Smooth Path Planning using Biclothoid Fillets for High Speed CNC Machines[☆]

Abbas Shahzadeh^a, Abbas Khosravi^a, Troy Robinette^b, Saeid Nahavandi^a

^a*Deakin University, 75 Pigdons Road, Waurn Ponds VIC 3216, Australia*
^b*ANCA Motion, 1 Bessemer Rd, Bayswater North VIC 3153, Australia*

Abstract

In high speed Computer Numerical Control (CNC) machines, cut velocities greater than 60 metres per minute and accelerations higher than 2g are used. In such high feedrates and accelerations, even a small discontinuity in curvature or in tangency can result in jerk spikes and consequently in machine vibrations, poor cut quality and decreased lifespan of the equipment. To prevent these consequences, using path smoothing techniques is necessary. Many path smoothing methods have been proposed in the literature to eliminate toolpath discontinuities. However the usage of almost all of these techniques is limited to purely linear toolpaths. In this paper a new path smoothing method using biclothoid fillets is introduced. The proposed method can be used to convert any given path consisted of lines and arcs to a curvature continuous path. The generated path is arc length parameterised which makes it easy to interpolate. The distance between the G2-continuous path and the original toolpath is limited to an adjustable tolerance. The proposed method has been tested on a CNC laser cutting machine and the results are reported. The main contribution of this paper is a fillet fitting method which is not limited to line to line transitions. The proposed smoothing fillets can be fitted between two arcs or a line and arc as well. A comparison with Bezier fillets, shows that using the proposed method results in a smoother curvature profile, higher feedrates and shorter cycle times.

Keywords:
CNC, path planning, corner smoothing, jerk, G2 continuity, clothoid

1. Introduction

Standard part programs processed by CNC controllers, define toolpaths which are composed of several lines and arcs [1]. At each line to line, line to arc or arc to arc transition, careful considerations are required to ensure that the physical limits of the machine are not exceeded. For example when the machine is moving at a constant feedrate, at the point where two successive non-tangent linear moves meet, there will be a sudden change in the velocity of the participating joints. Therefore, the controller has to foresee these transition points and reduce the path velocity to limit the side effects of a step change in the velocity. Similarly at line to arc and arc to arc transitions, even when the two moves are completely tangent, curvature discontinuities have to be addressed. Any discontinuity in the curvature results in a step change in acceleration and consequently in a jerk spike which can have detrimental effects on the quality of cut.

The same issue needs to be addressed in path planning for mobile robots and also in highway and railway design.

The problem can be explained in a more tangible way in the context of wheeled vehicles. Imagine a car which is being driven on a road. For moving on a straight line, the angle of the front wheels with the chassis of the car needs to be zero. However for turning on a circular bend with a radius equal to 15 metres, for a typical car, the angle of the front wheels has to be about 10 degrees. In a non-standard road where a straight line segment is immediately followed by a circular bend, the driver will have to change the angle of the front wheels from zero to 10 degrees instantaneously which is physically impossible. So in order to follow the road, the car has to stop, reorient its front wheels and start moving again [2]. To prevent cases like the given example, smoothing curves like clothoids are used in highway and railway design. The role of the smoothing curves is to prevent sudden changes in the curvature and to eliminate curvature discontinuities.

In the case of CNC machines, there are two modes of operation known as Exact Stop Mode and Continuous Mode [1]. In the Exact Stop Mode, discontinuities in the toolpath do not make any difference in the overall performance. The reason is that in this mode, the machine stops after each move and before starting the next move. However in the Continuous Mode, in order to have a jerk-limited smooth movement, planning a curvature continuous path without any discontinuities in curvature or tangency is necessary.

As mentioned before, the same problem has been studied for mobile robots and autonomous vehicles [2, 3, 4, 5]. However, CNC applications have an important additional requirement that needs to be considered in path smoothing. In these applications, the accuracy of the travelled path has to be maintained. To satisfy this requirement, the deviation introduced by the smoothing algorithm has to be limited. In other words the maximum distance between the smoothed path and the original path should be within a specified tolerance.

The rest of this paper is organized as follows. In section 2, the related work is discussed and the gaps in the literature which this paper is trying to fulfil are briefly explained. In section 3 the problem that this paper is trying to solve is outlined using a sample part program. A short introduction to clothoids is given in section 4. Section 5, introduces biclothoid fillets which are the fillets used for corner smoothing in this paper. Methods and algorithms used for fitting biclothoid fillets at line to line, line to arc or arc to arc transitions are explained in section 6. Section 7 covers the methods used in this work for measuring and limiting the maximum distance between the smoothed path and the original path. Formulas for calculating the maximum acceleration and jerk when moving on a clothoid and the maximum permissible velocity are given in section 8. This section is followed by a case study in section 9. In section 10, the proposed method is compared with the method from [6]. Finally section 11, concludes the paper.

2. Related Work

Papers discussing the problem of path smoothing for CNC machines can be divided into two main categories. In the first group of papers, methods for finding smooth splines that pass through a large number of input points are proposed [7, 8, 9]. Spline fitting is used when the toolpath is composed of a large number of small linear segments (also known as micro-lines). The second group of papers, discuss the problem of corner smoothing. Corner smoothing is used when the moves generated by the CAD-CAM software are longer (in comparison to micro-lines which are better smoothed using splines) and can be either lines or arc segments. This work falls in the latter category.

Several methods are proposed for corner smoothing in the literature. Polynomial curve splines are used as transition curves in some works. Erkorkmaz et al. in [10], use quintic splines composed of two or three quintic spline segments as smooth transition curves. The choice between two or three segments depends on the type of the servo controllers used in the machine. To get near arc length parameterisation, the authors of [10] use the method proposed by Yang and Wang in [11]. Despite using the mentioned technique for near arc length parameterisation, velocity discontinuities still exist in the generated spline [10] which have to be resolved by using special interpolation techniques [10, 12].

Yutkowitz in [13] uses two back to back quartic polynomials per axis to round the corners. For each corner, the curvature of the smoothing curve becomes maximum at the middle of the fillet where the two quartic curves join. In order to decrease feedrate fluctuations the value of the maximum curvature for each corner is optimised so that the polynomial parameter is close to the arc length. In other words, the smoothing curve is nearly arc length parameterised.

Beudaert et al. in [14] use cubic splines. They smooth each joint separately by fitting a cubic spline to the position stream and then measure the resulting geometric deviation. An optimisation algorithm tries to maximise the feedrate while keeping the introduced error below a specified tolerance.

Bezier curves are also used for corner smoothing. Yang and Sukkarieh in [15] use two back to back cubic Bezier curves to generate smooth transitions from a line to another line at the corners. Their method is intended for mobile robots and generates a continuous curvature path with an upper bound limit on curvature. In the intended application, the path deviation is not important and is not taken into account.

Sencer et al. in [6] also use Bezier curves to smooth line to line transitions. Instead of using two cubic Bezier curves, one quintic Bezier curve is used. They have devised a method to generate a smoothing curve with minimum curvature peak for any corner angle. They show that by decreasing the maximum curvature of the transition curve, higher feedrates can be achieved. Since Bezier curves are not arc length parameterised, a special interpolation method has to be used in order to prevent feedrate fluctuations [6, 15].

Bi et al. in [16] and [17] use a particular type of cubic Bezier curves first introduced in [18]. In [18], Walton and Meek, use cubic Bezier curves which normally have four control points. However, they merge two of the control points and place them at the corner. This ensures that the curvature profile of the Bezier curve has only one extremum at the middle of the curve [18]. Walton and Meek do not attempt to limit the deviation between the smoothing Bezier curve and the original curve. Bi et al. in [16] and [17], however, measure and limit the deviation to make sure that it is bounded to a predefined value.

Fan et al. in [19] use two back to back quartic Bezier curves to achieve G3 continuity. Each of the Bezier curves has 5 control points and the two back to back curves share one control point which leaves 9 control points to calculate. In determining the control points, they minimise the curvature variation energy (CVE) [19] in order to minimise the cycle time and improve the smoothness.

B-spline curves are also used as path smoothing fillets. Zhao et al. in [20] use cubic B-splines with 5 control points. By dividing the B-spline into two cubic Bezier curves they prove that the maximum curvature falls exactly in the middle of the B-spline. They limit the feedrate based on the maximum curvature of the smoothing B-spline.

Beudaert et al. in [21] also use cubic B-splines with 5 control points where the third control point is placed at the corner. Since there are three points on each line of the corner, the curvature at the beginning and at the end of the smoothing curve is zero which matches the curvatures of the comprising lines. They explain that the distance between control points can be adjusted in order to minimise the machining time. However, they also give a rule of thumb and state that if the distance between the first control point and the corner is 1.4 to 1.75 times larger than the distance between the second control point and the corner, satisfying results in terms of machining time can be achieved.

Tulsyan and Altintas in [22] and also Yang and Yuen in [23] use quintic B-splines with 7 control points in order to achieve G3 continuity. One of the 7 control points is placed at the corner so that there are four control points on each line of the corner. This arrangement results in zero curvature and also a zero derivative of curvature at the beginning and at the end of the smoothing curve. Their method is designed for smoothing corners in a 5-axis milling machine. Tulsyan and Altintas in [22] use a septic B-spline with 9 control points for smoothing the orientation.

Pythagorean-hodograph (PH) curves are another group of curves used for corner smoothing. Shi et al. in [24] use the method first introduced in [18]. They use a quintic PH curve which has 6 control points. The second and third control points are merged and also the fourth and fifth control points are merged to ensure a single curvature extremum on the curve. They extract an analytical expression for the maximum curvature of the smoothing curve.

Another method of corner smoothing proposed in [25] and [26] by Tajima and Sencer, uses a single step solution. It should be noted that in most of the algorithms for corner smoothing, at first a smoothing curve is fitted to replace the corner, and then the smoothed path is interpolated. In contrast to the common method, Tajima and Sencer propose smoothing the corner by directly calculating and controlling the jerk, acceleration and velocity when the toolpath is being interpolated. In [25] they explain their algorithm for a 3-axis machine. Using the proposed algorithm, the jerk of one of the joints is always at its maximum to ensure that the smoothing time is minimised. In [26] they extend their algorithm to a 5-axis case.

This paper, uses biclothoid fillets (two back to back clothoids) as transition curves to smooth the corners and eliminate discontinuities in the toolpath. Clothoids have been used in path planning for mobile robots. Shin et al. in [4], use three clothoids to connect every two successive points while matching the tangency and curvature at both ends. Since the intended application is path planning for robots, the deviation between the original path and the generated path is not limited [4]. Brezak and Petrovic in [5], use two clothoids for corner smoothing. They study both line to line and arc to line cases while always assuming zero curvature at one end. The reason for this assumption is that robots mainly move on straight lines. However, this assumption is not valid for CNC machines.

Shahzadeh et al. in [7], use clothoids to generate a G2-continuous clothoid spline with limited deviation from the original toolpath for CNC applications. However, the proposed method can only handle toolpaths that are symmetrical and G1-continuous. In other words, similar to the other works referenced in this section, the method in [7], cannot be used to smooth general line to arc or arc to arc discontinuities.

Clothoids have a number of advantages over other curves for corner smoothing which will be explained in the coming sections of this paper. However, they have not been widely used in CNC applications. The main concern for using clothoids in CNC applications is the computation intensity [15]. In order to interpolate a clothoid, calculation of Fresnel integrals is necessary and unfortunately these integrals do not have a closed form solution. Nevertheless, Fresnel integrals can be accurately (with error smaller than 4×10^{-8}) estimated using a few rational polynomials [27]. A clothoid can also be interpolated using a look up table in real time applications with limited CPU power [28]. Using either of the mentioned techniques, the computational intensity issue can be resolved.

The main limitation of the works referenced above (except for [4] and [5] which are not intended for CNC applications) is that they only discuss line to line transitions. However, in practice, part programs are consisted of both linear and helical (or arc) moves.

Another limitation when using polynomial curves, B-splines or Bezier curves is that for these curves, the arc length cannot be calculated using an analytical formula. For these curves, the arc length can only be estimated by integration [15, 24]. Not having an accurate arc length results in feedrate fluctuations [29, 30, 31]. Special interpolators are needed to decrease the feedrate fluctuations when using parametric curves. For example [9] and [32] use one or more 9th degree polynomials as feedrate correction functions to compensate for the lack of arc length parameterisation. Kinematic corner smoothing methods proposed in [25, 26] have the same problem.

PH curves have the advantage that their arc length can be calculated analytically for any value of the curve parameter. However, despite the fact that PH curves have an analytical arc length, they are not arc length parameterised. As a result again special interpolators are necessary to process these curves [33, 34].

Finally in majority of the papers referenced above, only maximum curvature is considered when calculating the maximum permissible feedrate. Also in some papers, in order to increase the feedrate, the maximum curvature has been minimised. Nevertheless, in section 8 it will be proved that for calculating the maximum permissible feedrate, the derivative of curvature (sharpness) is just as important as the curvature itself.
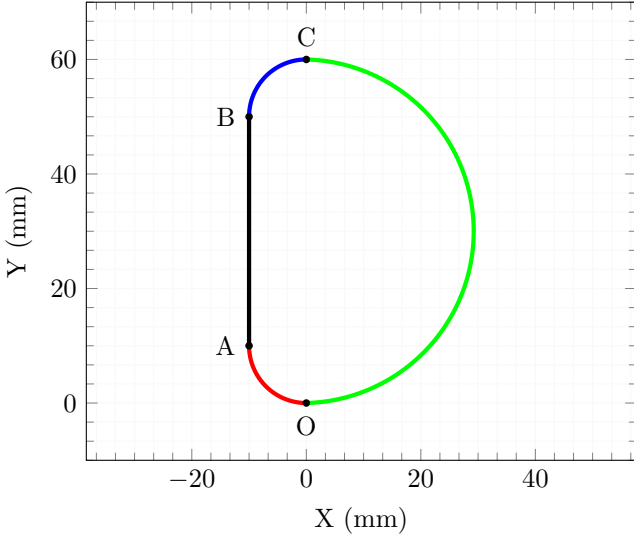
Figure 1: The toolpath generated by executing the part program in listing 1.

Figure 2: Acceleration of the X and Y axes when moving on the toolpath given in figure 1. A, B and C marked on the plot correspond to the same points in figure 1.

3. Problem Statement

In this section, in order to explain the problem that this work is trying to solve, a sample toolpath will be examined. Consider the part program presented in listing 1 below.

Listing 1: The part program used for experiments in this paper. The colours match those of figure 1 (the toolpath).

```
G90
F10000
G2X−10Y10R10 ( red )
G1Y50 ( black )
G2X0Y60R10 ( blue )
G2X0Y0R30.01 ( green )
```

The toolpath corresponding to this part program is plotted in figure 1. This closed toolpath ($OABCO$ in figure 1) starts from the origin and ends at the same point. The colours in the figure match those specified in the G-code program. It should be noted that in G-code language, G1 commands, specify linear moves, while G2 commands stand for clockwise circular arc moves [35].

The toolpath in figure 1 may look very smooth to the eye. However there are three discontinuities in this toolpath. The first discontinuity is at point A where the transition is made from the red arc to the black line. The curvature of the red arc is 0.1 mm^{-1} (the radius of the arc is 10 mm thus the curvature is $\frac{1}{10\ mm}$) while the curvature of the black line is zero. As a result there is a curvature discontinuity at point A where these two moves meet. A similar discontinuity exists at point B between the black line and the blue arc. However the worst case of discontinuity happens at point C (the transition from the blue arc to the green arc following it). Not only the radius changes
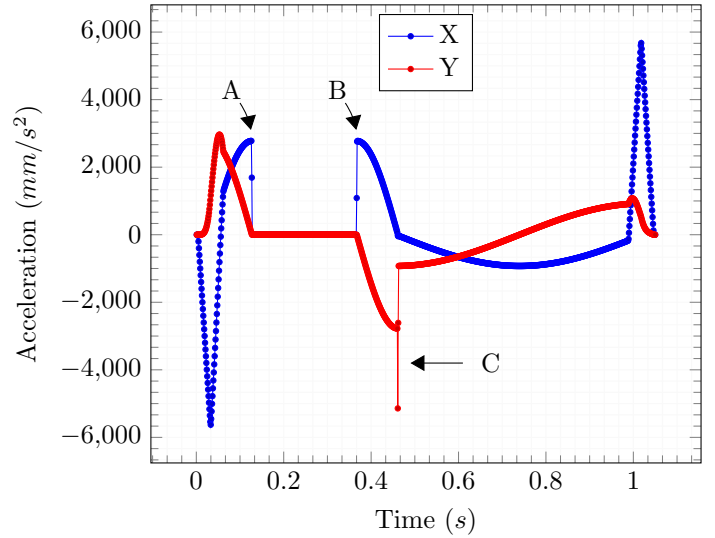
from 10 mm to 30.01 mm, but also there is a 1.48° angle between the two arcs.

The part program from listing 1 was given to a 3 axis CNC laser cutting machine controlled by an AMC5 ANCA Motion CNC controller [36]. The jerk limit was set to 200000 $\frac{mm}{s^3}$ and the acceleration limit was 9800 $\frac{mm}{s^2}$ (1.0g). In the first experiment, transition jerk limiting functionality was disabled on the machine. The resulting acceleration profiles for axes X and Y are plotted in figure 2.

The plots in figure 2 show that at points A and B on the plot, which correspond to the points with the same names in figure 1, the acceleration of the X axis has sudden changes. The cause of this jump in acceleration, is the curvature discontinuity when going from an arc to a line (at point A), or when going from a line to an arc (point B). At point C, due to discontinuity in both tangent angle and curvature, a bigger acceleration jump can be observed (for the Y axis).

The sudden changes in the acceleration translate to big jerk spikes as the plot in figure 3 shows. While the jerk limits for the axes X and Y of the machine were set to 200000 $\frac{mm}{s^3}$, because of the discontinuities in curvature and in tangency, huge jerk spikes greater than 10 times the limit are generated. These jerk spikes result in severe vibrations on the machine. In the experiment, clunking sounds could be heard at points A, B and C of the toolpath (refer to figure 1).

Modern CNC controllers have a mechanism to prevent these jerk spikes. The method that is currently in use in the industry is to slow down at these points so that the generated jerk is within the limits that the machine can deliver. In the second experiment, the transition jerk limiting functionality was turned on and the same toolpath was given to the machine. This time the jerk spikes
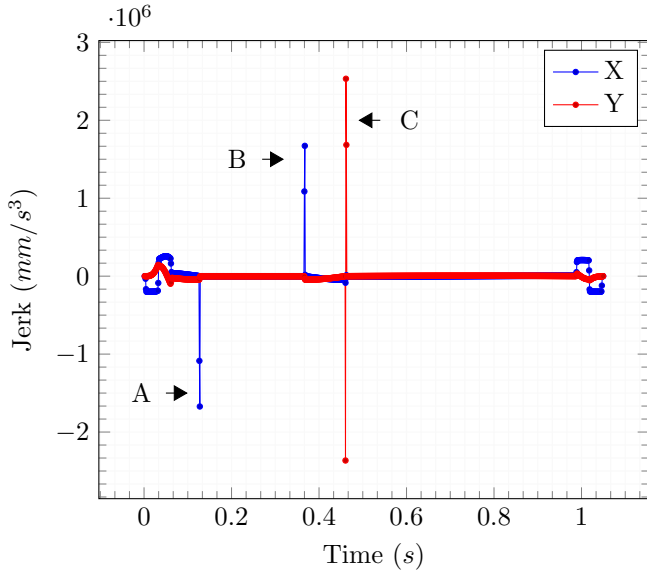
Figure 3: Jerk of the X and Y axes when moving on the toolpath given in figure 1.
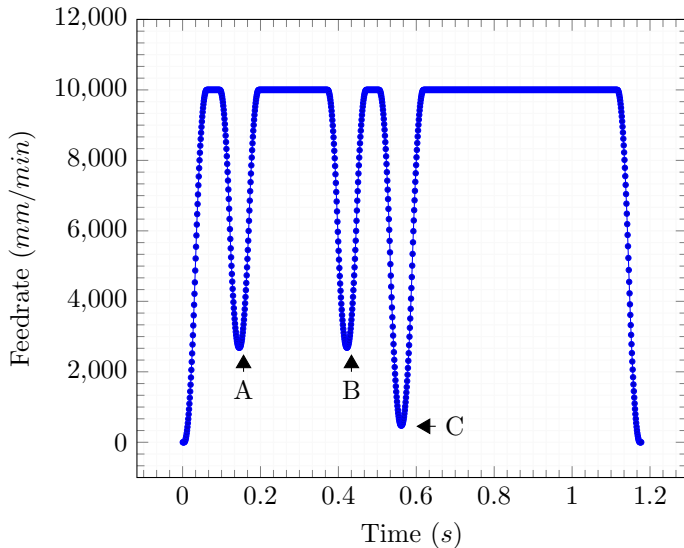

Figure 4: Path velocity when moving on the toolpath given in figure 1. The machine has to slow down before the transition points to limit the magnitude of the jerk spikes.

were not observed and the machine seemed to be moving smoothly with no unusual noises. However there was a considerable increase in the cycle time. The cycle time increased by 20% from $1s$ to $1.2s$. Cycle time increase can be even more prominent in cases where there are many discontinuities in the toolpath. The feedrate profile when the transition jerk limiting functionality is turned on, is depicted in figure 4.

Part programs generated by CAD-CAM software, are rarely curvature continuous and as a result cases like the sample toolpath portrayed in figure 1 are common in the

industry. Consequently to prevent the machine from slowing down at the points of discontinuity or imposing severe jerk spikes on the axes, an algorithm to convert any given toolpath to a smooth curvature-continuous toolpath is needed. Ideally the smoothed toolpath should have a number of characteristics:

a. G2-continuity: The generated toolpath should be G2-continuous. In other words there should be no discontinuities in the tangency or the curvature of the toolpath at any point.

b. Bounded maximum error: The maximum distance between the smoothed toolpath and the original toolpath, should not exceed a specified tolerance. A machine operator or a G-code programmer should be able to set this tolerance to any desirably small value.

c. Analytical arc length formula: Arc length of the path at any point should be analytically formulated. This property is essential for speed profile generation and for preventing feedrate fluctuations [37].

d. Arc length parameterisation: Having the travelled arc length, next point on the curve should be easily calculable. This will facilitate interpolation calculations.

e. Minimum curvature change: It will be proved in following sections that the centripetal jerk is proportional to the rate of change of curvature (known as sharpness). Consequently, having a smaller sharpness will permit higher feedrates without exceeding jerk limits.

This paper proposes a method which satisfies all of the above criteria. It will be demonstrated that the generated path can be processed by the machine with a higher average feedrate without violating jerk or acceleration limits of the machine.

4. Clothoids

A clothoid is a curve whose curvature is a linear function of its arc length. These curves can be used to provide a smooth transition between two curves with different curvatures. In this section, an easy to understand representation of clothoids will be given. Notation used to describe a clothoid is taken from [5].

In a clothoid, according to the definition, the curvature can be expressed as equation 1 where $\kappa_0 = 1/r_0$ is the initial curvature, c is the rate of change of curvature known as sharpness, and s is the arc length of the clothoid[5].

$$\kappa(s) = \kappa_0 + cs \tag{1}$$

All other clothoid equations are derived from equation 1. Curvature by definition is the derivative of the tangent

5

angle ($\theta(s)$) with respect to the arc length (s). Thus, the equation for $\theta(s)$ can be derived by integrating equation 1 as stated in equation 2[5].

$$\theta(s) = \theta_0 + \int_0^s \kappa(t)\,\mathrm{d}t = \theta_0 + \kappa_0 s + \frac{1}{2}cs^2 \qquad (2)$$

In equation 2, $\theta(s)$ is the angle that a tangent to the clothoid at arc length equal to s, makes with the positive direction of the X axis and θ_0 is the tangent angle at the start point. Having the tangent angle $\theta(s)$ from equation 2, the equation for unit tangent vector can be derived as in equation 3 in which, $\hat{\mathbf{i}}$ and $\hat{\mathbf{j}}$ are the unit vectors co-directional with X and Y axes respectively.

$$\begin{aligned}\hat{\mathbf{T}}(s) =& \cos(\theta(s))\hat{\mathbf{i}} + \sin(\theta(s))\hat{\mathbf{j}} \\ =& \cos(\theta_0 + \kappa_0 s + \frac{1}{2}cs^2)\hat{\mathbf{i}} \\ &+ \sin(\theta_0 + \kappa_0 s + \frac{1}{2}cs^2)\hat{\mathbf{j}}\end{aligned} \qquad (3)$$

By integrating the right hand side of equation 3, co-ordinates of any point on a clothoid can be calculated as given in equations 4, 5 and 6[5]:

$$\overrightarrow{r(s)} = x(s)\hat{\mathbf{i}} + y(s)\hat{\mathbf{j}} \qquad (4)$$

$$x(s) = x_0 + \int_0^s \cos(\theta_0 + \kappa_0 t + \frac{1}{2}ct^2)\,\mathrm{d}t \qquad (5)$$

$$y(s) = y_0 + \int_0^s \sin(\theta_0 + \kappa_0 t + \frac{1}{2}ct^2)\,\mathrm{d}t \qquad (6)$$

In these equations, $\overrightarrow{r(s)}$ is the position vector to the end point (or any point corresponding to arc length equal to s) and (x_0, y_0) are the coordinates of the start point of the clothoid segment. The definition below borrowed from [4] is used to simplify the descriptions in the rest of this paper.

Definition 1. *A posture like $P(x, y, \kappa, \theta)$ is a collection of four parameters which specify the position (x, y), curvature (κ) and tangent angle (θ) at a particular point on a shape.*

Whenever the start posture of a clothoid is given as $(P_0 = (x_0, y_0, \kappa_0, \theta_0))$, and its arc length (s) and sharpness (c) are also given, the end posture of the clothoid $(P_e = (x_e, y_e, \kappa_e, \theta_e))$ can be calculated using equations 1, 2, 5 and 6.

5. Biclothoids

Definition 2. *A biclothoid is a curve consisting of two back to back connected clothoids where the second clothoid's sharpness is the opposite of the sharpness of the first one.*

Equations 7 and 8 reflect definition 2 above:

$$c_1 = -c_2 \qquad (7)$$
$$s_1 + s_2 = s_{BC} \qquad (8)$$

The biclothoid is consisted of two clothoids, namely clothoid 1 and clothoid 2. In these equations, c_1 and c_2 are the sharpness values and s_1 and s_2 are the arc lengths of clothoids 1 and 2 respectively. Also s_{BC} stands for the arc length of the biclothoid. Using equations 1 and 2 for each of the back to back connected clothoids that constitute a biclothoid and combining them with 7, equations 9 and 10 below can be extracted for the curvature and tangent angle at the end of a biclothoid.

$$\kappa_e = \kappa_0 + c_1(s_1 - s_2) \qquad (9)$$
$$\theta_e = \theta_0 + \kappa_0(s_1 + s_2) + \frac{1}{2}c_1(s_1{}^2 - s_2{}^2 + 2s_1 s_2) \qquad (10)$$

Combining equations 7 to 10 and solving for s_1, s_2, c_1 and c_2 gives the equations below.

$$\Delta\theta = \theta_e - \theta_0$$
$$SQ = \Delta\theta^2 - \Delta\theta s_{BC}(\kappa_0 + \kappa_e) + \frac{s_{BC}^2}{2}(\kappa_0{}^2 + \kappa_e{}^2)$$
$$s_1 = \frac{\Delta\theta - \kappa_e s_{BC} - \sqrt{SQ}}{\kappa_0 - \kappa_e} \qquad (11)$$
$$s_2 = s_{BC} - s_1 \qquad (12)$$
$$c_1 = \frac{\kappa_e - \kappa_0}{s_1 - s_2} \qquad (13)$$
$$c_2 = -c_1 \qquad (14)$$

According to equations 11, 12, 13 and 14, a biclothoid can be fully determined if its total length (s_{BC}), start posture $(P_0 = (x_0, y_0, \kappa_0, \theta_0))$ and curvature and angle at its end point (κ_e, θ_e) are given. After determining the biclothoid, coordinates of its end point can be calculated easily by applying equations 5 and 6 to each of the back to back clothoids.

As an example, assume that $x_0 = 0$, $y_0 = 0$, $\kappa_0 = 0.1$, $\theta_0 = \frac{\pi}{4}$, are given as the start posture of the biclothoid and $\kappa_e = 0.2$, $\theta_e = \frac{3\pi}{4}$ are given for the curvature and angle at the end of the biclothoid, and the length of the biclothoid is set to $s_{BC} = 10mm$. Using equations 11, 12, 13 and 14 will give $c_1 = 0.012\frac{1}{mm^2}$, $c_2 = -0.012\frac{1}{mm^2}$, $s_1 = 9.34mm$ and $s_2 = 0.66mm$. Consequently equations 5 and 6 will give $(0.88, 8.95)$ as the end point of the biclothoid.

6. Fitting Biclothoids

The problem that this paper is trying to solve is to fit a biclothoid between two successive shapes like $Q_1(t)$ and $Q_2(t)$ where $(0 <= t <= 1)$. It is assumed that Q_1 and Q_2

Algorithm 1: The algorithm used for finding a biclothoid that starts at $Q_1(t_1{}^*)$, ends at a point on Q_2 and has G2-continuity with both shapes.

> **Input** : $(Q_1, Q_2, t_1{}^*)$: Q_1 and Q_2 define the first and the second shapes respectively. These shapes can be either two lines, two arcs, or an arc and a line, $t_1{}^*$: Specifies a point on Q_1 where the biclothoid starts.
>
> **Output:** (c_1, c_2, s_1, s_2): The biclothoid fillet that starts at $Q_1(t_1{}^*)$.

1 **Function** BICLOTHOID $(Q_1, Q_2, t_1{}^*)$

2 $\quad$ $\epsilon \leftarrow 1e - 5$

3 $\quad$ $(x_0{}^*, y_0{}^*, \kappa_0{}^*, \theta_0{}^*) \leftarrow Q_1(t_1{}^*)$

4 $\quad$ $s_{BC} \leftarrow$ Length of Q_2

5 $\quad$ $t_2 \leftarrow (1.0 - t_1{}^*) \times (\text{Length of } Q_1)/(\text{Length of } Q_2)$

6 $\quad$ **repeat**

7 $\quad\quad$ $(x_e{}^*, y_e{}^*, \kappa_e{}^*, \theta_e{}^*) \leftarrow Q_2(t_2)$

8 $\quad\quad$ Substitute $x_0{}^*, y_0{}^*, \kappa_0{}^*, \theta_0{}^*, \kappa_e{}^*, \theta_e{}^*$ and s_{BC} in equations 11, 12, 13 and 14 and calculate s_1, s_2, c_1 and c_2

9 $\quad\quad$ Calculate x_e and y_e using equations 5 and 6

10 $\quad\quad$ $f_1 \leftarrow x_e - x_e{}^*$

11 $\quad\quad$ $f_2 \leftarrow y_e - y_e{}^*$

12 $\quad\quad$ Repeat steps 7 to 11 using $s_{BC} + \epsilon$ and $t_2 + \epsilon$ and calculate partial derivatives:

13 $\quad\quad\quad$ $\frac{\partial f_1}{\partial s_{BC}} \leftarrow \frac{f_1(s_{BC}+\epsilon, t_2) - f_1(s_{BC}, t_2)}{\epsilon}$

14 $\quad\quad\quad$ $\frac{\partial f_2}{\partial s_{BC}} \leftarrow \frac{f_2(s_{BC}+\epsilon, t_2) - f_2(s_{BC}, t_2)}{\epsilon}$

15 $\quad\quad\quad$ $\frac{\partial f_1}{\partial t_2} \leftarrow \frac{f_1(s_{BC}, t_2+\epsilon) - f_1(s_{BC}, t_2)}{\epsilon}$

16 $\quad\quad\quad$ $\frac{\partial f_2}{\partial t_2} \leftarrow \frac{f_2(s_{BC}, t_2+\epsilon) - f_2(s_{BC}, t_2)}{\epsilon}$

17 $\quad\quad$ $J \leftarrow \begin{bmatrix} \frac{\partial f_1}{\partial s_{BC}} & \frac{\partial f_1}{\partial t_2} \\ \frac{\partial f_2}{\partial s_{BC}} & \frac{\partial f_2}{\partial t_2} \end{bmatrix}$

18 $\quad\quad$ $\begin{bmatrix} s_{BC} \\ t_2 \end{bmatrix} \leftarrow J^{-1} \begin{bmatrix} f_1(s_{BC}, t_2) \\ f_2(s_{BC}, t_2) \end{bmatrix}$

19 $\quad$ **until** $Abs(f_1) < \epsilon$ *and* $Abs(f_2) < \epsilon$

20 $\quad$ return (c_1, c_2, s_1, s_2)

21 **end**

are connected with only G0-continuity or G1-continuity. Therefore, there is a tangent discontinuity or a curvature discontinuity at their junction point. Q_1 and Q_2 can be either both lines, or both arcs or one arc and one line.

The start point of the biclothoid will be on Q_1 and its end point will reside on Q_2. The biclothoid has G2-continuity with both Q_1 and Q_2 which means that the tangent angle and the curvature of the biclothoid at its start point are equal to the tangent angle and curvature of Q_1. The same is true for the end point of the biclothoid where it joins Q_2 with G2-continuity. Without loss of generality it is assumed that the arc length of Q_2 is greater than or equal to the arc length of Q_1 and the curvature of Q_2 is smaller than or equal to the curvature of Q_1. These

two assumptions do not add any restrictions to the problem: if Q_1 is longer than Q_2, then part of Q_1 can be cut away to make the lengths equal, and if Q_2 has a greater curvature than Q_1, the two shapes can be simply swapped.

In the case where both shapes are line segments, analytical solutions for fitting biclothoids exist. An analytic algorithm for fitting biclothoids when both Q_1 and Q_2 are line segments is proposed in [5]. However, when one or both shapes are arcs, a different method is needed. Algorithm 1 is used to fit a biclothoid between a given point on Q_1 and Q_2. The algorithm uses Newton method to first find the length of the biclothoid (s_{BC}) and the parameter value (t_2) for Q_2 at the point where the biclothoid and Q_2 join. It then uses the calculated values for s_{BC} and t_2 to determine other parameters of the biclothoid. For example BICLOTHOID(Q_1, Q_2, 0.5) returns the biclothoid that starts at the middle of Q_1 and joins Q_1 and Q_2 with curvature continuity.

7. Limiting the Deviation

In this section the method used for measuring and limiting the deviation between the fitted curve and the original toolpath is explained.

As mentioned earlier, in a CNC application, having a high accuracy is crucial. Fitting biclothoids at discontinuity points results in a new toolpath which will have some distance with the original toolpath. In order to satisfy the accuracy requirements, the distance between each of the fitted biclothoids and the original toolpath has to be limited to a specified tolerance. The first step in limiting this deviation is measuring it. Next subsection explains the method used for measuring the deviation.

7.1. Measuring the Deviation

The maximum deviation of the fitted biclothoid from the original path can be defined as directed Hausdorff distance. Directed Hausdorff distance from curve U to curve V is defined in equation 15 below [38]. Hausdorff distance is the maximum distance between any point on the curve U from its closest point on curve V.

$$H(U,V) = \max_{\forall A \in U} \min_{\forall B \in V} ||A, B|| \qquad (15)$$

In [5], it has been proved that when both shapes are lines, the fillet will be symmetrical in respect to the bisector of the corner. In this case, the distance between the two paths can be easily calculated by finding the distance between the corner and the middle point of the fillet[5]. However when one or both of the shapes are arcs, in order to calculate the accurate Hausdorff distance, two nested numerical solvers should be used. In figure 5, for each point like A_1 on the original toolpath (curve U), the closest point on the modified toolpath (V), is a point like B_1. This point can be found using a numerical method like false position minimisation algorithm. This minimum distance is represented in equation 15 as $\min_{\forall B \in V} ||A, B||$. For every point on U, the minimum distance with V can be calculated as explained. Figure 6, shows these minimum values for 100 points along curve U in figure 5. Maximum deviation between U and V is the largest minimum distance in figure 6. False position maximisation algorithm or any other numerical solver can
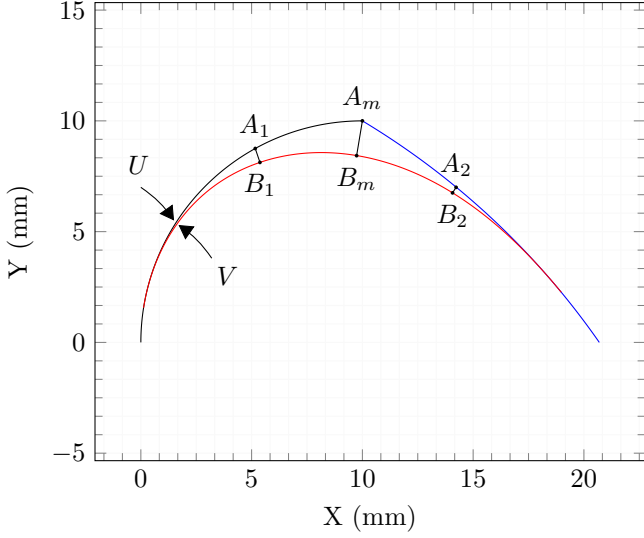
Figure 5: Measuring the distance between the original toolpath (U: black and blue arcs) and the fitted biclothoid (V: red curve). A_1 and A_2 are arbitrary points on U and B_1 and B_2 are the closest points to them which reside on V. Maximum distance takes place at A_m which is $1.59mm$ away from B_m. B_m is the closest point on V to A_m.
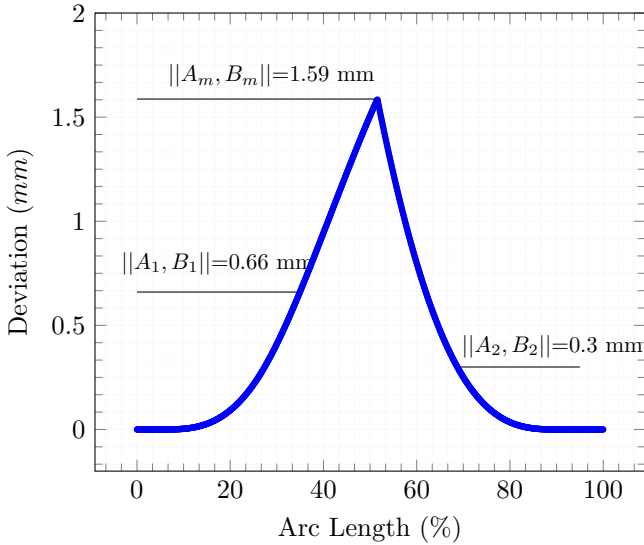

Figure 6: Deviation between the original toolpath and the biclothoid in figure 5. A_1, B_1, A_2, B_2, A_m and B_m correspond to the same points in figure 5. The horizontal axis, shows the arc length as a percentage of the total length of the original toolpath and the vertical axis shows the deviation.

again be used to find the maximum. In figure 5, the maximum distance happens at point A_m on U. The closest point on V to A_m is B_m and $||A_m, B_m||$ which is the maximum deviation between U and V, equals $1.59\ mm$.

7.2. Decreasing the deviation

If it turns out that the distance between the fitted biclothoid and the original toolpath is greater than the specified
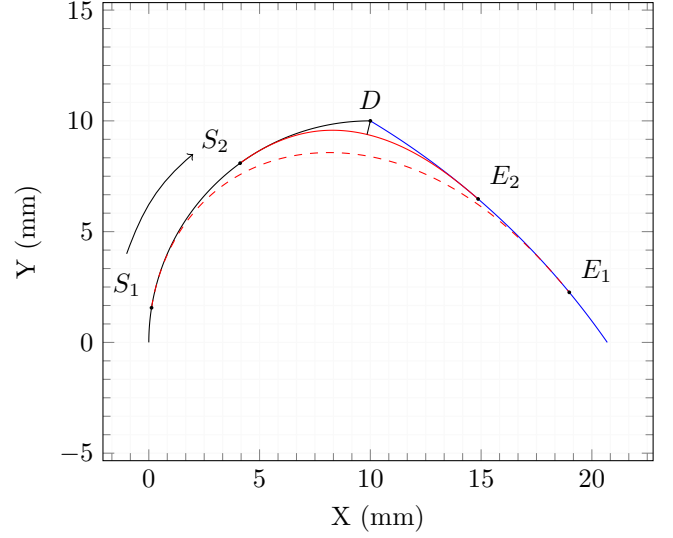

Figure 7: Limiting the deviation. By moving the start point of the biclothoid from S_1 to S_2, maximum distance between the two paths decreases from $1.59mm$ to $0.63mm$.

tolerance, in order to maintain the cutting accuracy, the distance has to be reduced. In order to reduce the distance, it is sufficient to move the start point of the biclothoid closer to the point of discontinuity on the original toolpath. Figure 7, shows an example. The point of discontinuity in the original toolpath is marked as D. When the biclothoid starts at point S_1, the maximum deviation is $1.59\ mm$. When the start point of the biclothoid moves to S_2 which is closer to D, maximum deviation decreases to $0.63\ mm$.

Again if both shapes are lines, analytical methods from [5] can be used to easily find a solution which satisfies tolerance requirements. However for the cases that one of the shapes is an arc, algorithm 2 is used to limit the deviation. This algorithm uses false position method to find the start point of the biclothoid so that the deviation is limited to the given tolerance. In practice, in most of the cases the deviation is so small that no re-fitting is required.

7.3. Robustness and Speed

In order to assess the robustness and speed of the proposed method, an extensive test was carried out. For line to line corners, angles from 0.00001 degrees to 150.00000 degrees with steps of 0.000001 degrees were examined. Also a wide range of line to arc and arc to arc cases, with arc radii ranging from 0.1mm to 1000mm and angles ranging from 0 to 150 degrees were tested. Overall, more than 10 million different cases were tested and the algorithm converged in all of the tests. Six of these cases are portrayed in figure 8 to show some examples of the different cases that have been examined.

Using a computer with a dual-core CPU running at a clock rate of 2.4 GHz, the average fitting time for line to line corners was only 5 microseconds. For line to arc and arc to arc cases the average fitting time was 42 microseconds. Although the fillet fitting time for arc to arc transitions is longer than the simpler case of line to line transitions, it is still small enough for using in real time applications.

8

Algorithm 2: The algorithm used for finding a biclothoid that starts at a point on Q_1, ends at a point on Q_2 and has G2-continuity with both shapes so that the deviation between the biclothoid and the toolpath composed of Q_1 and Q_2 is smaller than *tol*.

Input : (Q_1, Q_2, tol): Q_1 and Q_2 define the first shape and the second shape respectively. These shapes can be either two lines, two arcs, or an arc and a line, *tol*: The maximum permissible distance between the fitted biclothoid and the original toolpath consisted of Q_1 and Q_2.

Output: (c_1, c_2, s_1, s_2): The biclothoid fillet with limited deviation from the original toolpath.

1 Function BICLOTHOIDWITHTOLERANCE (Q_1, Q_2, tol)
2 $(c_1, c_2, s_1, s_2) \leftarrow$ BICLOTHOID$(Q_1, Q_2, 0.0)$
3 $d \leftarrow$ distance between Q_1, Q_2 and the biclothoid (c_1, c_2, s_1, s_2)
4   **if** $(d < tol)$ **then**
5   return (c_1, c_2, s_1, s_2)
6   **end**
7 $\epsilon \leftarrow 0.01 \times tol$
8 $t_1 \leftarrow 0.0$
9 $t_2 \leftarrow 1.0$
10 $y_1 \leftarrow d - tol$
11 $y_2 \leftarrow 0.0$
12   **repeat**
13 $t_3 = \frac{t_1 y_2 - t_2 y_1}{y_2 - y_1}$
14 $(c_1, c_2, s_1, s_2) \leftarrow$ BICLOTHOID(Q_1, Q_2, t_3)
15 $d \leftarrow$ distance between Q_1, Q_2 and the biclothoid (c_1, c_2, s_1, s_2)
16 $y_3 \leftarrow d - tol$
17   **if** $(y_3 \times y_1 > 0)$ **then**
18 $y_1 = y_3$
19 $t_1 = t_3$
20   **else**
21 $y_2 = y_3$
22 $t_2 = t_3$
23   **end**
24   **until** $(y_3 < \epsilon)$
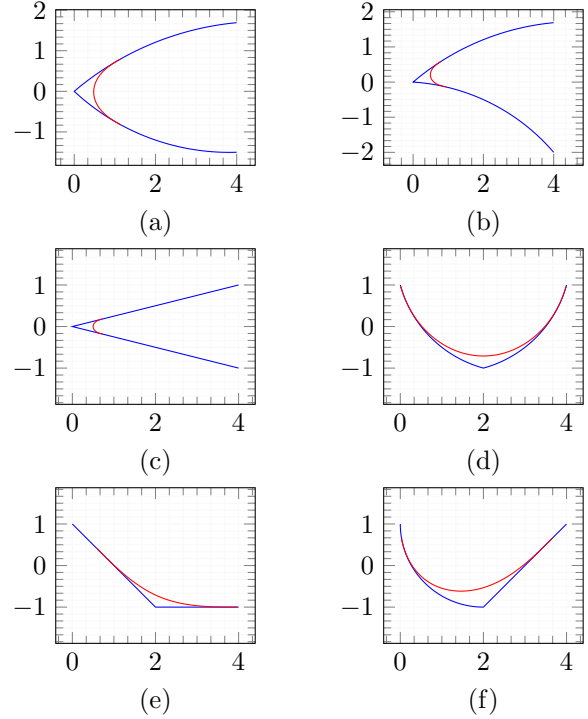**25**   return (c_1, c_2, s_1, s_2)
26 end

Figure 8: Examples of biclothoid fillets. In all 6 cases, the deviation between the original toolpath and the fillet is limited to 0.5 mm. (a) Arc to arc with acute angle (b) Arc to arc with acute angle and change of curvature sign (c) Line to line with acute angle (d) Arc to arc with obtuse angle (e) Line to line with obtuse angle (f) Arc to line with obtuse angle.

For fitting time measurements, an ordinary desktop computer was used. Nevertheless, new generations of CNC controllers are equipped with powerful processors on par with the one that was used for the measurements. For example, Sinumerik 840D manufactured by Siemens, takes advantage of a multi-core CPU with a clock rate up to 2.4 GHz [39]. As another example, the CNC controller that was used for machining trials reported in section 9, has a quad-core CPU with a clock frequency of 2.3 GHz [36]. It should also be noted that the fitting algorithm needs to run only once for each discontinuity. Therefore, the computational burden of the smoothing algorithm will be minimal.

8. Maximum Velocity On Biclothoid Fillets

In this section, the maximum permissible velocity when moving on a clothoid will be calculated. It should be noted that moving on any path which is not linear, like a circle or a clothoid, requires non-zero acceleration and jerk to be applied by the axes that are participating in the move. Assuming constant tangential velocity (path velocity), in the case of a circle, it can be proved that the maximum acceleration and jerk of each of the participating axes, can be calculated from equations 16 and 17 respectively.

$$a_{max} = \frac{v^2}{r} = \kappa \times v^2 \tag{16}$$

$$j_{max} = \frac{v^3}{r^2} = \kappa^2 \times v^3 \tag{17}$$

In these equations, v is the path velocity, $\kappa = \frac{1}{r}$ denotes the curvature and a_{max} and j_{max}, show the maximum acceleration and jerk that will be generated on each of the participating axes respectively. These equations show that even when the path velocity (v) is fixed, the motors which are moving the machine, will have to apply potentially large acceleration and jerk values in order to move the machine on a circle. Since the maximum acceleration and jerk that a motor can generate are limited, path velocity should be limited accordingly. Equations 18 and 19 can be used to calculate the maximum velocity for a full circle.

$$v_1 = \sqrt{a_{lim} \times r} = \sqrt{\frac{a_{lim}}{\kappa}} \tag{18}$$

$$v_2 = \sqrt[3]{j_{lim} \times r^2} = \sqrt[3]{\frac{j_{lim}}{\kappa^2}} \tag{19}$$

The maximum path velocity that can be applied will be $\min(v_1, v_2)$ where v_1 and v_2 are calculated using equations 18 and 19 respectively. In these equations, a_{lim} is the acceleration limit and j_{lim} denotes the jerk limit. In the rest of this section, similar equations will be derived for clothoids. Since a circular arc is a special case of a clothoid, the equations presented above will be proved in the process as well.

Starting from equation 5, the velocity of the X axis can be derived as equation 20.

$$\begin{aligned} v_x(s) &= \frac{dx(s)}{dt} = \frac{ds}{dt} \times \frac{dx(s)}{ds} \\ &= v \times cos(\theta_0 + \kappa_0 s + \frac{1}{2}cs^2) \end{aligned} \tag{20}$$

In this equation, $v_x(s)$ denotes the velocity of the X axis, v is the path velocity, and θ_0, κ_0, c and s have the same meaning as in equation 5.

Assuming constant tangential velocity, taking the derivative of equation 20, will give the acceleration of the X axis in equation 21.

$$\begin{aligned} a_x(s) &= \frac{dv_x(s)}{dt} = \frac{ds}{dt} \times \frac{dv_x(s)}{ds} \\ &= -v^2 \times (cs + \kappa_0) \times sin(\theta_0 + \kappa_0 s + \frac{1}{2}cs^2) \end{aligned} \tag{21}$$

Finally taking the derivative once again, gives equation 22 for the jerk of the X axis when moving on a clothoid.

$$\begin{aligned} j_x(s) &= \frac{da_x(s)}{dt} = \frac{ds}{dt} \times \frac{da_x(s)}{ds} \\ &= -v^3 \times c \times sin(\theta_0 + \kappa_0 s + \frac{1}{2}cs^2) \\ &\quad -v^3 \times (cs + \kappa_0)^2 \times cos(\theta_0 + \kappa_0 s + \frac{1}{2}cs^2) \end{aligned} \tag{22}$$

Similar equations can be extracted for the Y axis. The equation for jerk in 22 has two components. The first component $(-v^3 \times c \times sin(\theta_0 + \kappa_0 s + \frac{1}{2}cs^2))$ makes a right angle with the unit tangent vector and is known as centripetal jerk [40, 41]. It is evident that this component is directly proportional to the derivative of curvature (sharpness) denoted by c in the equation.

In equations 21 and 22, The maximum values for a_x and j_x depend on several factors. In order to simplify the equations

and derive a limit that works in all situations, a clothoid which covers 360 degrees (like a full circle but with potentially variable radius), and its curvature varies from κ_0 to $\kappa_e = cs + \kappa_0$ with a maximum curvature of $\kappa_{max} = \max(\kappa_0, \kappa_e)$ can be assumed. Using this assumption, the maximum of the sine and cosine terms will become 1. In this case, maximum acceleration and jerk values can be calculated using equations 23 and 24 respectively.

$$a_{max} = \kappa_{max} \times v^2 \tag{23}$$

$$j_{max} = \sqrt{c^2 + \kappa_{max}{}^4} \times v^3 \tag{24}$$

Finally, the maximum path velocity that can be applied to ensure that the acceleration and jerk limits are not exceeded can be calculated from equations 25 to 27.

$$v_1 = \sqrt{\frac{a_{lim}}{\kappa_{max}}} \tag{25}$$

$$v_2 = \sqrt[3]{\frac{j_{lim}}{\sqrt{c^2 + \kappa_{max}{}^4}}} \tag{26}$$

$$v = \min(v_1, v_2) \tag{27}$$

It is evident that equations 18 and 19 are special cases of equations 25 and 26 when $c = 0$ and $\kappa_{max} = \kappa$. This reflects the fact that circular arcs are special cases of clothoids.

9. Case Study

In this section, the example given in section 3 is processed and three biclothoids are fitted to eliminate the discontinuities. The smoothed toolpath is plotted in figure 9. The introduced deviation is very small (100 microns) and consequently it is not easy to distinguish between the original toolpath and the smoothed toolpath in the plot. To show the deviation, biclothoids and transition points are portrayed in three magnified boxes in figure 9. In order to better highlight how clothoids act as transition curves between two arcs and between a line and an arc, the curvature profile of the smoothed toolpath is also plotted in figure 10. In figure 10, the clothoids are characterised by their linear curvature profiles, while the arcs can be distinguished by their constant curvature.

Table 1 shows the data for the three fitted biclothoids. The maximum feedrate value in the table is calculated using equations 25 to 27 assuming $j_{max} = 200000\frac{mm}{s^3}$ and $a_{max} = 9800\frac{mm}{s^2}$ for jerk and acceleration limits respectively. As long as the feedrate is limited to these values, the acceleration and jerk when the biclothoid move is being processed, will not exceed the limits.

The smoothed toolpath was given to an ANCA Motion AMC5 CNC controller [36] on a laser cutting machine (figure 11). Figure 12 shows the path velocity. Unlike the non-smoothed case in figure 4, the machine can maintain a high feedrate when cutting the smoothed toolpath.

Figure 13 showing the acceleration of the X and Y axes exhibits no step changes in acceleration. This is different from figure 2 where sudden changes in the acceleration are visible. Similarly looking at the jerk profile in 14, no jerk spikes can be seen.

Two pictures of the cut results are given in figure 15. The part on the top was cut without using the smoothing fillets,
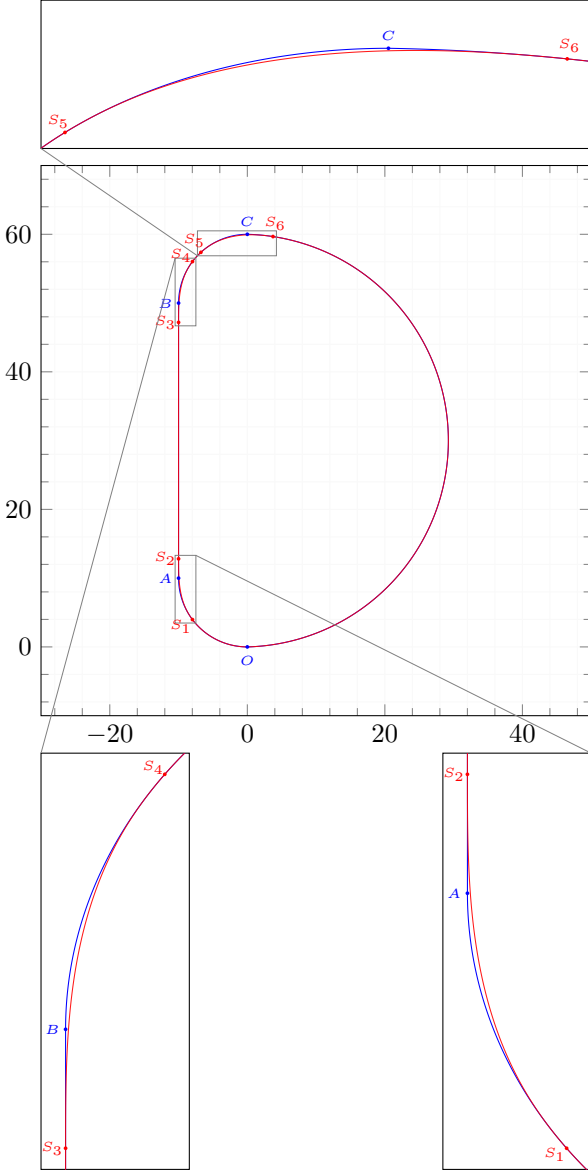
Figure 9: Smoothed toolpath (red curve) versus the original toolpath (blue curve). Discontinuities at points A, B and C are smoothed by biclothoids S_1S_2, S_3S_4 and S_5S_6 respectively. These three biclothoids are magnified in the figure and their specifications are given in table 1.

Table 1: Biclothoids specifications. For maximum feedrate calculations, $j_{max} = 200000\frac{mm}{s^3}$ and $a_{max} = 9800\frac{mm}{s^2}$ were used as the jerk and acceleration limits respectively.

Biclothoid	Sharpness	Maximum Curvature	Maximum Feedrate
	(c)	(κ_{max})	(f_{max})
S_1S_2	$0.016 \ \frac{1}{mm^2}$	$0.124 \ \frac{1}{mm}$	$12486 \ \frac{mm}{min}$
S_3S_4	$0.016 \ \frac{1}{mm^2}$	$0.124 \ \frac{1}{mm}$	$12486 \ \frac{mm}{min}$
S_5S_6	$0.009 \ \frac{1}{mm^2}$	$0.116 \ \frac{1}{mm}$	$13871 \ \frac{mm}{min}$

while the part at the bottom was cut using the proposed bi-
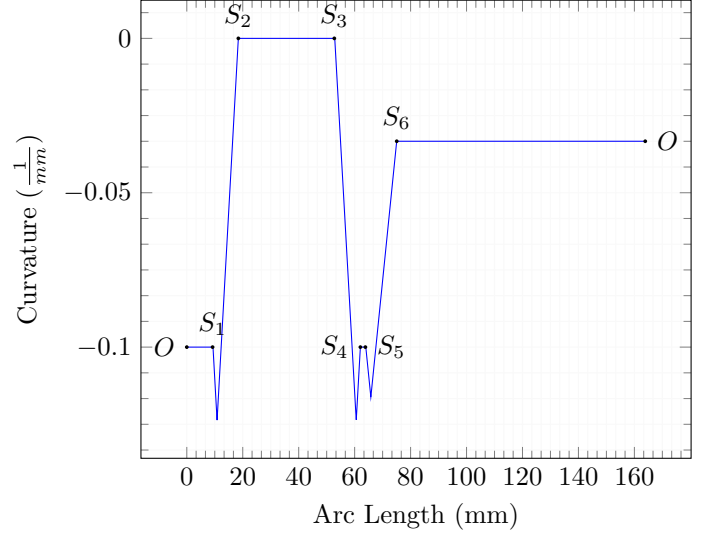
Figure 10: The curvature profile of the smoothed toolpath. The points marked on the plot correspond to the points with the same name in figure 9.
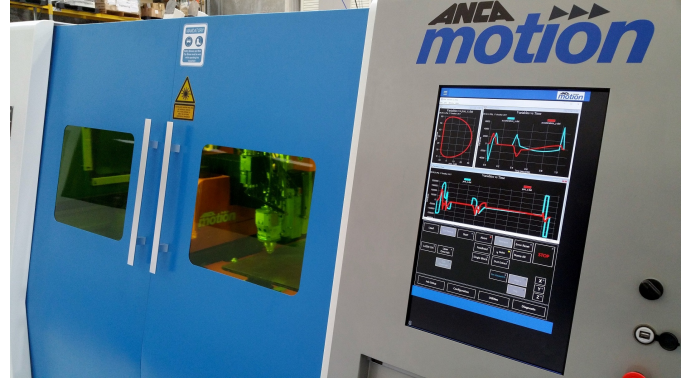

Figure 11: The CNC machine used for evaluating the proposed method.

clothoid fillets. In the top picture from figure 15, two burning marks are visible. These two points are at the same position as points B and C in figure 1. As explained before, without using the smoothing fillets, machine has to slow down at these two points (figure 4 shows that the feedrate drops at these two points) and this results in a longer exposure to laser and consequently in higher temperatures and burning. As the picture at the bottom of figure 15 shows, there will be no burning marks when the proposed fillets are used. The reason is that since there are no curvature or tangency discontinuities in the smoothed toolpath, the machine can cut the part without slowing down at any point. The experiment proves that using the proposed method improves both the cycle time and the quality of cut.

10. Comparison With Bezier Fillets

Using Bezier fillets for path smoothing in CNC applications is proposed in several works [6, 15, 16, 17]. Sencer et al. in [6], developed an algorithm to generate optimised Bezier curves in order to achieve the highest feedrate while adhering to motion
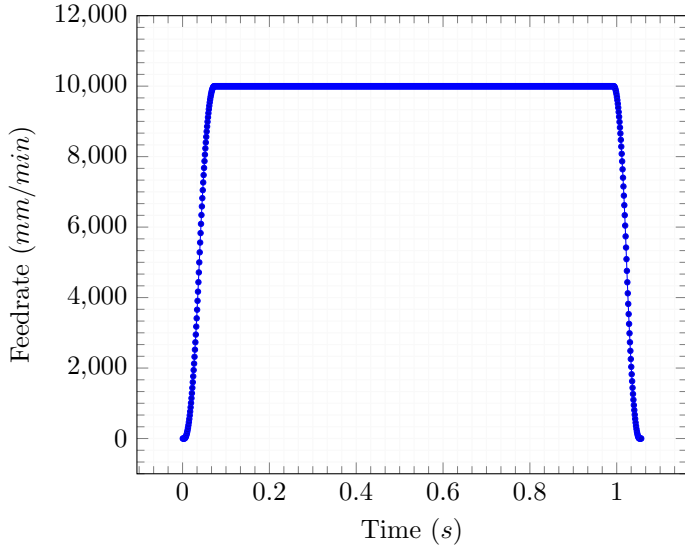
Figure 12: Path velocity when moving on the smoothed toolpath given in figure 9. The machine does not slow down at transitions and can maintain a high speed.
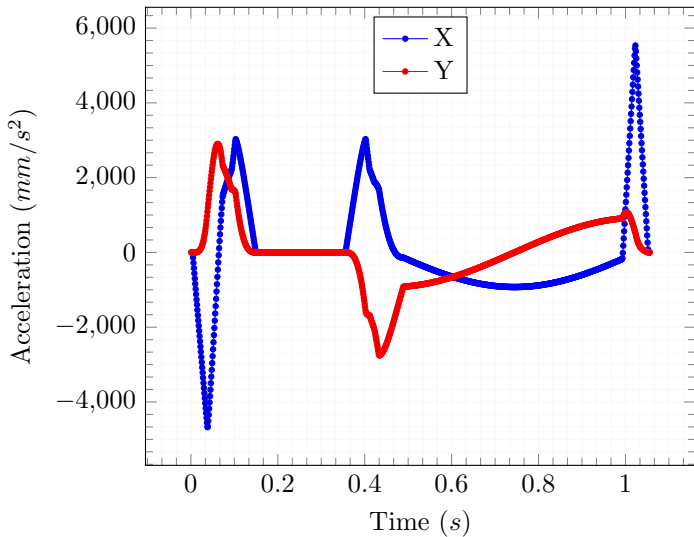

Figure 14: Jerk of the X and Y axis when moving on the toolpath given in figure 9. Jerk is limited within the specified values.

Figure 13: Acceleration of the X and Y axes when moving on the smoothed toolpath given in figure 9. There are no step changes in the acceleration.

Table 2: Comparison between a biclothoid fillet and a Bezier fillet for the same corner (figure 16). For maximum feedrate calculations, $j_{max} = 200000 \frac{mm}{s^3}$ and $a_{max} = 9800 \frac{mm}{s^2}$ were used as the jerk and acceleration limits respectively.

Fillet Type	Length	Maximum Curvature	Maximum Sharpness	Maximum Feedrate
Bi-clothoid	2.22 mm	0.47 $\frac{1}{mm}$	0.42 $\frac{1}{mm^2}$	4489 $\frac{mm}{min}$
Bezier	1.71 mm	0.36 $\frac{1}{mm}$	2.09 $\frac{1}{mm^2}$	2744 $\frac{mm}{min}$

in figure 16. The two fillets look very similar and actually cannot be distinguished in the figure. However their curvature and sharpness profiles, plotted in figure 17 look completely different.

Some of the differences between the two fillets, identified by analysing figures 16 and 17 are summarised in table 2. According to the table, the length of the biclothoid fillet is 2.22 mm, while the Bezier fillet is shorter and is only 1.71 mm long. Maximum curvature for the biclothoid fillet is also higher than the maximum curvature of the Bezier fillet (0.47 $\frac{1}{mm}$ for the biclothoid and 0.36 $\frac{1}{mm}$ for the Bezier fillet). However, the maximum derivative of curvature (plotted at the bottom of figure 17) is much smaller for the biclothoid fillet (Maximum sharpness is 0.42 $\frac{1}{mm^2}$ for the biclothoid fillet and 2.09 $\frac{1}{mm^2}$ for the Bezier curve).

Using the curvature profile and equations 25 to 27 from section 8, the maximum permissible feedrate for each fillet can be calculated. The calculated values are given in table 2. The maximum feedrate that can be applied on the biclothoid fillet without exceeding acceleration and jerk limits is 4489 $\frac{mm}{min}$ while the highest permissible feedrate on the Bezier fillet is only 2744 $\frac{mm}{min}$. Taking the length difference between the two curves into account, for this particular line to line transition, using a biclothoid fillet instead of a Bezier fillet can improve the cycle time by 20%.

limits. In this section, biclothoid fillets are compared with the optimised Bezier curves proposed in [6] to see which method gives higher feedrates and shorter cycle times. Two sample toolpaths, one simple and one more complex, will be analysed in this section.

10.1. Case 1

The method proposed in [6] can only be applied to line to line transitions. Therefore, for the first comparison case, a simple toolpath consisted of two linear moves making a 150 degree angle at their intersection is considered. The original toolpath, the biclothoid fillet and the optimised Bezier curve are plotted
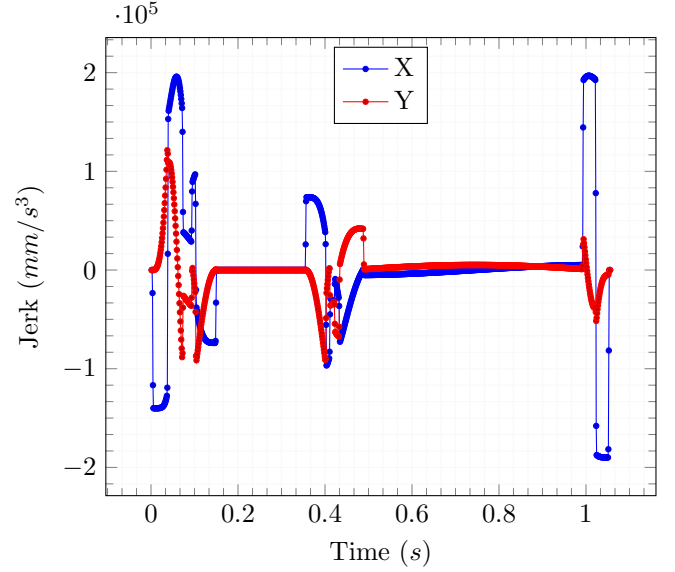
(a) The above part was cut without using smoothing fillets.

(b) The same part was cut using the proposed method for smoothing the corners.

Figure 15: (a) Cut result without using biclothoid fillets: burning marks are created at the points that the machine has to slow down. (b) Cut result using the proposed method: the cut quality has improved.
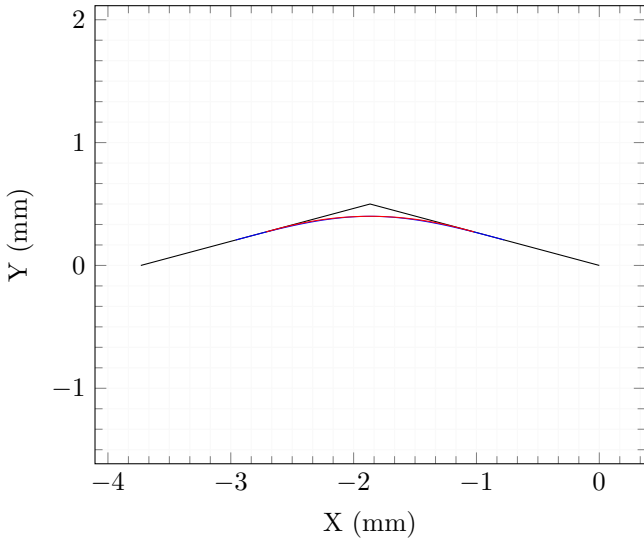

Figure 16: A simple toolpath consisted of two lines which make a 150° angle (black), a biclothoid fillet (blue) and a Bezier fillet (red) are plotted. The two fillets look very similar. However their curvature and sharpness profiles in figure 17 are completely different.

10.2. Case 2

For the second comparison case, the toolpath from figure 1 is smoothed using Bezier fillets and the results are compared with the same toolpath smoothed with biclothoid fillets. Since Bezier fillets can only be used for line to line transitions, as the
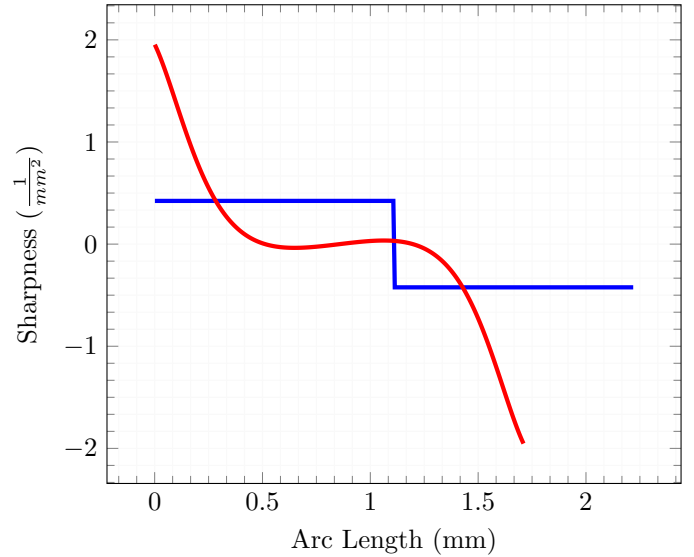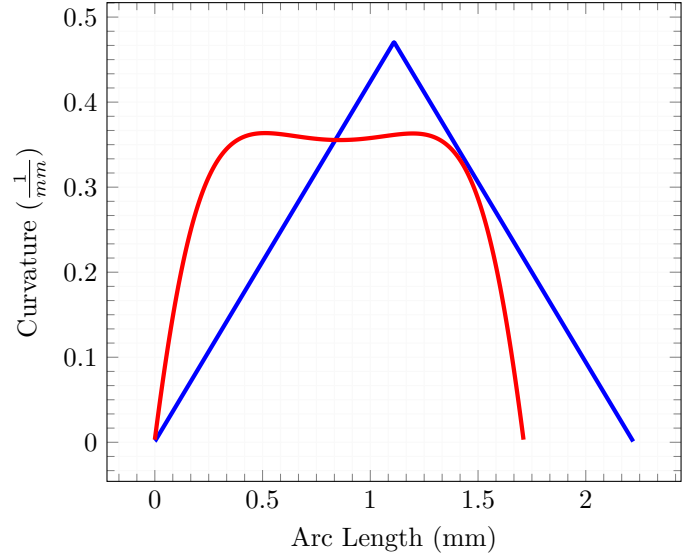

Figure 17: (Top) Curvature profiles of the biclothoid fillet (blue) and the Bezier fillet (red) in figure 16. (Bottom) Sharpness profile of the biclothoid fillet (blue) and the Bezier fillet (red) in figure 16.

first step, the given toolpath has to be converted to a series of line segments. In order to approximate the toolpath using line segments, subdivision algorithm from [42] is used. The number of lines for approximating the toolpath depends on the acceptable error tolerance. In order to achieve a 100 micron tolerance, the three arcs in figure 1 have to be replaced with at least 31 line segments.

After converting the original toolpath to a purely linear toolpath, the number of discontinuities increases from 3 to 31. Consequently 31 Bezier fillets are needed to smooth the toolpath. Figure 18 shows the resulting smoothed toolpath. The curvature profile of the smoothed toolpath is given in figure 19 as well.

A careful comparison between figures 18 and 9 shows that biclothoid fillets give a smoother toolpath than Bezier fillets. The difference is more obvious when the corresponding curvature profiles in figures 19 and 10 are compared. Figure 19 shows that there are many fluctuations in the curvature profile when
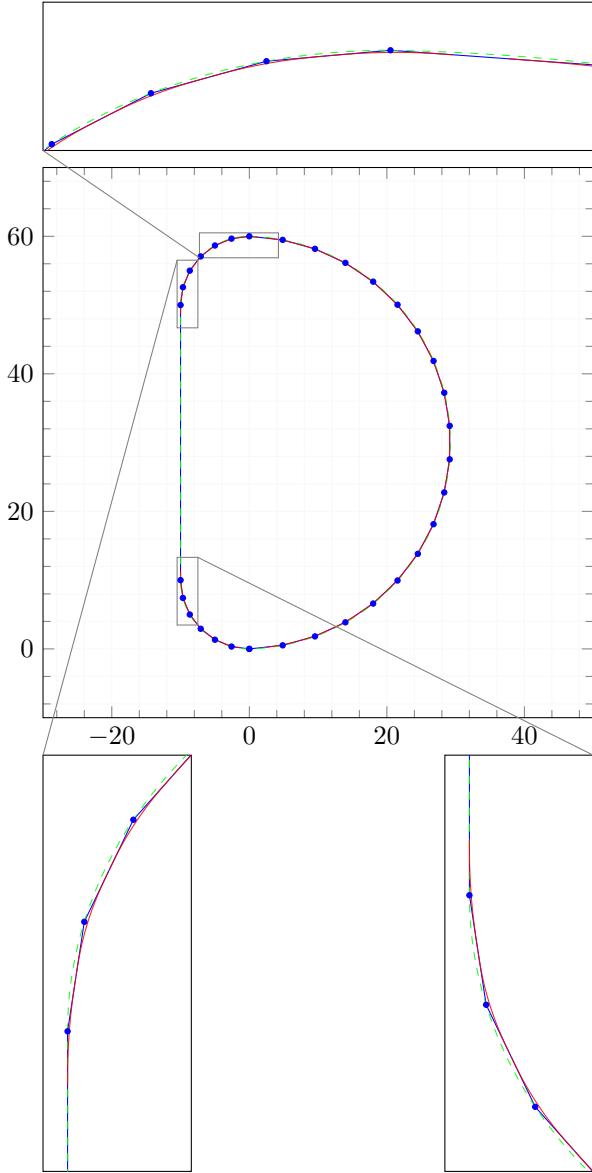
Figure 18: In order to use Bezier fillets, at first the toolpath has to be approximated by line segments. For a 0.1mm tolerance, 31 line segments are needed. Afterwards, 31 Bezier fillets should be fitted to smooth the toolpath. In this figure, the green dashed curve is the original toolpath (from figure 1), the blue curve shows the line segments which approximate the original toolpath and the red curves are the Bezier fillets. Compare this figure with figure 9 which shows the same toolpath smoothed using biclothoid fillets.

Bezier fillets are fitted. The reason is that while only three biclothoid fillets are enough to smooth the discontinuities of the original toolpath, to get the same result, 31 Bezier fillets have to be used. Furthermore, since Bezier fillets can only be fitted between two lines, the curvature has to drop to zero at both ends of the fillet. This is not the case for biclothoid fillets which can be fitted between two arcs, or a line and an arc as well.

One other important difference between the curvature profiles is that according to table 1, the maximum sharpness when using biclothoid fillets is only $0.016 \frac{1}{mm^2}$. In comparison, when
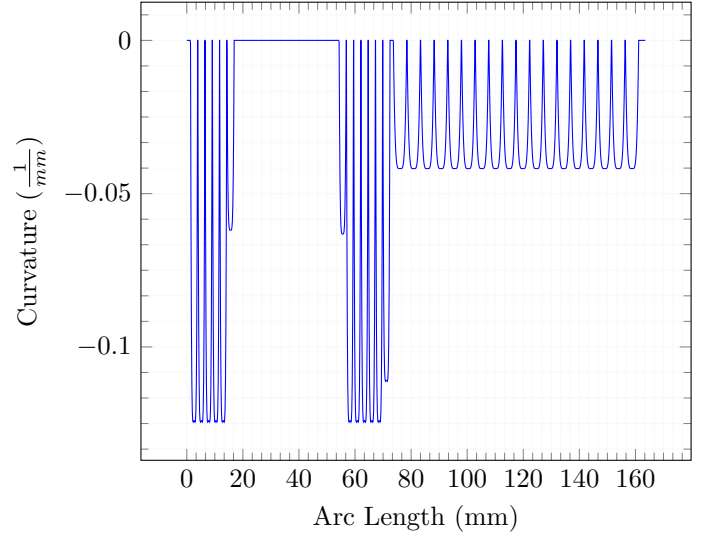

Figure 19: Curvature profile for the smoothed toolpath portrayed in figure 18. Comparing figure 10 with this plot, shows that using biclothoid fillets results in a much smoother curvature profile.
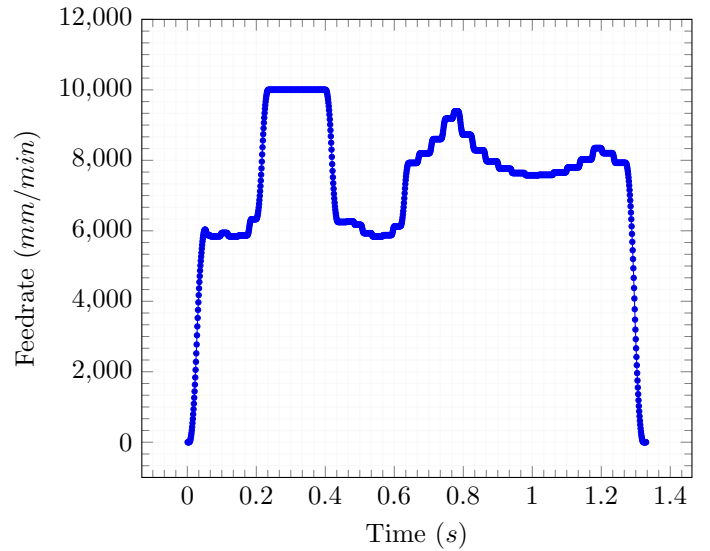

Figure 20: Path velocity when moving on the toolpath given in figure 18. The machine has to slow down on the Bezier fillets and the cycle time increases in comparison to the case where Biclothoid fillets were used (figure 12).

Bezier fillets are used, the maximum derivative of curvature (maximum slope in figure 19) is $0.35 \frac{1}{mm^2}$ which is more than 20 times higher than the maximum sharpness of biclothoid fillets. Higher sharpness values will result in lower maximum feedrates as figure 20 shows.

Figure 20 portrays the feedrate profile when the smoothed toolpath in figure 18 is processed. The plot shows that the machine has to slow down on the fillets in order to adhere to the jerk and acceleration limits. Consequently the cycle time increases to 1.32s, which is 25% longer than the case where biclothoid fillets are used (compare with figure 12).

The difference between biclothoid fillets and Bezier fillets

Table 3: Comparison between biclothoid fillets and Bezier fillets for smoothing the toolpath from figure 1 with two different tolerance values. When the tolerance gets tighter, a larger number of Bezier fillets have to be used and the cycle time increases significantly. Biclothoid fillets on the contrary, show a good performance at smaller tolerance values.

Fillet type	Tolerance	Number of fillets	Cycle time
Biclothoid	$0.1mm$	3	$1.048s$
Biclothoid	$0.01mm$	3	$1.053s$
Bezier	$0.1mm$	31	$1.322s$
Bezier	$0.01mm$	96	$1.723s$

becomes more prominent at tighter tolerance settings. Table 3 compares Bezier and biclothoid fillets when used to smooth the toolpath from figure 1 at two different tolerance settings. When the tolerance is set to 10 microns, 96 Bezier fillets will be needed and the cycle time increases to $1.72s$. For achieving the same tolerance, only three biclothoid fillets will suffice and the cycle time will drop to $1.05s$. In this scenario, the cycle time that Bezier fillets give is 63% longer than the case where biclothoid fillets are used.

11. Conclusion and Future Work

A method for smoothing corners using biclothoid fillets was presented. The proposed method can be used to smooth out discontinuities in curvature and in tangency for line to line, line to arc, and arc to arc transitions. To the authors' knowledge this is the first method that can be applied to arc to arc and line to arc corners in addition to line to line transitions.

For smoothing line to line transitions, analytical solutions have been developed. However for transitions between two arcs or a line and arc, specially if arc length parameterisation is of importance, iterative algorithms have to be used. To evaluate the robustness of the proposed method, the performance of the developed algorithms was extensively tested and the results were reported.

A comparison with Bezier fillets shows that using the proposed method can result in a significantly shorter cycle time with a much smaller number of fillets. It was demonstrated that since Bezier fillets can only be fitted between two lines, the toolpath has to be broken into a large number of lines and consequently a large number of fillets have to be fitted. The proposed method does not have this limitation.

The proposed method has a number of important advantages and satisfies all of the requirements explained in section 3:

a. G2-continuity: The generated toolpath is curvature continuous and eliminates any discontinuities in the original toolpath.

b. Bounded maximum error: the deviation between the smoothed toolpath and the original toolpath can be limited to a specified tolerance to maintain the accuracy of the manufactured parts.

c. Analytical arc length formula: the arc length of the generated fillets is readily available and unlike Bezier

curves, Quintic splines, etc. there is no need to use integration to calculate the arc length.

d. Arc length parameterisation: the generated path is arc length parameterised which makes it easier to interpolate. There will be no feedrate fluctuations when interpolating the path and unlike PH curves, special interpolators are not required.

e. Minimum curvature change: biclothoid fillets have a triangle shaped curvature profile which gives the smallest sharpness for a specified arc length [3].

To demonstrate the effectiveness of the proposed method a CNC controller equipped with a jerk limited interpolator was used. It was verified that without modifying the toolpath, the controller has to slow down the machine at line to arc and arc to arc transitions to keep the jerk within the limits. Using the proposed method, the controller is able to maintain a high feedrate without introducing any jerk spikes. For calculating the feedrate limit not only the curvature but also the derivative of curvature were taken into account.

The proposed method can be extended to achieve higher degrees of continuity like G3-continuity, provided that the CNC controller can generate a jounce limited velocity profile. For G3-continuity, instead of clothoids, cubic spirals [43] may have to be used. Future work could also investigate using clothoids for generating smooth toolpaths for 5 axis CNC applications.

[1] S.-H. Suh, Theory and design of CNC systems, Springer, 2008.

[2] A. Scheuer, T. Fraichard, Planning continuous-curvature paths for car-like robots, in: Proceedings of the 1996 IEEE/RSJ International Conference on Intelligent Robots and Systems '96, IROS 96, Vol. 3, 1996, pp. 1304–1311. doi:10.1109/IROS.1996.568985.

[3] W. Nelson, Continuous-curvature paths for autonomous vehicles, Proceedings, 1989 International Conference on Robotics and Automation (1989) 1260–1264doi:10.1109/ROBOT.1989.100153.

[4] D. H. Shin, S. Singh, Path Generation for Robot Vehicles Using Composite Clothoid Segments, Tech. Rep. CMU-RI-TR-90-31, Robotics Institute, Pittsburgh, PA (Dec. 1990).

[5] M. Brezak, I. Petrovic, Path smooth Using Clothoids for Differential Drive Mobile Robots, in: 18th IFAC World Congress 2011, 2011.

[6] B. Sencer, K. Ishizaki, E. Shamoto, A curvature optimal sharp corner smooth algorithm for high-speed feed motion generation of NC systems along linear tool paths, International Journal of Advanced Manufacturing Technology 76 (9-12) (2014) 1977–1992. doi:10.1007/s00170-014-6386-2.

[7] A. Shahzadeh, A. Khosravi, S. Nahavandi, Path planning for cnc machines considering centripetal acceleration and jerk, in: 2013 IEEE International Conference on Systems, Man, and Cybernetics, 2013, pp. 1759–1764. doi:10.1109/SMC.2013.303.

[8] K. Erkorkmaz, Y. Altintas, High speed CNC system design. Part I: Jerk limited trajectory generation and quintic spline interpolation, International Journal of Machine Tools and Manufacture 41 (9) (2001) 1323–1345. doi:10.1016/S0890-6955(01)00002-5.

[9] A. Yuen, K. Zhang, Y. Altintas, Smooth trajectory generation for five-axis machine tools, International Journal of Machine Tools and Manufacture 71 (2013) 11–19. doi:10.1016/j.ijmachtools.2013.04.002.

[10] K. Erkorkmaz, C. H. Yeung, Y. Altintas, Virtual CNC system. Part II. High speed contouring application, International Journal of Machine Tools and Manufacture 46 (10) (2006) 1124–1138. `doi:10.1016/j.ijmachtools.2005.08.001`.

[11] F. C. Wang, D. C. H. Yang, Nearly arc-length parameterized quintic-spline interpolation for precision machining, Computer-Aided Design 25 (5) (1993) 281–288. `doi:10.1016/0010-4485(93)90085-3`.

[12] K. Erkorkmaz, Y. Altintas, Quintic Spline Interpolation With Minimal Feed Fluctuation (2005). `doi:10.1115/1.1830493`.

[13] S. Yutkowitz, Apparatus and method for smooth cornering in a motion control system, US Patent 6,922,606 (Jul. 26 2005).

[14] X. Beudaert, P. Y. Pechard, C. Tournier, 5-Axis Tool Path Smoothing Based on Drive Constraints, International Journal of Machine Tools and Manufacture 51 (12) (2011) 958–965. `doi:10.1016/j.ijmachtools.2011.08.014`.

[15] K. Yang, S. Sukkarieh, An Analytical Continuous-Curvature Path-smoothing Algorithm, IEEE Transactions on Robotics 26 (3) (2010) 561–568. `doi:10.1109/TRO.2010.2042990`.

[16] Q. Bi, Y. Wang, L. Zhu, H. Ding, A Practical Continuous-Curvature Bezier Transition Algorithm for High-Speed Machining of Linear Tool Path, Intelligent Robotics and Applications: 4th International Conference, ICIRA 2011, Aachen, Germany, December 6-8, 2011, Proceedings, Part II (2011) 465–476`doi:10.1007/978-3-642-25489-5_45`.
URL `http://dx.doi.org/10.1007/978-3-642-25489-5{_}45`

[17] Q. Bi, J. Shi, Y. Wang, L. Zhu, H. Ding, Analytical curvature-continuous dual-Bezier corner transition for five-axis linear tool path, International Journal of Machine Tools and Manufacture 91 (2015) 96–108. `doi:10.1016/j.ijmachtools.2015.02.002`.

[18] D. J. Walton, D. S. Meek, G2 blends of linear segments with cubics and Pythagorean-hodograph quintics, International Journal of Computer Mathematics 86 (9) (2009) 1498–1511. `doi:10.1080/00207160701828157`.
URL `http://www.tandfonline.com/doi/abs/10.1080/00207160701828157`

[19] W. Fan, C.-H. Lee, J.-H. Chen, A realtime curvature-smooth interpolation scheme and motion planning for CNC machining of short line segments, International Journal of Machine Tools and Manufacture 96 (2015) 27–46. `doi:10.1016/j.ijmachtools.2015.04.009`.

[20] H. Zhao, L. Zhu, H. Ding, A real-time look-ahead interpolation methodology with curvature-continuous B-spline transition scheme for CNC machining of short line segments, International Journal of Machine Tools and Manufacture 65 (5-8) (2013) 88–98. `doi:10.1016/j.ijmachtools.2012.10.005`.
URL `https://doi.org/10.1016/j.ijmachtools.2012.10.005http://link.springer.com/10.1007/s00170-015-7776-9http://linkinghub.elsevier.com/retrieve/pii/S0890695512001885`

[21] X. Beudaert, S. Lavernhe, C. Tournier, 5-Axis Local Corner Rounding of Linear Tool Path Discontinuities, International Journal of Machine Tools and Manufacture 73 (2013) 9–16. `doi:10.1016/j.ijmachtools.2013.05.008`.

[22] S. Tulsyan, Y. Altintas, Local toolpath smoothing for five-axis machine tools, International Journal of Machine Tools and Manufacture 96 (2015) 15–26. `doi:10.1016/j.ijmachtools.2015.04.014`.

[23] J. Yang, A. Yuen, An analytical local corner smoothing algorithm for five-axis CNC machining, International Journal of Machine Tools and Manufacture 123 (July) (2017) 22–35. `doi:10.1016/j.ijmachtools.2017.07.007`.

[24] J. Shi, Q. Z. Bi, L. M. Zhu, Y. H. Wang, Corner rounding of linear five-axis tool path by dual PH curves blending, International Journal of Machine Tools and Manufacture 88 (2015) 223–236. `doi:10.1016/j.ijmachtools.2014.09.007`.

[25] S. Tajima, B. Sencer, Kinematic corner smoothing for high speed machine tools, International Journal of Machine Tools and Manufacture 108 (2016) 27–43. `doi:10.1016/j.ijmachtools.2016.05.009`.

[26] S. Tajima, B. Sencer, Global tool-path smoothing for CNC machine tools with uninterrupted acceleration, International Journal of Machine Tools and Manufacture 121 (November 2016) (2017) 81–95. `doi:10.1016/j.ijmachtools.2017.03.002`.

[27] M. a. Heald, Rational Approximations for the Fresnel Integrals, Mathematics of Computation 44 (170) (1985) 459. `doi:10.2307/2007965`.

[28] M. Brezak, I. Petrovic, Real-time approximation of clothoids with bounded error for path planning applications, IEEE Transactions on Robotics 30 (2) (2014) 507–515. `doi:10.1109/TRO.2013.2283928`.

[29] H. Zhao, L. Zhu, H. Ding, A parametric interpolator with minimal feed fluctuation for CNC machine tools using arc-length compensation and feedback correction, International Journal of Machine Tools and Manufacture 75 (2013) 1–8. `doi:10.1016/j.ijmachtools.2013.08.002`.

[30] Y. Sun, Y. Zhao, Y. Bao, D. Guo, A smooth curve evolution approach to the feedrate planning on five-axis toolpath with geometric and kinematic constraints, International Journal of Machine Tools and Manufacture 97 (2015) 86–97. `doi:10.1016/j.ijmachtools.2015.07.002`.

[31] M. Chen, W. S. Zhao, X. C. Xi, Augmented Taylor's expansion method for B-spline curve interpolation for CNC machine tools, International Journal of Machine Tools and Manufacture 94 (2015) 109–119. `doi:10.1016/j.ijmachtools.2015.04.013`.

[32] M. Heng, K. Erkorkmaz, Design of a nurbs interpolator with minimal feed fluctuation and continuous feed modulation capability, International Journal of Machine Tools and Manufacture 50 (3) (2010) 281 – 293. `doi:https://doi.org/10.1016/j.ijmachtools.2009.11.005`.

[33] R. T. Farouki, S. Shah, Real-time CNC interpolators for Pythagorean-hodograph curves, Computer Aided Geometric Design 13 (7) (1996) 583–600. `doi:10.1016/0167-8396(95)00047-X`.

[34] R. Farouki, J. Manjunathaiah, D. Nicholas, G.-F. Yuan, S. Jee, Variable-feedrate CNC interpolators for constant material removal rates along Pythagorean-hodograph curves, Computer-Aided Design 30 (8) (1998) 631–640. `doi:10.1016/S0010-4485(98)00020-7`.

[35] T. R. Kramer, F. M. Proctor, E. Messina, The NIST RS274/NGC Interpreter-Version 3, National Institute of Standards and Technology (NIST), NISTIR 6556.

[36] AMC5 CNC by ANCA Motion, `https://www.ancamotion.com/cnc/amc5`, accessed: 2017-10-07.

[37] J. Jahanpour, M.-C. Tsai, M.-Y. Cheng, I.-H. Liu, Contour Following Using C2 PH Quintic Spline Curve Interpolators, IFAC Proceedings Volumes 44 (1) (2011) 9361 – 9366, 18th IFAC World Congress.

[38] A. A. Taha, A. Hanbury, An efficient algorithm for calculating the exact Hausdorff distance, IEEE transactions on pattern analysis and machine intelligence 37 (11) (2015) 2153–2163.

[39] Catalog NC 62: SINUMERIK 840 Equipment for Machine Tools (2016).

[40] W. L. Nelson, Continuous steering-function control of robot carts, IEEE Transactions on Industrial Electronics 36 (3) (1989) 330–337. `doi:10.1109/41.31495`.

[41] A. C. Lee, M. T. Lin, Y. R. Pan, W. Y. Lin, The feedrate scheduling of NURBS interpolator for CNC machine tools, CAD Computer Aided Design 43 (6) (2011) 612–628. `arXiv:ISSN:00104485, doi:10.1016/j.cad.2011.02.014`.

[42] D. Walton, D. Meek, Approximation of quadratic bzier curves by arc splines, Journal of Computational and Applied Mathematics 54 (1) (1994) 107 – 120. `doi:https://doi.org/10.1016/0377-0427(94)90398-0`.
URL `http://www.sciencedirect.com/science/article/pii/0377042794903980`

[43] Y. Kanayama, B. I. Hartman, Smooth local path planning for autonomous vehicles, in: Proceedings, 1989 International Conference on Robotics and Automation, 1989, pp. 1265–1270 vol.3. `doi:10.1109/ROBOT.1989.100154`.

16