



## A Note on Robust Biarc Computation

Enrico Bertolazzi<sup>1</sup> , Marco Frego<sup>2</sup> ,

<sup>1</sup>Department of Industrial Engineering – University of Trento, Italy, [enrico.bertolazzi@unitn.it](mailto:enrico.bertolazzi@unitn.it)

<sup>2</sup>Department of Information Engineering and Computer Science – University of Trento, [marco.frego@unitn.it](mailto:marco.frego@unitn.it)

Corresponding author: Enrico Bertolazzi, [enrico.bertolazzi@unitn.it](mailto:enrico.bertolazzi@unitn.it)

**Abstract.** A robust algorithm for the numerical computation of biarcs, i.e.  $G^1$  curves composed of two arcs of circle, is presented. Many algorithms exist, but are based on geometric constructions, which must consider different geometrical configurations. The proposed algorithm uses an analytic construction, which explicitly states the solution and does not require the identification of geometric cases. Moreover, singular angles configurations are treated smoothly.

The proposed algorithm is compared with the Matlab's routine `rscvn` that solves geometrically the same problem. Numerical experiments show that Matlab's routine sometimes fails near quasi singular configurations. Moreover, Matlab's routine has a non natural choice of the solution for large angles. Finally, the proposed solution depends smoothly on the geometrical parameters, so that it can be easily included in more complex algorithms involving splines of biarcs or least squares data fitting.

**Keywords:** Biarc, Two-point  $G^1$  Hermite interpolation, Geometric Continuity

**DOI:** <https://doi.org/10.14733/cadaps.2019.822-835>

### 1 Introduction

In the industrial applications of curves there are two philosophies: one is the use of highly sophisticated polynomial splines or transcendental curves like high degree Bèzier curves [7], rational functions [28, 35, 12], clothoid curves [3, 23, 30, 21, 32], hodographs [11, 16], etc., which can produce continuous paths up to the curvature or higher derivatives, but at a relatively expensive computational cost, usually because there are no closed form solutions and a system of nonlinear equations must be numerically solved. The other side of the coin is the employment of low degree polynomials, for instance piecewise linear interpolants or circular arc splines. The advantage of using this family of curves is that, at the price of losing some precision and smoothness, the computational times required to produce a path are in practice negligible, because the associated interpolation problem can be solved with elementary actions. Moreover, sometimes it is simply not necessary to go beyond  $G^1$  continuity, a typical case is represented by real time applications.

In this paper we discuss an improvement of the algorithm for  $G^1$  biarc fitting used in Matlab. A biarc is a curve obtained by connecting two circle arcs that match with  $G^1$  continuity and interpolate two given points and two tangent directions expressed as angles. Biarcs have several interesting properties, first of all, they are easy to understand and to use: in fact the arclength computation is straightforward, the tangent vector field is continuous and defined everywhere, the curvature is defined almost everywhere and is piecewise constant. Moreover, they are very useful in several applications, for instance, they are effectively used in the approximation of higher degree curves [26, 18, 10] or spirals [23], they easily produce curves particularly used in CNC machining and milling, where the cutting devices follow the so called G-code, i.e. a path composed of straight lines and circles. Other applications of biarcs are in Computer Aided Design or Manufacturing (CAD-CAM), where they are used to specify the path [33] or the offset of a more general curve, [13] or approximating NURBS and data [25, 26, 34, 15].

**Related work.** Biarcs were originally proposed in an industrial environment rather than in an academic one, and from the 1970s they have been studied extensively by Bèzier [7], Bolton [8] and Sabin [27]. A general theoretical framework for a complete classification of the biarcs, in the Möbius plane, is proposed in [17]. The solution of the biarc interpolation problem is not unique because the imposed constraints leave one degree of freedom, thus there is a one-dimensional family of interpolating biarcs to general planar  $G^1$  Hermite data. Different choices of this free parameter give origin to different interpolation schemes. The most used construction techniques build the biarc by equal chord or by parallel tangent [31], other constructions minimize the distance between a supporting polygon and the biarc [24]. In the first case the length of the two arcs is chosen equal, in the second case the tangent at the joint point is chosen parallel to the segment that connects the initial with the final point, [20, 22, 23]. In all cases, it can be shown, see for instance [31], all the possible joint points must be on a certain circle. These solutions are based on a geometric approach and consider many different cases, as detailed in [17] and in the references therein contained. Typical cases are the C-shaped, S-shaped and J-shaped biarcs [14, 17, 9].

**Paper contribution.** The algorithm herein proposed extends the range of Hermite data of Matlab's implementation which fails or gives a non-consistent solution in certain configurations. These cases are discussed with examples in Section 4. Following our approach used for the solution of the  $G^1$  Hermite Interpolation Problem with clothoid curves, [6, 4], we propose herein a purely analytic solution to the biarc problem, that does not require to split the problem in mutually exclusive cases. This analytic construction is based on the representation of a circle arc with the smooth function  $\text{sinc}(x) := \sin(x)/x$  that permits to manage smoothly also the singular cases, i.e. when the circle degenerates to a straight line. At the best of the Authors' knowledge, this approach based on the  $\text{sinc}(x)$  function is a novelty. We select the free parameter required to close the system of equations in the same way as Matlab's Curve Fitting Toolbox implementation ([19], page 12-218). The construction is explained in detail in the next sections. The issue of Matlab's function for biarcs (`rscvn`) is that it cannot solve certain configurations of angles and that it gives a non-consistent solution for some range of angles, as it is discussed in the experimental section. We show how to overcome this problem while maintaining the same approach for the construction of the biarc. A C++ implementation of the proposed algorithm is available from [2] (with other useful tools for handling clothoids) also with Matlab interface via mex files. There is also a standalone version available at Matlab Central [5].

## 2 Biarc Formulation

The biarc problem requires to find the pair of circle segments (possibly degenerate, as we will clarify next) that connect two points in the plane with assigned initial and final angles [9]. More formally, it is the solution of the  $G^1$  Hermite Interpolation Problem with two arcs. Let  $\mathbf{p}_0 = (x_0, y_0)^T$  and  $\mathbf{p}_1 = (x_1, y_1)^T$  be two points in the plane  $\mathbb{R}^2$ ,  $\vartheta_0$  and  $\vartheta_1$  be the associated angles, then the biarc problem requires to find the solution of

the following Boundary Value Problem (BVP):

$$\begin{aligned} x'(\ell) &= \cos \theta(\ell), & x(0) &= x_0, & x(L) &= x_1, \\ y'(\ell) &= \sin \theta(\ell), & y(0) &= y_0, & y(L) &= y_1, \\ \theta'(\ell) &= k(\ell), & \theta(0) &= \vartheta_0, & \theta(L) &= \vartheta_1, \end{aligned} \quad (1)$$

where the curvilinear abscissa  $\ell$  is in the range  $[0, L]$ .

As for the  $G^1$  fitting of the Clothoids (see [3]), the angles  $\vartheta_0$  and  $\vartheta_1$  are assumed bounded in the range

$$(\vartheta_0 - \omega) \in [-\pi, \pi], \quad (\vartheta_1 - \omega) \in [-\pi, \pi], \quad (2)$$

where  $\omega = \text{atan2}(y_1 - y_0, x_1 - x_0)$  is the solution of the problem  $x_1 - x_0 = d \cos \omega$ ,  $y_1 - y_0 = d \sin \omega$  with  $d = \sqrt{(x_1 - x_0)^2 + (y_1 - y_0)^2}$ .

The above equations ensure that the solution exhibits  $G^1$  continuity, however, because there are not enough degrees of freedom, in general, it is not possible to satisfy (1) with a single arc or straight line. Therefore, the curvature must be piecewise constant:

$$k(\ell) = \begin{cases} \kappa_0 & 0 \leq \ell < \ell_*, \\ \kappa_1 & \ell_* \leq \ell \leq L, \end{cases} \quad (3)$$

where we assume that the curvilinear abscissa  $\ell$  runs from 0 to  $L$  and the curvature has a jump at  $\ell_*$ , with  $\ell_* \in (0, L)$ . The point at  $\ell_*$  is where the two arcs join. The two curvatures  $\kappa_0, \kappa_1$  are real values, which can take the value zero. These values are associated to the radii of curvature of the two circles, if they are different from zero. This formulation of the problem also contains degenerate cases, where the solution is not composed of two circles (i.e. we allow  $\kappa_0 = 0$  or  $\kappa_1 = 0$ ), meaning that a straight line can be part of the solution. Other particular cases are represented by a single arc of circle or by a single straight line. Fixing the  $G^1$  Hermite condition, i.e. the boundary condition of the BVP (1), the solution depends on four free parameters  $\ell_*, L, \kappa_0, \kappa_1$ , to be determined.

**Lemma 1** *The solution of BVP (1) as a function of the parameters  $\ell_*, L, \kappa_0, \kappa_1$  can be written as*

$$\begin{aligned} x(\ell) &= \begin{cases} x_0 + f(\ell, \vartheta_0, \kappa_0) & \ell \leq \ell_*, \\ x_1 + f(\ell - L, \vartheta_1, \kappa_1) & \ell \geq \ell_*, \end{cases} & f(\ell, \vartheta, \kappa) &= \ell \operatorname{sinc} \left( \frac{\kappa \ell}{2} \right) \cos \left( \vartheta + \frac{\kappa \ell}{2} \right) \\ y(\ell) &= \begin{cases} y_0 + g(\ell, \vartheta_0, \kappa_0) & \ell \leq \ell_*, \\ y_1 + g(\ell - L, \vartheta_1, \kappa_1) & \ell \geq \ell_*, \end{cases} & g(\ell, \vartheta, \kappa) &= \ell \operatorname{sinc} \left( \frac{\kappa \ell}{2} \right) \sin \left( \vartheta + \frac{\kappa \ell}{2} \right) \\ \theta(\ell) &= \begin{cases} \vartheta_0 + \ell \kappa_0 & \ell \leq \ell_*, \\ \vartheta_1 + (\ell - L) \kappa_1 & \ell \geq \ell_*, \end{cases} \end{aligned} \quad (4)$$

where  $\ell$  is the arc length of the curve and the function  $\operatorname{sinc} z$  is

$$\operatorname{sinc} z := \frac{\sin z}{z} = 1 + \sum_{n=1}^{\infty} (-1)^n \frac{z^{2n}}{(2n+1)!}. \quad (5)$$

**proof.** By direct integration, the solution of BVP (1) is for  $\ell \leq \ell_*$ :

$$x(\ell) = x_0 - \frac{\sin \vartheta_0 + \sin(\kappa_0 \ell + \vartheta_0)}{\kappa_0}, \quad y(\ell) = y_0 + \frac{\cos \vartheta_0 - \cos(\kappa_0 \ell + \vartheta_0)}{\kappa_0}.$$

The application of the standard trigonometric addition identities

$$\sin(\alpha + \beta) = \sin \alpha \cos \beta + \cos \alpha \sin \beta, \quad \cos(\alpha + \beta) = \cos \alpha \cos \beta - \sin \alpha \sin \beta, \quad (6)$$

to  $\sin(\kappa_0 \ell + \vartheta_0)$  and  $\cos(\kappa_0 \ell + \vartheta_0)$  with  $\alpha = \kappa_0 \ell$  and  $\beta = \vartheta_0$  yields

$$\begin{aligned} x(\ell) &= x_0 - \frac{\sin \vartheta_0 + \sin(\kappa_0 \ell) \cos \vartheta_0 + \cos(\kappa_0 \ell) \sin \vartheta_0}{\kappa_0}, \\ y(\ell) &= y_0 + \frac{\cos \vartheta_0 - \cos(\kappa_0 \ell) \cos \vartheta_0 + \sin(\kappa_0 \ell) \sin \vartheta_0}{\kappa_0}. \end{aligned}$$

The collection of  $\sin \vartheta_0$ ,  $\cos \vartheta_0$ , followed by a multiplication and division by  $\ell$  results in

$$\begin{aligned} x(\ell) &= x_0 + \ell \cos \vartheta_0 \frac{\sin(\kappa_0 \ell)}{\kappa_0 \ell} - \ell \sin \vartheta_0 \frac{1 - \cos(\kappa_0 \ell)}{\kappa_0 \ell}, \\ y(\ell) &= y_0 + \ell \cos \vartheta_0 \frac{1 - \cos(\kappa_0 \ell)}{\kappa_0 \ell} + \ell \sin \vartheta_0 \frac{\sin(\kappa_0 \ell)}{\kappa_0 \ell}. \end{aligned}$$

Using the identity

$$\frac{\sin z}{z} = \cos \frac{z}{2} \operatorname{sinc} \frac{z}{2}, \quad \frac{1 - \cos z}{z} = \sin \frac{z}{2} \operatorname{sinc} \frac{z}{2},$$

it follows that

$$\begin{aligned} x(\ell) &= x_0 + \ell \operatorname{sinc} \frac{\kappa_0 \ell}{2} \left( \cos \vartheta_0 \cos \frac{\kappa_0 \ell}{2} - \sin \vartheta_0 \sin \frac{\kappa_0 \ell}{2} \right), \\ y(\ell) &= y_0 + \ell \operatorname{sinc} \frac{\kappa_0 \ell}{2} \left( \cos \vartheta_0 \sin \frac{\kappa_0 \ell}{2} + \sin \vartheta_0 \cos \frac{\kappa_0 \ell}{2} \right). \end{aligned}$$

With the application of the standard trigonometric identities (6), equation (4) easily follows (for  $\ell \leq \ell_*$ ). The branch  $\ell \geq \ell_*$  is computed analogously.  $\square$

**Remark 1** To compute (5) near the critical point  $z = 0$ , its Taylor approximations with a small number of terms is used. For example, to limit the error below  $1.3 \cdot 10^{-20}$ , from

$$\left| \left( 1 - \frac{z^2}{6} \left( 1 - \frac{z^2}{20} \right) \right) - \frac{\sin z}{z} \right| \leq \frac{|z|^6}{5040},$$

it is enough to have  $|z| \leq 0.002$ .

The continuity conditions for (4) introduce the three constraints:

$$\begin{aligned} \ell_* \operatorname{sinc}\left(\frac{\ell_* \kappa_0}{2}\right) \cos\left(\vartheta_0 + \frac{\ell_* \kappa_0}{2}\right) - (\ell_* - L) \operatorname{sinc}\left(\frac{(\ell_* - L) \kappa_1}{2}\right) \cos\left(\vartheta_1 - \frac{(\ell_* - L) \kappa_1}{2}\right) &= x_1 - x_0, \\ \ell_* \operatorname{sinc}\left(\frac{\ell_* \kappa_0}{2}\right) \sin\left(\vartheta_0 + \frac{\ell_* \kappa_0}{2}\right) - (\ell_* - L) \operatorname{sinc}\left(\frac{(\ell_* - L) \kappa_1}{2}\right) \sin\left(\vartheta_1 - \frac{(\ell_* - L) \kappa_1}{2}\right) &= y_1 - y_0, \\ \vartheta_0 + \ell_* \kappa_0 &= \vartheta_1 + (\ell_* - L) \kappa_1. \end{aligned} \quad (7)$$

As pointed out in several references, [20, 18, 17], the solution of the biarc problem is *not unique*, in fact the three constraints on the four parameters leave one degree of freedom that allows many different geometric constructions [31]. Introducing as free parameter the junction angle  $\vartheta_* = \theta(\ell_*)$ , it is possible to parameterize all the solutions of BVP (1). The use of  $\vartheta_*$  in the angle conditions of (7) permits to eliminate  $\kappa_0$  and  $\kappa_1$ . Starting with the identities for the angles computed at  $\ell = \ell_*$ :

$$\ell_* \kappa_0 = \vartheta_* - \vartheta_0, \quad (\ell_* - L) \kappa_1 = \vartheta_* - \vartheta_1. \quad (8)$$

If we introduce the scaled lengths

$$a = \ell_* \operatorname{sinc}\left(\frac{\vartheta_* - \vartheta_0}{2}\right), \quad b = (L - \ell_*) \operatorname{sinc}\left(\frac{\vartheta_* - \vartheta_1}{2}\right), \quad (9)$$

the continuity conditions (7) are equivalent to the linear system:

$$\begin{pmatrix} \cos\left(\frac{\vartheta_* + \vartheta_0}{2}\right) & \cos\left(\frac{\vartheta_* + \vartheta_1}{2}\right) \\ \sin\left(\frac{\vartheta_* + \vartheta_0}{2}\right) & \sin\left(\frac{\vartheta_* + \vartheta_1}{2}\right) \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} x_1 - x_0 \\ y_1 - y_0 \end{pmatrix}. \quad (10)$$

All the possible solutions of the biarc problem are given by the angles  $\vartheta_*$  for which the solution of the linear system (10) exists. The determinant of the linear system is  $\sin\left(\frac{\vartheta_1 - \vartheta_0}{2}\right)$  and is zero for angles  $\vartheta_1 = \vartheta_0 + k\pi$  and  $k \in \mathbb{Z}$ .

**Remark 2** The condition  $\vartheta_1 = \vartheta_0 + k\pi$  with  $k \in \mathbb{Z}$  cannot be excluded. For example, when  $\vartheta_0 = \vartheta_1 = \vartheta$ ,  $x_0 = y_0 = 0$ ,  $x_1 = d \cos \omega$ ,  $y_1 = d \sin \omega$  the linear system (10) reduces to

$$\begin{pmatrix} \cos\left(\frac{\vartheta_* + \vartheta}{2}\right) & \cos\left(\frac{\vartheta_* + \vartheta}{2}\right) \\ \sin\left(\frac{\vartheta_* + \vartheta}{2}\right) & \sin\left(\frac{\vartheta_* + \vartheta}{2}\right) \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} = d \begin{pmatrix} \cos \omega \\ \sin \omega \end{pmatrix},$$

thus, a natural choice is to set  $\frac{\vartheta_* + \vartheta}{2} = \omega$ , so that  $\vartheta_* = 2\omega - \vartheta$  and

$$\begin{pmatrix} \cos \omega & \cos \omega \\ \sin \omega & \sin \omega \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} = d \begin{pmatrix} \cos \omega \\ \sin \omega \end{pmatrix}. \quad (11)$$

However, the solution is not unique because all  $a$  and  $b$  such that  $a + b = d$  solve the linear system and the solutions are two half-circles with radii that satisfy  $r_0 + r_1 = d/2$ .

From, the previous remark, when the linear system is singular, an additional condition must be considered in order to have a unique solution.

For example, a minimization of the curvature jump or the weighted sum of the squares of the curvatures (as in [34]) implies  $a = b = 1/2$ . The same result can be obtained looking for the minimum norm least squares solution for linear system (11), which is obtained by multiplying the r.h.s. with the Moore-Penrose pseudo-inverse of the coefficients matrix [1]. The pseudo-inverse is here used to obtain a solution that smoothly depends on its parameters [29]. The use of the minimum norm least squares solution of problem (10) does not guarantee the satisfaction of constraints (7). To have a solution of problem (10) with *smooth* dependence on the parameters  $x_0, y_0, x_1, y_1, \vartheta_0$  and  $\vartheta_1$ , the free parameter  $\vartheta_*$  is assumed to satisfy

$$\vartheta_* = 2\omega - \mathcal{A}(\vartheta_0, \vartheta_1), \quad (12)$$

with  $\mathcal{A}(\vartheta_0, \vartheta_1)$  a smooth function such that

$$\mathcal{A}(\vartheta, \vartheta) = \vartheta, \quad \min\{\vartheta_0, \vartheta_1\} < \mathcal{A}(\vartheta_0, \vartheta_1) < \max\{\vartheta_0, \vartheta_1\}, \text{ for } \vartheta_0 \neq \vartheta_1. \quad (13)$$

For example,  $\mathcal{A}(\vartheta_0, \vartheta_1) = (\vartheta_0 + \vartheta_1)/2$ , which will coincide with the Matlab geometric construction in the non-singular case.

**Lemma 2** *The minimum norm least squares solution  $(a, b)$  of the linear system (10) with  $\vartheta_*$  given by (12), together with properties (13) and*

$$|\vartheta_0 - \omega| \leq \pi, \quad |\vartheta_1 - \omega| \leq \pi, \quad |\vartheta_1 - \vartheta_0| < 2\pi,$$

*exists, is unique, and satisfies  $a > 0$  and  $b > 0$ . Notice that the requirement for the angles assumed in (2) implies the above constraints.*

**proof.** If  $\vartheta_0 \neq \vartheta_1$ , from  $|\vartheta_1 - \vartheta_0| < 2\pi$ , the determinant of (10) is different from zero, the solution is unique and coincides with the minimum norm least squares solution. The solution of the linear system is computed using Cramer's Rule with r.h.s. written in polar coordinates, e.g.

$$\begin{pmatrix} x_1 - x_0 \\ y_1 - y_0 \end{pmatrix} = d \begin{pmatrix} \cos \omega \\ \sin \omega \end{pmatrix}, \quad (14)$$

and using standard trigonometric identities, the solution reduces to

$$a = d \frac{\sin\left(\frac{\vartheta_* + \vartheta_1}{2} - \omega\right)}{\sin\left(\frac{\vartheta_1 - \vartheta_0}{2}\right)} = d \frac{\sin\left(\frac{\vartheta_1 - \mathcal{A}(\vartheta_0, \vartheta_1)}{2}\right)}{\sin\left(\frac{\vartheta_1 - \vartheta_0}{2}\right)}, \quad b = d \frac{\sin\left(\omega - \frac{\vartheta_* + \vartheta_0}{2}\right)}{\sin\left(\frac{\vartheta_1 - \vartheta_0}{2}\right)} = d \frac{\sin\left(\frac{\mathcal{A}(\vartheta_0, \vartheta_1) - \vartheta_0}{2}\right)}{\sin\left(\frac{\vartheta_1 - \vartheta_0}{2}\right)}. \quad (15)$$

Finally, properties (13) easily imply  $a > 0$  and  $b > 0$ . If  $\vartheta_0 = \vartheta_1 = \vartheta$ , system (10) with r.h.s. (14) becomes

$$\begin{pmatrix} \cos \omega & \cos \omega \\ \sin \omega & \sin \omega \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} = d \begin{pmatrix} \cos \omega \\ \sin \omega \end{pmatrix}$$

and the least squares minimum norm solution computed using the Moore-Penrose pseudo-inverse is  $a = b = d/2 > 0$ .  $\square$

**Corollary 3** For  $\mathcal{A}(\vartheta_0, \vartheta_1) = (\vartheta_0 + \vartheta_1)/2$  the solution of the biarc fitting problem (1) can be written as

$$\begin{aligned} t &= d \operatorname{sinc}\left(\frac{\vartheta_1 - \vartheta_0}{4}\right) / \operatorname{sinc}\left(\frac{\vartheta_1 - \vartheta_0}{2}\right), & \vartheta_\star &= 2\omega - \frac{\vartheta_0 + \vartheta_1}{2}, \\ \ell_0 &= t / \left(2 \operatorname{sinc}\left(\frac{\vartheta_\star - \vartheta_0}{2}\right)\right), & \ell_1 &= t / \left(2 \operatorname{sinc}\left(\frac{\vartheta_\star - \vartheta_1}{2}\right)\right), \\ \kappa_0 &= \frac{\vartheta_\star - \vartheta_0}{\ell_0} = \frac{4}{t} \sin\left(\frac{\vartheta_\star - \vartheta_0}{2}\right), & \kappa_1 &= -\frac{\vartheta_\star - \vartheta_1}{\ell_1} = -\frac{4}{t} \sin\left(\frac{\vartheta_\star - \vartheta_1}{2}\right), \end{aligned}$$

where  $\ell_0 = \ell_\star$  and  $\ell_1 = L - \ell_\star$  are the lengths of the first and second arc, respectively.

**proof.** A simple substitution of  $\mathcal{A}(\vartheta_0, \vartheta_1)$  into (15) and with (5) to avoid 0/0, permits to compute

$$a = b = \frac{d}{2} \operatorname{sinc}\left(\frac{\vartheta_1 - \vartheta_0}{4}\right) / \operatorname{sinc}\left(\frac{\vartheta_1 - \vartheta_0}{2}\right).$$

The other biarc parameters follow from (9) and (8). □

**Remark 3** The position of the junction point  $(x_\star, y_\star)$ , as a function of the angle  $\vartheta_\star$ , is given by

$$x_\star = x_0 + \frac{t}{2} \cos\left(\frac{\vartheta_\star + \vartheta_0}{2}\right), \quad y_\star = y_0 + \frac{t}{2} \sin\left(\frac{\vartheta_\star + \vartheta_0}{2}\right),$$

where  $t$  and  $\vartheta_\star$  are the values described in Corollary 3. The value of  $t$  does not depend on  $\vartheta_\star$ , hence the above equations represent a circle, or, in other words, the position of the junction point varies on a circle for different values of  $\vartheta_\star$ . This property confirms a known result, proved for example in [31].

In Corollary 3, the angles are assumed in the range specified in (2). In the practical implementation, the angles are normalized by adding  $2k\pi$  with  $k \in \mathbb{Z}$ , to meet this assumption. Notice that for the computation of  $\kappa_0$  and  $\kappa_1$ , the expression using trigonometric functions are to be preferred instead of the expression that use the division by  $\ell_0$  and  $\ell_1$ . In fact, the division by  $\ell_0$  and  $\ell_1$  is ill conditioned for very small values of these lengths. Moreover, the trigonometric version is valid and accurate also when the problem degenerates to a single segment, i.e., when  $\ell_0 = 0$  or  $\ell_1 = 0$ .

The complete biarc algorithm is implemented in Algorithm 1.

### 3 Matlab's approach

Now we focus on the solution proposed and implemented in Matlab's `rscvn` function, [19], page 12-218, which uses the degree of freedom to assign the direction of the (unit) normal vector  $\mathbf{n}(\ell)$  to the trajectory at  $\ell_\star$ :

$$\mathbf{n}(\ell_\star) = (-\sin\theta(\ell_\star), \cos\theta(\ell_\star))^T. \quad (16)$$

The consequence of assigning  $\mathbf{n}(\ell_\star) = \mathbf{v}$  is that problem (1)–(3) will have at most one solution. According to Matlab's Handbook, such normal vector

“ $\mathbf{v}$  is chosen as the reflection, across the perpendicular to the segment from  $\mathbf{p}_0$  to  $\mathbf{p}_1$ , of the average of the vectors  $\mathbf{n}(0)$  and  $\mathbf{n}(L)$ ”.

**Algorithm 1:** Biarc solution algorithm

---

```

Biarc ( $x_0, y_0, \vartheta_0, x_1, y_1, \vartheta_1$ );
begin
   $d_x \leftarrow x_1 - x_0;$             $d_y \leftarrow y_1 - y_0;$ 
   $d \leftarrow (d_x^2 + d_y^2)^{1/2}$     $\omega \leftarrow \text{atan2}(d_y, d_x);$ 
   $\theta_0 \leftarrow \omega + \text{Range}(\vartheta_0 - \omega);$     $\theta_1 \leftarrow \omega + \text{Range}(\vartheta_1 - \omega);$ 
   $t \leftarrow d \text{sinc}(\frac{\theta_1 - \theta_0}{4}) / \text{sinc}(\frac{\theta_1 - \theta_0}{2})$     $\theta_* \leftarrow 2\omega - \frac{\theta_0 + \theta_1}{2};$ 
   $\Delta\theta_0 \leftarrow \frac{\theta_* - \theta_0}{2};$             $\Delta\theta_1 \leftarrow \frac{\theta_* - \theta_1}{2};$ 
   $\ell_0 \leftarrow t / (2 \text{sinc}(\Delta\theta_0));$     $\ell_1 \leftarrow t / (2 \text{sinc}(\Delta\theta_1));$ 
   $\kappa_0 \leftarrow \frac{4}{t} \sin(\Delta\theta_0);$         $\kappa_1 \leftarrow -\frac{4}{t} \sin(\Delta\theta_1);$ 
   $x_* \leftarrow x_0 + \frac{t}{2} \cos(\frac{\theta_* + \theta_0}{2});$     $y_* \leftarrow y_0 + \frac{t}{2} \sin(\frac{\theta_* + \theta_0}{2});$ 
  return  $[\ell_0, \theta_0, \kappa_0], [\ell_1, \theta_1, \kappa_1], x_*, y_*, \theta_*$ ;
end

Sinc ( $x$ ) // approximate  $(\sin x)/x$  with error  $\leq 1.3 \cdot 10^{-20}$ 
begin
  if  $|x| < 0.002$  then return  $1 + \frac{x^2}{6} \left(1 - \frac{x^2}{20}\right);$ 
  return  $(\sin x)/x$ 
end

Range ( $\theta$ ) // return  $\theta + 2k\pi$  with  $k$  such that the angle is in  $[-\pi, \pi]$ 
begin
  while  $\theta > +\pi$  do  $\theta \leftarrow \theta - 2\pi;$ 
  while  $\theta < -\pi$  do  $\theta \leftarrow \theta + 2\pi;$ 
  return  $\theta$ 
end

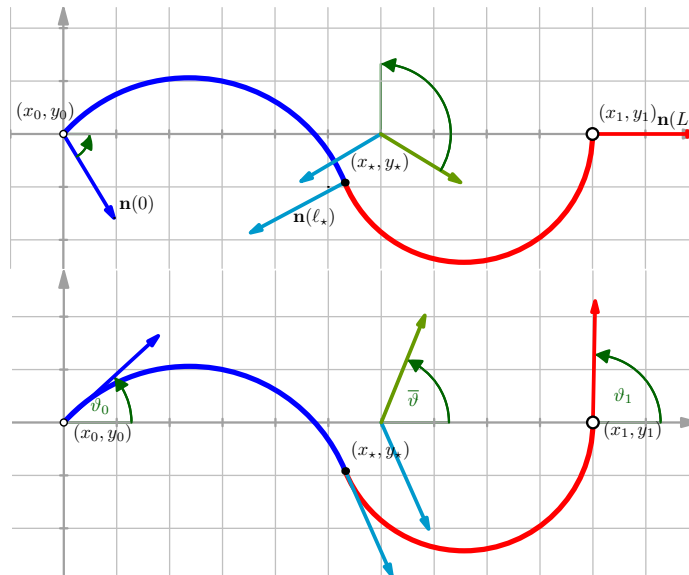
```

---

We elaborate this construction by recasting it into an equivalent one, expressed with the tangent vectors  $\mathbf{t}(\ell)$ . The application of a rotation of  $\pi/2$  to  $\mathbf{n}(\ell_*) = \mathbf{v}$  yields an equivalent condition  $\mathbf{t}(\ell_*) = \mathbf{w}$ , where  $\mathbf{w}$  is reflected along the segment from  $\mathbf{p}_0$  to  $\mathbf{p}_1$ , of the average of the tangents  $\mathbf{t}(0)$  and  $\mathbf{t}(L)$ , see Figure 1. Moreover, this construction can be improved by reasoning on the *angles* instead of the *tangent vectors*. Indeed, it is more convenient to use the average of the angles rather than the average of the vectors, especially when the average of the vectors will yield a null (or very small) vector. In such cases, the normal vector defined by Matlab's choice is not well posed, but the average of the angles (12) is always well defined. Indeed, it falls in the range  $(\omega - \pi, \omega + \pi)$ , so that no  $2\pi$  overflow occurs.

**Remark 4** *The average of the angles, as described above, coincides with the Matlab method of vectors average. In fact, the two results are exactly the same for angles in the range  $(-\frac{\pi}{2}, \frac{\pi}{2})$ . For values outside that range, Matlab reverses the angles: it keeps only the direction, but not the orientation. This produces a path that is travelled from the final to the initial point, unless direction is not relevant for the application, it can be considered a bug. The average of the vectors introduces also numerical problems of cancellation, whereas the proposed solution is stable. These behaviours are discussed more in detail in the numerical tests.*





**Figure 1:** Generalisation of Matlab biarc interpolation scheme, converted from normal vectors (top) to tangent vectors (bottom). The figure shows the case of  $\mathbf{p}_0$  and  $\mathbf{p}_1$  aligned with the  $x$  axis and  $(x_*, y_*)$  the joint point.

Thus, we construct  $\mathbf{w}$  on condition (16) as  $\mathbf{w} = (\cos \vartheta_*, \sin \vartheta_*)^T$  and  $\vartheta_*$  is computed as:

$$\bar{\vartheta} := \frac{\vartheta_0 + \vartheta_1}{2}, \quad \vartheta_* = \omega + (\omega - \bar{\vartheta}) = 2\omega - \bar{\vartheta}$$

with  $\omega = \text{atan2}(y_1 - y_0, x_1 - x_0)$ , e.g.  $\omega$  is the angle that satisfies

$$\begin{cases} x_1 - x_0 = d \cos \omega, \\ y_1 - y_0 = d \sin \omega, \end{cases} \quad d = \left\| \begin{pmatrix} x_1 - x_0 \\ y_1 - y_0 \end{pmatrix} \right\|.$$

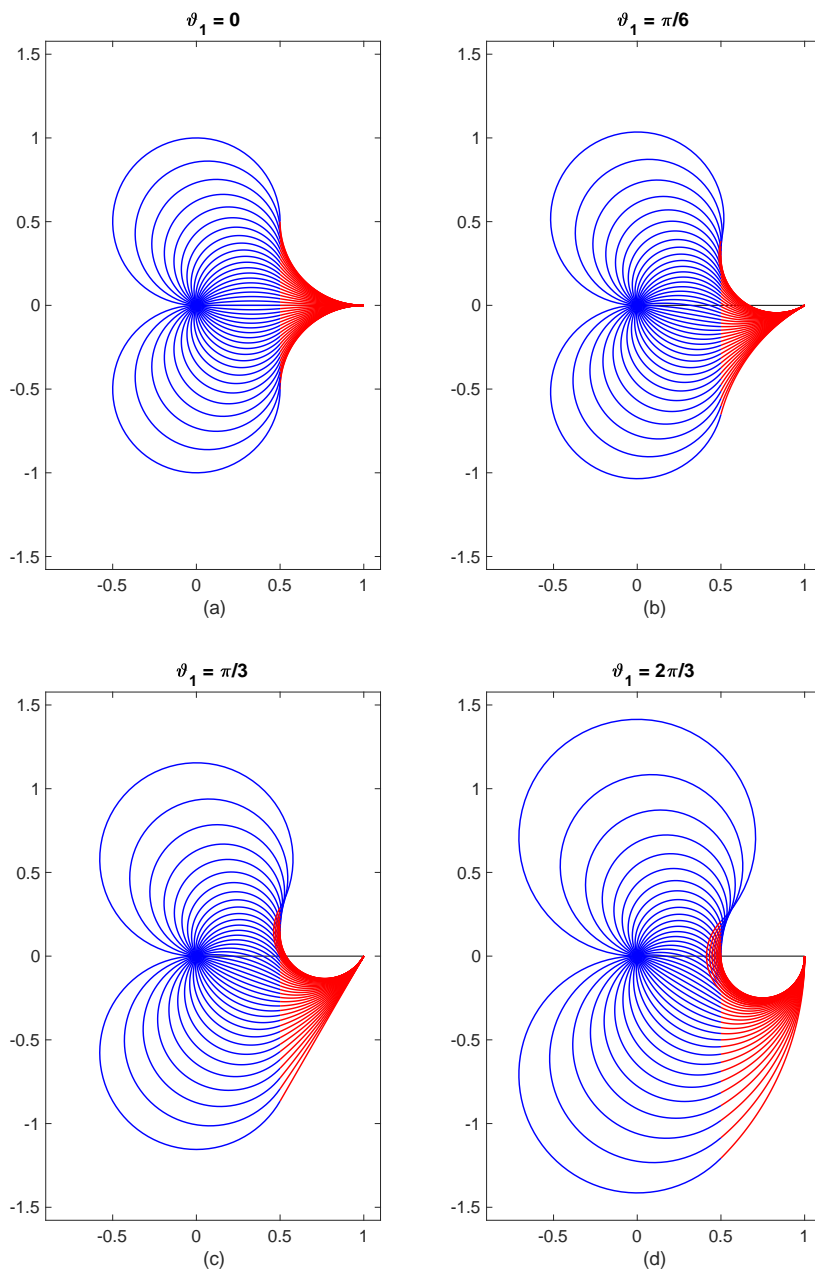
The condition  $\mathbf{n}(\ell_*) = \mathbf{v}$  becomes therefore  $\theta(\ell_*) = \vartheta_*$ .

#### 4 Numerical Tests

In this section we show some numerical experiments to validate the presented algorithm.

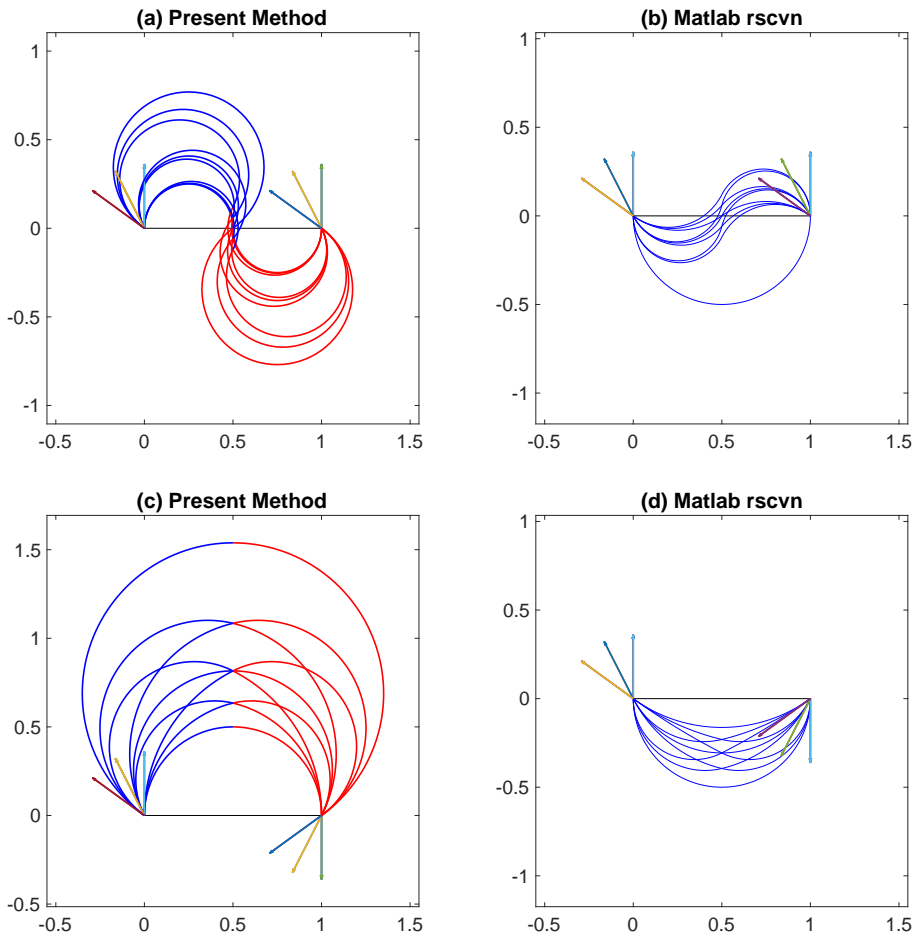
**Test 1.** In the first test, see Figure 2, we create a bouquet of biarcs, all starting in  $\mathbf{p}_0 = (0, 0)$  with angles in the range  $(-\pi, \pi)$  and ending at the point  $\mathbf{p}_1 = (1, 0)$  with different final angles  $\vartheta_1 \in \{0, \pi/6, \pi/3, 2\pi/3\}$ . Figure 2 confirms that the solution of the problem varies with continuity.

**Test 2.** In this test we show that the continuity of the solution is not a property of Matlab's biarc function. In fact, we can see in Figure 3 a direct comparison on the same tests between the algorithm herein proposed (cases (a) and (c)) and Matlab (cases (b) and (d)). In Figure 3 (a) and (c) there is continuity in the variation of the solution, whereas in Figure 3 (b) and (d) we can notice a jump in the solution, which is an undesirable behaviour. In Figure 3 (a) and (b) we plot the solutions for  $\mathbf{p}_0 = (0, 0)$  and  $\mathbf{p}_1 = (1, 0)$ , the angles range



**Figure 2:** Four examples of biarc interpolation with different initial and final angles. The first arc is plotted in blue, the second arc in red.

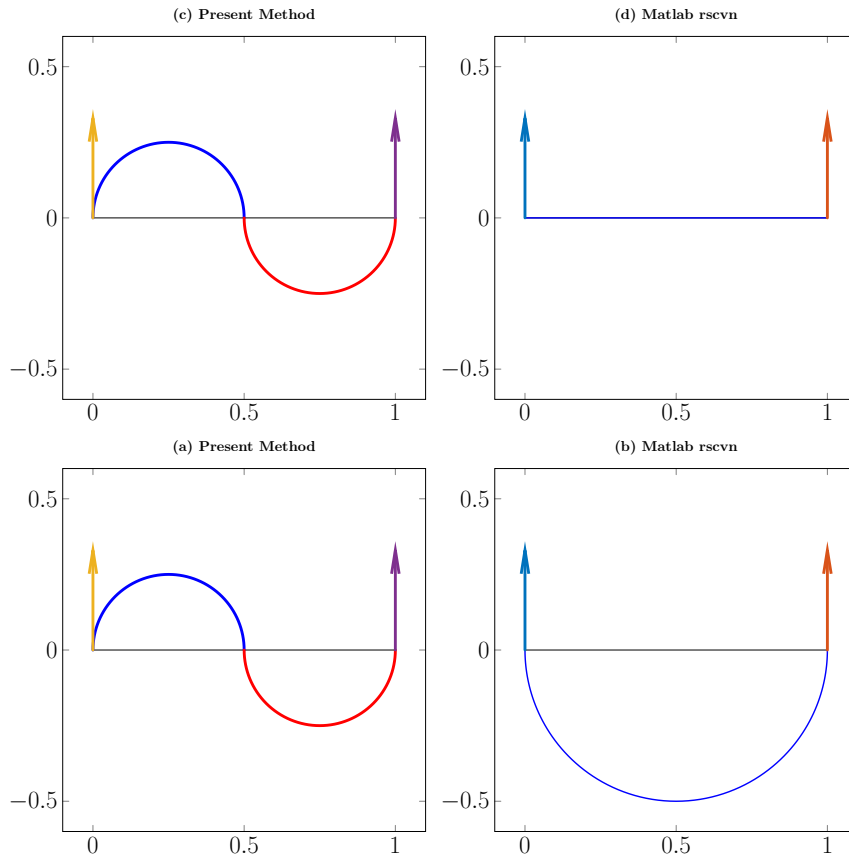
in  $[\pi/2, 4/5\pi]$ , some tangent vectors are shown as arrows. In Figure 3 (c) and (d) we plot the solutions for  $p_0 = (0,0)$  and  $p_1 = (1,0)$ , the initial angles range in  $[\pi/2, 4/5\pi]$ , the final angles are in the range



**Figure 3:** Comparison between present method (a), (c) and Matlab (b) and (c). Arrows indicate the initial and final tangent vectors. The three initial directions are combined with three final direction giving nine curves. Matlab's output exhibits wrong selections in the solution, which does not vary with continuity.

$[-4/5\pi, -\pi/2]$ . In both cases (b) and (d) Matlab selects a non-natural solution.

**Test 3.** As a last example, we show in Figure 4 two cases where Matlab produces a wrong solution, when it is close to singular configurations, that is, when the average of the vectors used to find the joint point is zero or almost zero. In Figure 4 (a), our algorithm correctly interpolates  $\mathbf{p}_0 = (0, 0)$  and  $\mathbf{p}_1 = (1, 0)$  with  $\vartheta_0 = \vartheta_1 = \pi/2$ , producing a classic S-shaped biarc, whereas in (b), Matlab selects the wrong angle and produces a C-shaped biarc that violates the tangent at the initial point. In Figure 4 (c) and (d) we show the solution of the same problem with slightly perturbed angles:  $\mathbf{p}_0 = (0, 0)$ ,  $\mathbf{p}_1 = (1, 0)$ , but  $\vartheta_0 = \vartheta_1 = \pi/2 - 10^4\epsilon$ , where  $\epsilon$  is the machine epsilon, i.e. a very small number. In Figure 4 (c), our algorithm produces a solution that is very close to the non-perturbed case (a), whereas Matlab gives a line segment, that is incompatible with the correct solution (c) or with the non-perturbed (still wrong) solution of (b).



**Figure 4:** Comparison between present method (a), (c) with Matlab (b) and (c). Arrows indicate the initial and final tangent vectors. Cases (a) and (b) are the non-perturbed angles  $\vartheta_0 = \vartheta_1 = \pi/2$ , cases (c) and (d) have  $\vartheta_0 = \vartheta_1 = \pi/2 - 10^4\epsilon$ , where  $\epsilon$  is the machine epsilon.

## 5 Conclusions

A robust algorithm for the numerical computation of biarcs is presented. Differently from geometric based solutions, it is not necessary to consider many geometrical configurations and the (unique) solution is given in closed form. The singular configuration (when the angles satisfy  $\vartheta_0 = \vartheta_1$ ) is solved smoothly by using the  $\text{sinc}(x)$  function. The Matlab's routine `rscvn` solves geometrically the same problem; this has the drawback that it is not possible to find the correct biarc in all the configurations. Finally, `rscvn` fails to compute the biarc when the configuration is *almost singular*. The biarc computed by the herein proposed algorithm smoothly depends on the parameters so that it can be easily included in more complex algorithms like splines of biarcs or least squares data fitting.

## 6 Acknowledgements

This paper has received funding from the European Unions Horizon 2020 Research and Innovation Programme - Societal Challenge 1 (DG CON- NECT/H) under grant agreement *n.* 643644 "ACANTO - A Cyberphysical social NeTwork using robot friends".

## ORCID

Enrico Bertolazzi <http://orcid.org/0000-0003-0487-5210>

Marco Frego <http://orcid.org/0000-0003-2855-9052>

## References

- [1] Berman, A.; Plemmons, R.J.: Nonnegative Matrices in the Mathematical Sciences. Classics in Applied Mathematics. Society for Industrial and Applied Mathematics, 1994. ISBN 9780898713213.
- [2] Bertolazzi, E.; Frego, M.:  $G^1$  fitting with clothoids. <http://www.mathworks.com/matlabcentral/fileexchange/42113-g1-fitting-with-clothoids>, 2013.
- [3] Bertolazzi, E.; Frego, M.:  $G^1$  fitting with clothoids. Mathematical Methods in the Applied Sciences, 38(5), 881–897, 2015. ISSN 1099-1476. <http://doi.org/10.1002/mma.3114>.
- [4] Bertolazzi, E.; Frego, M.: Interpolating clothoid splines with curvature continuity. Mathematical Methods in the Applied Sciences, 41(4), 1723–1737, 2017. <http://doi.org/10.1002/mma.4700>.
- [5] Bertolazzi, E.; Frego, M.: Biarc. <https://it.mathworks.com/matlabcentral/fileexchange/69782-biarc>, 2018.
- [6] Bertolazzi, E.; Frego, M.: On the  $G^2$  Hermite interpolation problem with clothoids. Journal of Computational and Applied Mathematics, 341, 99–116, 2018. ISSN 0377-0427. <http://doi.org/10.1016/j.cam.2018.03.029>.
- [7] Bézier, P.: Numerical control: mathematics and applications. Wiley, 1970.
- [8] Bolton, K.: Biarc curves. Computer-Aided Design, 7(2), 89–92, 1975.
- [9] Deng, C.; Ma, W.: Matching admissible  $G^2$  Hermite data by a biarc-based subdivision scheme. Computer Aided Geometric Design, 29(6), 363–378, 2012. ISSN 0167-8396. <http://doi.org/10.1016/j.cagd.2012.03.010>.
- [10] Deng, C.; Ma, W.: A biarc based subdivision scheme for space curve interpolation. Computer Aided Geometric Design, 31(9), 656–673, 2014. ISSN 0167-8396. <http://doi.org/10.1016/j.cagd.2014.07.003>.
- [11] Dong, B.; Farouki, R.T.: Algorithm 952: PHquintic: a library of basic functions for the construction and analysis of planar quintic Pythagorean-hodograph curves. ACM Transactions on Mathematical Software, 41(4), Art. 28, 20, 2015. ISSN 0098-3500. <http://doi.org/10.1145/2699467>.
- [12] Farin, G.E.: NURBS. A K Peters, Ltd., Natick, MA, second ed., 1999. ISBN 1-56881-084-9.
- [13] Kim, Y.J.; Lee, J.; Kim, M.S.; Elber, G.: Efficient offset trimming for planar rational curves using biarc trees. Computer Aided Geometric Design, 29(7), 555–564, 2012. ISSN 0167-8396. <http://doi.org/10.1016/j.cagd.2012.03.014>. Geometric Modeling and Processing 2012.
- [14] Koc, B.; Lee, Y.S.; Ma, Y.: Max-fit biarc fitting to stl models for rapid prototyping processes. In Proceedings of the Sixth ACM Symposium on Solid Modeling and Applications, SMA '01, 225–233. ACM, New York, NY, USA, 2001. ISBN 1-58113-366-9. <http://doi.org/10.1145/376957.376983>.
- [15] Kovač, B.; Žagar, E.: Curvature approximation of circular arcs by low-degree parametric polynomials. Journal of Numerical Mathematics, 24(2), 95–104, 2016. <http://doi.org/10.1515/jnma-2014-0046>.
- [16] Kozak, J.; Krajnc, M.; Rogina, M.; Vitrih, V.: Pythagorean-hodograph cycloidal curves. Journal of Numerical Mathematics, 23(4), 345–360, 2015. ISSN 1570-2820. <http://doi.org/10.1515/jnma-2015-0023>.
- [17] Kurnosenko, A.I.: Biarcs and bilens. Comput. Aided Geom. Des., 30(3), 310–330, 2013. ISSN 0167-8396. <http://doi.org/10.1016/j.cagd.2012.12.002>.

- [18] Maier, G.: Optimal arc spline approximation. *Computer Aided Geometric Design*, 31(5), 211–226, 2014. ISSN 0167-8396. <http://doi.org/http://dx.doi.org/10.1016/j.cagd.2014.02.011>.
- [19] MathWorks: MATLAB 2017a: Curve Fitting Toolbox User's Guide. The MathWorks Inc., 2017.
- [20] Meek, D.; Walton, D.: Approximating smooth planar curves by arc splines. *Journal of Computational and Applied Mathematics*, 59(2), 221–231, 1995. ISSN 0377-0427. [http://doi.org/10.1016/0377-0427\(94\)00029-Z](http://doi.org/10.1016/0377-0427(94)00029-Z).
- [21] Meek, D.; Walton, D.: Planar spirals that match  $G^2$  Hermite data. *Computer Aided Geometric Design*, 15(2), 103–126, 1998. ISSN 0167-8396. [http://doi.org/10.1016/S0167-8396\(97\)00020-4](http://doi.org/10.1016/S0167-8396(97)00020-4).
- [22] Meek, D.; Walton, D.: The family of biarcs that matches planar, two-point  $G^1$  hermite data. *Journal of Computational and Applied Mathematics*, 212(1), 31–45, 2008. ISSN 0377-0427. <http://doi.org/10.1016/j.cam.2006.11.018>.
- [23] Narayan, S.: Approximating cornu spirals by arc splines. *Journal of Computational and Applied Mathematics*, 255(C), 789–804, 2014. ISSN 0377-0427. <http://doi.org/10.1016/j.cam.2013.06.038>.
- [24] Park, H.: Optimal single biarc fitting and its applications. *Computer-Aided Design and Applications*, 1(1–4), 187–195, 2004. <http://doi.org/10.1080/16864360.2004.10738258>.
- [25] Piegl, L.; Tiller, W.: Data approximation using biarcs. *Engineering with Computers*, 18(1), 59–65, 2002. ISSN 1435-5663. <http://doi.org/10.1007/s003660200005>.
- [26] Piegl, L.A.; Tiller, W.: Biarc approximation of nurbs curves. *Computer-Aided Design*, 34(11), 807–814, 2002. ISSN 0010-4485. [http://doi.org/10.1016/S0010-4485\(01\)00160-9](http://doi.org/10.1016/S0010-4485(01)00160-9).
- [27] Sabin, M.: *The Use of Piecewise Forms for the Numerical Representation of Shape*. Computer & Automation Institute, Hungarian Academy of Sciences, 1976. ISBN 9789633110355.
- [28] Saini, D.; Kumar, S.; Gulati, T.R.: Reconstruction of free-form space curves using nurbs-snakes and a quadratic programming approach. *Computer Aided Geometric Design*, 33, 30–45, 2015. ISSN 0167-8396. <http://doi.org/10.1016/j.cagd.2015.01.001>.
- [29] Stewart, G.W.: On the continuity of the generalized inverse. *SIAM Journal on Applied Mathematics*, 17, 33–45, 1969.
- [30] Stoer, J.: Curve fitting with clothoidal splines. National Bureau of Standards. *Journal of Research*, 87(4), 317–346, 1982. ISSN 0022-4340. <http://doi.org/10.6028/jres.087.021>.
- [31] Šir, Z.; Feichtinger, R.; Jüttler, B.: Approximating curves and their offsets using biarcs and pythagorean hodograph quintics. *Computer Aided Geometric Design*, 38(6), 608–618, 2006. <http://doi.org/10.1016/j.cad.2006.02.003>.
- [32] Walton, D.J.; Meek, D.S.:  $G^1$  interpolation with a single Cornu spiral segment. *Journal of Computational and Applied Mathematics*, 223(1), 86–96, 2009. ISSN 0377-0427. <http://doi.org/10.1016/j.cam.2007.12.022>.
- [33] Yang, X.; Chen, Z.C.: A practicable approach to  $G^1$  biarc approximations for making accurate, smooth and non-gouged profile features in CNC contouring. *Computer Aided Geometric Design*, 38(11), 1205–1213, 2006. ISSN 0010-4485. <http://doi.org/10.1016/j.cad.2006.07.006>.
- [34] Yang, X.; Wang, G.: Planar point set fairing and fitting by arc splines. *Computer-Aided Design*, 33(1), 35–43, 2001. ISSN 0010-4485. [http://doi.org/10.1016/S0010-4485\(00\)00059-2](http://doi.org/10.1016/S0010-4485(00)00059-2).
- [35] Zheng, J.:  $C^1$  NURBS representations of  $G^1$  composite rational bézier curves. *Computing*, 86(2), 257, 2009. ISSN 1436-5057. <http://doi.org/10.1007/s00607-009-0057-4>.