerance limit is reduced. The rate of increase in the number of biarcs determines the performance of the scheme. The NURBS curve fitting scheme first generates the curve on the initial data points and, the biarc fitting scheme then approximates the NURBS curve. Figures 8.1 and 8.2 illustrate two shapes drawn using NURBS curve; a camshaft and a ping-pong paddle. The details of these figures and the data sets have been discussed in the previous section. The biarc approximations of these shapes are shown in Figures 8.3 and 8.4.

8.3 Results

This section discusses the results obtained from generating and approximating the two shapes, using the algorithm in section 7.4. The results obtained from the biarc curve fitting of the camshaft and the ping-pong paddle are compared with a recently introduced biarc fitting scheme by Kimia et al. [1]. Kimia's method seems to be suitable as it uses the criterion for minimizing the difference in the curvature of the two arcs of the biarc, which is different from that of the proposed scheme. Although the algorithm for the implementation remains the same, only the biarc fitting scheme has been replaced with that of Kimia's.

Results for Camshaft

The camshaft model constructed using biarcs is shown in Figure 8.3. This figure has been generated for the case where the tolerance limit value is 10^{-5}; 70 biarcs were generated. The control polygon of the NURBS curve that generated the original camshaft curve is also shown. The variation in the number of biarcs with the increase in tolerance limit value is shown below in Table 8.3. From Table 8.3, there is very little difference between the two methods while approximating a camshaft in terms of the number of biarcs used. For the case where the tolerance limit $\epsilon = 10^{-1}$, the number of biarcs is the same, but the trend is to deviate from each other as the tolerance limit is decreased.

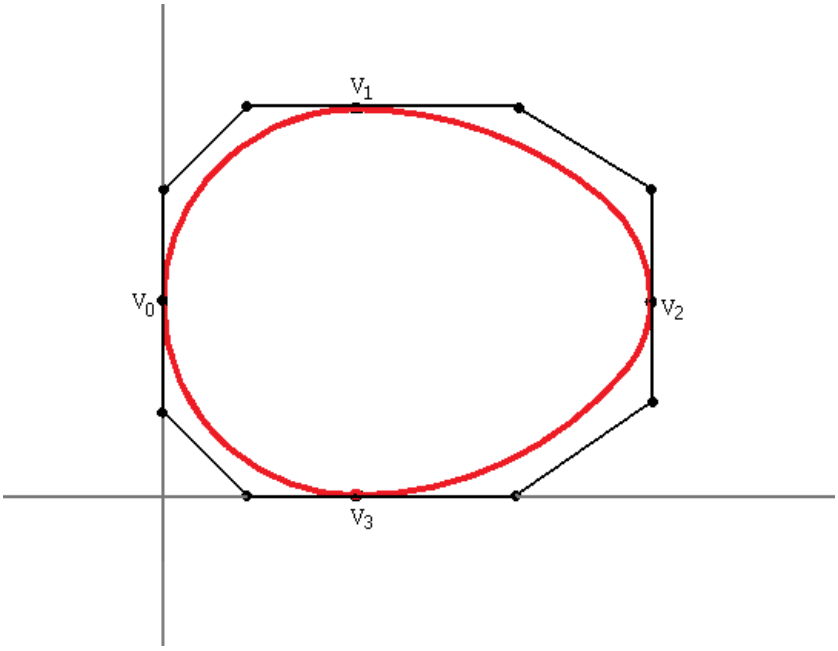

Figure 8.3: A biarc fitted camshaft: tolerance limit = 10^{-5}, biarcs = 70.

Results for Ping-Pong paddle

The shape of the Ping-pong paddle generated by the NURBS curve is shown in Figure 8.2. The biarc fitting method was then used to approximate the NURBS curve. Figure 8.4 shows the ping-pong paddle generated when the tolerance limit is 10^{-5}. The number of biarcs used to approximate the curve with such a low tolerance limit is 196. On the other hand, a huge difference is recorded in the number of biarcs required to approximate the same curve when Kimia et al.'s method is applied on the same model. The reason could be the presence of more cubic NURBS curve in the ping-pong paddle as compared to the camshaft. The cause could be the way the biarc is constructed in Kimia's method. Kimia uses the criteria of minimizing the curvature of the two arcs while constructing the biarc that; this may result in the use

Table 8.3: Biarc Fitting: Camshaft

	No. of Biarcs	
Tolerance limit	Proposed method	Kimia's method
10^{-1}	6	6
10^{-2}	10	18
10^{-3}	18	26
10^{-4}	32	44
10^{-5}	70	86
10^{-6}	142	166

of a large number of biarcs to approximate the shapes. Figures 8.5 and 8.6 provide

Table 8.4: Biarc Fitting: Ping-Pong paddle

	No. of Biarcs	
Tolerance limit	Proposed method	Kimia's method
10^{-1}	26	20
10^{-2}	34	54
10^{-3}	44	156
10^{-4}	82	376
10^{-5}	196	1116
10^{-6}	420	1712

the trends of the two methods graphically, for the two scenarios discussed.

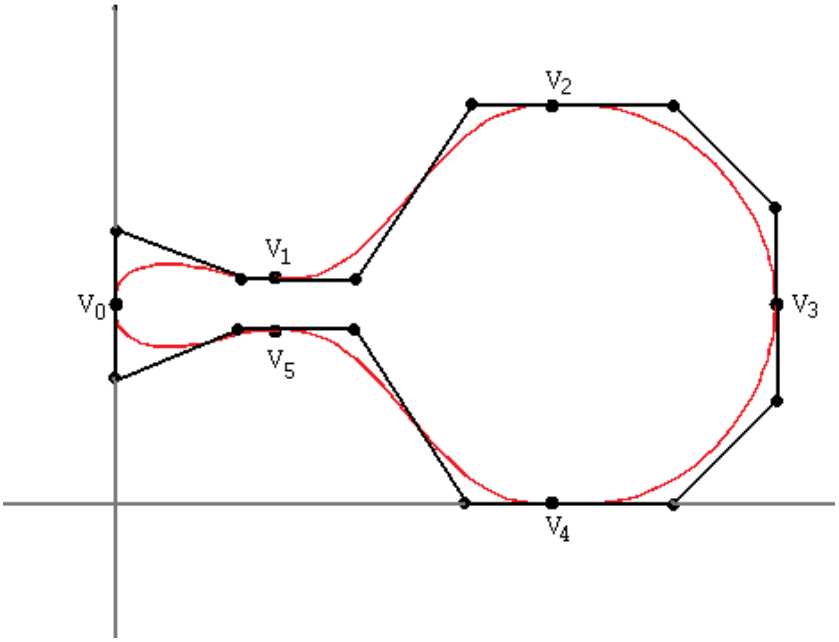

Figure 8.4: A biarc fitted ping-pong paddle: tolerance limit = 10^{-5}, biarcs = 196.

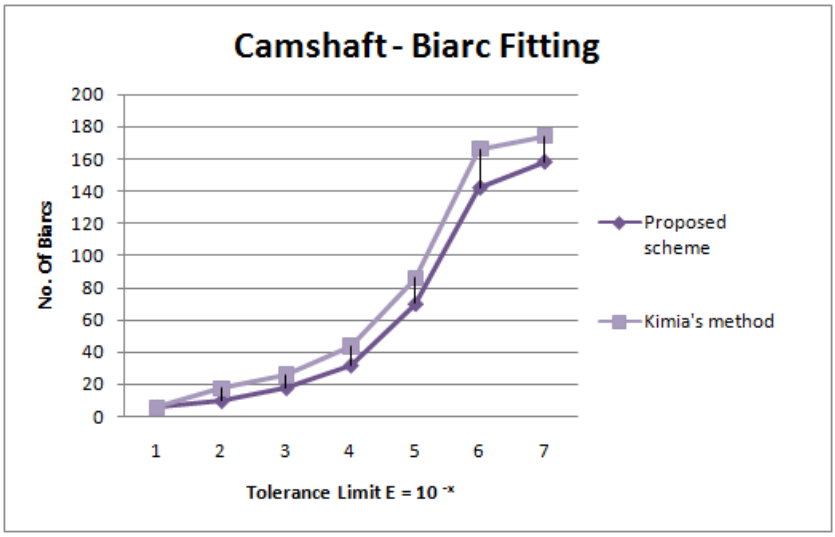

Figure 8.5: Camshaft interpolation

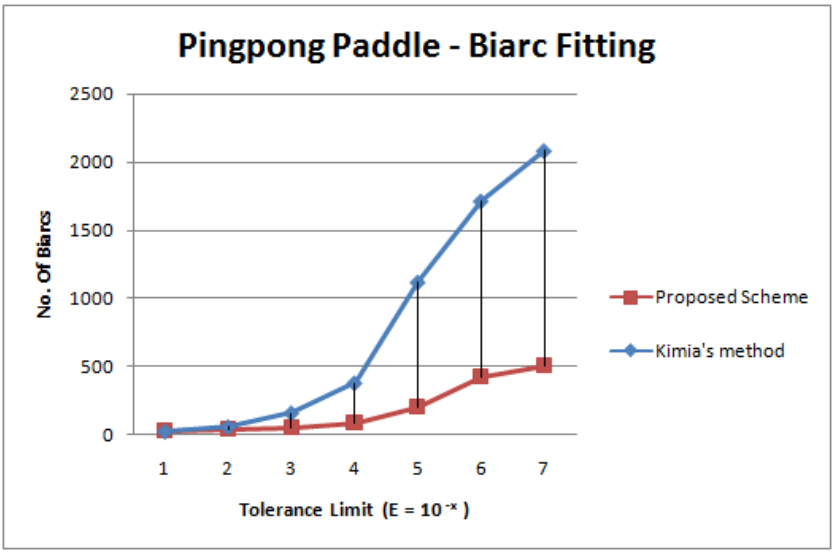

Figure 8.6: Ping-Pong paddle interpolation

Chapter 9

Conclusion

Rational curves have become a significant tool in the design of curves and surfaces. They have the ability to perform a accurate shape-preserving interpolation with straight forward mathematical calculations. Piegl [24], Piegl [25], Goodman & Unsworth [28] and Seymour and Unsworth [29] for example, are some of the works that tend to exploit the properties of the rational curves for fitting data points while preserving the shape of the interpolation. It is noticeable that, as the degree of the curve increases, the complexity tends to increase too. This justifies the use of lower degree curves like quadratic (2), cubic (3) or quartic (4) curves, in most of the literature.

In the scheme proposed in this thesis, the property of quadratic NURBS curves to exactly represent circular arcs, has been utilized. Given a set of Hermite data (points $\overline{A}$, $\overline{B}$ with tangents $\overline{T}_A$, $\overline{T}_B$), the scheme draws two quadratic NURBS curve ($\overline{C}_L(u)$ and $\overline{C}_R(u)$) from either endpoint. The scheme then performs a linear blending of the two curves. If the data points belong to a circular arc, a circular arc is reproduced or

else a non-circular cubic NURBS curve is obtained. The constrained linear blending of two quadratic NURBS curves to produce a cubic ensures that the shape of the original dataset is preserved during the interpolation. In similar work, Farin [33] reproduced the circular arc using a rational cubic curves from G^1 Hermite data of a circular arc. The method differs in the way the circular arcs are obtained. Farin achieves this by the degree-raised rational cubics. In the proposed work, the two quadratic NURBS curves are linearly blended to obtain a cubic representing a circular arc. Further, if the points do not belong to a circular arc, a shape preserving cubic NURBS curve is obtained. Hence, the scheme introduced in this thesis is generalized in terms of interpolation compared to Farin's method. The scheme in this thesis was proposed well before the work by Farin [33] was published, i.e. the thesis proposal was submitted in late May 2007 and got reviewed by the Graduate committee on June 16^{th}, 2007 whereas, Farin's work was officially accepted on 10 January, 2008 and published in the journal 'Computer-Aided Design' in April 2008.

The significance of the proposed scheme is the way the problem of curve and biarc fitting has been addressed together for shape preserving interpolation. For instance, a camshaft is designed using the NURBS curve fitting scheme and a tool path is generated for it using the biarc curve fitting method. The regenerated circular arc (or arcs) can be immediately applied into the CNC toolpath, and the non-circular cubic NURBS curve can be approximated with the biarcs. This would greatly reduce the overhead for creating toolpaths in the machines. Further, the use of NURBS curve ensure that the curve is shape-preserving while biarc fitting ensures that good biarcs are drawn within a pre-requisite tolerance limit to the curve. Thus, the two

schemes are closely related when a toolpath is required of the designs in Computer Aided Machining.

In conclusion, the contribution of this work can be summarized as follows. The use of the parameter θ to determine a unique biarc from a family of biarcs for given G^1 Hermite data. The scheme enables the use, in a wide range of applications, of both C-shaped and S-shaped biarcs. The applicability of this scheme is twofold: NURBS curves can be used for design in Computer Aided Designing, while biarcs can be used to create toolpaths for designs that need to be machined in Computer Aided Machining applications. For future work, a more detailed analysis of the biarc curve fitting scheme will be performed for a more systematic determination of an optimal or near optimal value of the parameter θ for better biarc approximations.

Bibliography

[1] Benjamin B. Kimia, Ilana Frankel, and Ana-Maria Popescu, "Euler spiral for shape completion", *Int. J. Comput. Vision*, vol. 54, no. 1-3, pp. 157–180, 2003.

[2] Yawei Ma Yuan-Shin Lee and Koc Bahattin, "Smoothing STL files by max-fit biarc curves for rapid prototyping", *Rapid Prototyping Journal*, vol. 6, no. 3, pp. 186–203, 2005.

[3] Millan K. Yeung and Desmond J. Walton, "Curve fitting with arc splines for NC toolpath generation", *Computer-Aided Design*, vol. 26, no. 11, pp. 845 – 849, 1994.

[4] K. M. Bolton, "Biarc curves", *Computer-Aided Design*, vol. 7, no. 2, pp. 89–92, 1975.

[5] D.J. Walton and D.S. Meek, "Approximation of quadratic Bézier curves by arc splines", *Journal of Computational and Applied Mathematics*, vol. 54, no. 1, pp. 107 – 120, 1994.

[6] D S Meek and D J Walton, "Approximating quadratic NURBS curves by arc

splines.", *Journal of Computational and Applied Mathematics*, vol. 25, no. 6, pp. 371 – 376, 1993.

[7] D S Meek and D J Walton, "Approximating smooth planar curves by arc splines.", *Computer-Aided Design*, vol. 59, no. 2, pp. 221–231, 1995.

[8] D.J. Walton and D.S. Meek, "Approximation of a planar cubic Bézier spiral by circular arcs", *Journal of Computational and Applied Mathematics*, vol. 75, no. 1, pp. 47 – 56, 1996.

[9] C.J. Ong, Y.S. Wong, H.T. Loh, and X.G. Hong, "Optimization approach for biarc curve-fitting of B-spline curves", *Computer-Aided Design*, vol. 28, no. 12, pp. 951 – 959, 1996.

[10] Young Joon Ahn, Hong Oh Kim, and Kyoung-Yong Lee, "G^1 arc spline approximation of quadratic Bézier curves", *Computer-Aided Design*, vol. 30, no. 8, pp. 615–620, 1998/7.

[11] L.A. Piegl and W. Tiller, "Biarc approximation of NURBS curves", *Computer-Aided Design*, vol. 34, no. 11, pp. 807–814, 2002.

[12] D. S. Meek and D. J. Walton, "The family of biarcs that matches planar, two-point G^1 hermite data", *Journal of Computational and Applied Mathematics*, vol. in press, 2007.

[13] Gerald Farin, *Curves and Surfaces for CAGD: A practical guide*, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2002.

[14] M.A. Sabin, "The use of piecewise forms for the numerical representaion of shapes.", Tech. Rep. 60/1977, Hungarian Academy of Sciences, 1977.

[15] B-Q Su and D-Y Liu, *Computational Geometry-Curve and Surface Modeling*, Academic Press, USA, 1989.

[16] D S Meek and D J Walton, "Approximation of discrete data by G^1 arc splines", *Computer-Aided Design*, vol. 24, no. 6, pp. 301–306, 1992.

[17] D.B. Parkinson and D.N. Moreton, "Optimal biarc-curve fitting", *Computer-Aided Design*, vol. 23, no. 6, pp. 411 – 419, 1991.

[18] A. W. Nutbourne and R. R. Martin, *Differential Geometry Applied to Curve and Surface Design. Vol. I:Foundations*, Ellis Horwood, UK, 1988.

[19] L.A. Piegl and W. Tiller, "Data approximation using biarcs", *Engineering with Computers*, vol. 18, no. 1, pp. 59 – 65, 2002.

[20] Duncan Marsh, *Applied Geometry for Computer Graphics and CAD*, Springer, 1999.

[21] Les Piegl and Wayne Tiller, *The NURBS book*, Springer-Verlag, London, UK, 1995.

[22] D.N. Moreton, D.B. Parkinson, and W.K. Wu, "Application of a biarc technique in CNC machining", *Computer-Aided Engineering Journal*, vol. 8, no. 2, pp. 54 – 60, 1991.

[23] Richard H. Bartels, John C. Beatty, and Brian A. Barsky, *An introduction to splines for use in computer graphics & geometric modeling*, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1987.

[24] Les Piegl, "On NURBS: a Survey", *IEEE Comput. Graph. Appl.*, vol. 11, no. 1, pp. 55–71, 1991.

[25] Les Piegl, "Curve fitting algorithm for rough cutting", *Computer-Aided Design*, vol. 18, no. 2, pp. 79–82, 1986.

[26] Xujing Yang and Zezhong C. Chen, "A practicable approach to G^1 Biarc approximations for making accurate, smooth and non-gouged profile features in CNC contouring", *Computer-Aided Design*, vol. 38, no. 11, pp. 1205–1213, 2006.

[27] T. N. T. Goodman and K. Unsworth, "Shape preserving interpolation by curvature continuous parametric curves", *Computer-Aided Geometric Design*, vol. 5, no. 4, pp. 323–340, 1988.

[28] T. A. Foley, T. N. T. Goodman, and K. Unsworth, "An algorithm for shape preserving parametric interpolating curves with G^2 continuity", pp. 249–259, 1989.

[29] Chris Seymour and Keith Unsworth, "Interactive shape preserving interpolation by curvature continuous rational cubic splines", *Journal of Computational and Applied Mathematics*, vol. 102, no. 1, pp. 87–117, 1999.

[30] Qi Duan, K. Djidjeli, W. G. Price, and E. H. Twizell, "Constrained control and

approximation properties of a rational interpolating curve", *Inf. Sci.*, vol. 152, no. 1, pp. 181–194, 2003.

[31] T. N. T. Goodman and D. S. Meek, "Planar interpolation with a pair of rational spirals", *Journal of Computational and Applied Mathematics*, vol. 201, no. 1, pp. 112–127, 2007.

[32] John E. Lavery, "Shape-preserving, first-derivative-based parametric and nonparametric cubic L1 spline curves", *Computer-Aided Geometric Design*, vol. 23, no. 3, pp. 276–296, 2006.

[33] Gerald Farin, "Geometric hermite interpolation with circular precision", *Comput. Aided Des.*, vol. 40, no. 4, pp. 476–479, 2008.