Journal of Computational and Applied Mathematics 273 (2015) 1–12

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

journal homepage: www.elsevier.com/locate/cam

Efficient robust approximation of the generalised Cornu spiral

Benjamin Cross, Robert J. Cripps *

Geometric Modelling Group, School of Mechanical Engineering, University of Birmingham, Birmingham, B15 2TT, UK

ARTICLE INFO

Article history:
Received 21 November 2012
Received in revised form 29 March 2014

Keywords:
Approximation
Generalised Cornu spiral
Bezier
CAD

ABSTRACT

A Generalised Cornu Spiral (GCS) is a planar curve defined to have a monotonic rational linear curvature profile and as such these curves are considered fair. However, their implementation in current CAD systems is not straight forward partly due to not being in the usual polynomial form. A GCS cannot be expressed exactly using a finite polynomial and so a compromise can be achieved by instead approximating the GCS with a suitable polynomial.

An efficient robust approximation of the GCS using quintic polynomials is presented. The approximation satisfies the G^2 continuity conditions at the end points and the remaining four degrees of freedom are argued for by looking at G^3 approximations. The method begins by reparameterising the GCS in terms of more intuitive geometric descriptions; the winding angle, change in curvature and a shape factor. The G^3 approximations provide insight to help define values for the free parameters, and the new geometric form allows for the shortcomings in the G^3 approximations to be controlled.

The efficiency of the approximation is improved compared to earlier methods which required a numerical search. Also, there is strong evidence that the method guarantees a satisfactory approximation when the GCS lies within certain identified bounds.

1. Introduction

A Generalised Cornu Spiral (GCS) is a planar curve defined to have a monotonic rational linear curvature profile [1] and as such these curves are considered fair [2]. Fair curves are useful within CAD, as they provide the aesthetics designers require [3]. Controlling curvature, as the second derivative with respect to (w.r.t.) arc length, has practical uses corresponding to physical properties; such as unilateral force experienced when travelling on a path with constant speed. GCS curves are therefore useful in span generation such as transition curves between two data points [1]. This is particularly true for the Cornu spiral, which is itself a member of the GCS family, having applications in highway design, robotics and roller coaster design [4,5].

In order to utilise a GCS curve within CAD the coordinates for an arbitrary point must be calculated. These points are found by first integrating the curvature, then integrating these functions from within the sin/cos functions, both w.r.t. arc length, allowing for initial conditions. These integrals contain non-fundamental functions, such as the Fresnel integrals in the case of the Cornu spiral, and thus are usually obtained by numerical integration [1]. As a consequence this representation is impractical for direct use within CAD.

A more widely used representation for CAD based curves is a polynomial, such as a B-spline or a Bézier curve. The problem with these representations is that the curvature is hard to control and hence fairness is difficult to obtain [6]. A GCS cannot

* Corresponding author. Tel.: +44 121 414 4223; fax: +44 121 414 3515.
E-mail address: r.cripps@bham.ac.uk (R.J. Cripps).

http://dx.doi.org/10.1016/j.cam.2014.05.021
0377-0427/

be expressed exactly using a finite polynomial [7] and so a compromise can be achieved by instead approximating the GCS with a suitable polynomial.

The best approximations should mimic the property of rational linear curvature. This suggests that rather than using traditional measures of error, such as Hausdorff distance, the curvature profiles of the GCS and the approximation should be compared; as proposed in [8].

A well known approximation method uses the Hermite spline [9] which interpolates start and end kth derivative data with the unique polynomial of order $(2k+1)$. However, in order to obtain a satisfactory approximation, k may be too large for practical use.

A different approach was proposed in [8] wherein the degree of the polynomial was restricted to quintic. Position, tangent direction and curvature values were interpolated at the start and end points leaving four degrees of freedom. These four parameters were then varied in a search until the approximation was deemed satisfactory. This occurred when the relative curvature difference between the curves was less than some tolerance.

A similar approach was taken in [10] also using quintic curves. Two free parameters from the Hermite approximation, relating to end tangent magnitudes, were discovered. Values for these parameters were then determined by minimising the energy, that is the integral of the curvature squared w.r.t. arc length, using an optimisation procedure. The problem with both of these techniques is that the numerical search and the optimisation procedure are computationally expensive. Each configuration required the quintic polynomial to be reparameterised with respect to arc length to enable comparison of curvature values or energy.

The method of [8] was improved in [6] by insisting that the start and end points obey the G^3 constraints reducing the degrees of freedom from four to two. Approximations are formed using initial values, corresponding to matching end tangent magnitudes, and often produce satisfactory approximations. However, when the initial values for these parameters produce unsatisfactory approximations, different values are determined using a numerical search. The reduced degrees of freedom as well as understanding of the search domain produces a search routine more efficient than the one in [8].

A weakness of the approximation techniques in [8,10,6] is that they are *non-deterministic* in the sense that they require a numerical search. Any search based solutions raise concerns with efficiency and robustness. They can be considered inefficient as they consume relatively large computational overheads. Furthermore, it is not possible to guarantee that after performing these computations a satisfactory solution will be found. For this reason they cannot be considered robust. This paper addresses these concerns by presenting a method such that given suitable bounds on a GCS, an acceptable approximation can be found without the need for a search.

It is the intention of this research to eventually be able to produce an efficient polynomial curve construction algorithm that yields curves with high-quality shape characteristics based upon geometric data. While other high-quality curves described directly via their curvature profile do exist, such as polynomial functions of arc-length [11], deterministic algorithms to convert these curves into a polynomial form do not. Furthermore, the GCS is considered a suitable candidate from this family since existing research has already shown that a GCS can be fitted to G^2 data within a reasonable tolerance [1]. An initial approach would be to fit a GCS to some data and then form the polynomial approximation. This process of finding the GCS satisfying given geometric data is currently a numerically expensive process. However, it may be possible to *reverse engineer* the GCS approximation from the data without having to find the specific GCS. This idea is discussed in more detail in the conclusion.

In the next two sections some background information and preliminaries are discussed. Using this information an approximation method is described in section four. Analysis and examples are discussed in section five and six respectively which is then followed in section seven by some concluding remarks including possible improvements and future research.

2. Background information

2.1. The GCS

GCS curves are defined to have a rational linear monotonic curvature profile. Explicitly, the curvature profile of a GCS has the form:

$$\kappa(s) = \frac{\kappa_0 S + (\kappa_1 - \kappa_0 + r\kappa_1)s}{S + rs} \qquad s \in [0, S],\ \ r \in (-1, \infty), \tag{1}$$

where the parameter s represents the arc length of the curve, κ_0 and κ_1 represent the start and end curvature values respectively, S corresponds to the length of the GCS and r, $(r > -1)$, is referred to as the shape factor.

The curvature profile however, only describes the shape of the curve. Translations or rotations do not affect the curvature. Therefore, in order to completely define the curve in $\mathbb{R}^2$, its initial location, (x_0, y_0), and orientation, θ_0, must be defined [12]. The orientation is decided by the angle, θ_0, that the initial tangent vector makes with the positive x-axis.

Scaling of the curve can also be accounted for. Applying a scaling factor λ, new values for the parameters can be calculated as: $S^* = \lambda S$, $\kappa_0^* = \frac{\kappa_0}{\lambda}$, $\kappa_1^* = \frac{\kappa_1}{\lambda}$ and $r^* = r$ [12].