Faculty of Sciences and Mathematics, University of Niš, Serbia
Available at: http://www.pmf.ni.ac.yu/filomat

Filomat **23:2** (2009), 108–116

CURVEBEND GRAPHICAL TOOL FOR PRESENTATION OF INFINITESIMAL BENDING OF CURVES

Svetozar R. Rančić, Ljubica S. Velimirović and Milan Lj. Zlatanović

Abstract

An infinitesimal bending of the curve at E^3 is considered and the infinitesimal bending field is determined and discussed. CurveBend, tool for graphical presentation of non rigid curves is presented. Influence of infinitesimal bending field on curves is discussed and visualized by the tool.

1 Introduction

Infinitesimal bending of surfaces and curves is a part of the more general bending theory, which presents one of the main consisting parts of the global differential geometry. A concept of infinitesimal deformation dealt first with infinitesimal deformation of surfaces and then with the same problem at the theory of curves and manifolds.

Under bending surface is included in continuous family of isometrical surfaces, so that the curve preserves its arc length and the angles are also preserved. It is known that two surfaces are trivially isometrical if we get them one from another by rigid motion or by plane symmetry (or by finite number of such transformations). A surface is uniquely defined if there are only trivially isometrical surfaces. Each uniquely defined surface is rigid in a sense of isometrical bending (as there are not isometrical surfaces bent from initial).

2000 *Mathematics Subject Classification.* 53A05, 53C45, 68U05.

Key words and phrases. Infinitesimal bending, infinitesimal deformation, OpenGL, C++

The second and the third author were supported by Project 144032D MNTR Serbia

On the other hand, infinitesimal bending of surfaces is not an isometric deformation, or roughly speaking it is with appropriate precision. Arc length is stationary under infinitesimal bending.

The theory of infinitesimal deformation has numerous applications in mathematics and mechanics (rigidity of shells). First results of the infinitesimal bending on non-convex surfaces belong to H. Liebman [9], [10]. He has proved that the torus and analytic surfaces containing the convex strip are rigid in a sense of infinitesimal bending. Later Efimov at [8] has given condition for $z(u)$ to be an infinitesimal bending field of a regular curve.

Computer graphic is rapidly developing area following and inspiring fast growth in computing power. Nowadays there are many scientific and industrial areas which use computer programs based on computer graphics. Infinitesimal bending of curves and surfaces has a lack of specially developed and oriented programming tools in this area, even through we can find in articles some graphically presented examples of flexible curves and surfaces.

Such tool has to fulfill some requirements and compose into a whole different area in computer science together with mathematical theory of infinitesimal bending. Tool aimed for graphical presentation of curves and their infinitesimal bent shapes needs basic numeric and symbolic calculation ability. It is obvious for parametric defined curves, defining functions incorporated in bending and checking correctness of definitions. Symbolic differentiation is also a requisite. Obtained curves are drawn for parameter values in some interval supplied by the user. Tool incorporates ability for numerical calculations of points belonging to curves. Drawing initial curves and their infinitesimally bent shapes as 3D objects should use some graphic library and we use OpenGL as industrial standard. Incorporation of OpenGL gives fast drawing capability and ability to interactively examine obtained 3D object. Tool is developed in C++ under *Microsoft Windows* platform and gives high level of interactive examination.

2 Preliminaries

Infinitesimal bending of surfaces and manifolds was widely studied in [8], [11], [12], [19], [20]. Infinitesimal bending of curves at E^3 was studied at [8], [20], [21] and [22]. This work presents a follow up of the results given at [20].

At the beginning we are giving some basic facts, definitions and theorems discussed at the [8] and [20].

Definition 2.1 *Let us consider a closed regular curve*

$$C : \mathbf{r} = \mathbf{r}(u), \tag{1}$$

included in a family of the curves

$$C_\varepsilon : \mathbf{r}_\varepsilon = \mathbf{r}(u) + \varepsilon \mathbf{z}(u),\ (\varepsilon \geq 0,\ \varepsilon \to 0,\ \varepsilon \in \Re) \tag{2}$$

where u is a real parameter and we get C for $\varepsilon = 0$ $(C \equiv C_0)$. Family of curves C_ε is **infinitesimal bending of a curve** *C if*

$$ds_\varepsilon^2 - ds^2 = o(\varepsilon), \tag{3}$$

where $\mathbf{z} = \mathbf{z}(u)$ is **infinitesimal bending field** *of the curve C.*

Theorem 2.1 *[8] Necessary and sufficient condition for $\mathbf{z}(\mathbf{u})$ to be an infinitesimal bending field of a curve C is*

$$d\mathbf{r} \cdot d\mathbf{z} = 0. \qquad \square$$

The next theorem is related to determination of the infinitesimal bending field of a curve C.

Theorem 2.2 *[20] The infinitesimal bending field for the curve C (1) is*

$$\mathbf{z}(u) = \int [p(u)\mathbf{n}(u) + q(u)\mathbf{b}(u)]du + const, \tag{4}$$

where $p(u), q(u)$, are arbitrary integrable functions, and the vectors $\mathbf{n}(u),\ \mathbf{b}(u)$ are respectively unit principal normal and binormal vector field of a curve C. $\square$

Having in mind that unit binormal and normal field of the curve (1) can be written in the form

$$\mathbf{b} = \frac{\dot{\mathbf{r}} \times \ddot{\mathbf{r}}}{|\dot{\mathbf{r}} \times \ddot{\mathbf{r}}|}, \quad \mathbf{n} = \frac{(\dot{\mathbf{r}} \cdot \dot{\mathbf{r}})\ddot{\mathbf{r}} - (\dot{\mathbf{r}} \cdot \ddot{\mathbf{r}})\dot{\mathbf{r}}}{|\dot{\mathbf{r}}||\dot{\mathbf{r}} \times \ddot{\mathbf{r}}|}, \tag{5}$$

infinitesimal bending field can be written in the form

$$\mathbf{z}(u) = \int [p(u)\frac{(\dot{\mathbf{r}} \cdot \dot{\mathbf{r}})\ddot{\mathbf{r}} - (\dot{\mathbf{r}} \cdot \ddot{\mathbf{r}})\dot{\mathbf{r}}}{|\dot{\mathbf{r}}||\dot{\mathbf{r}} \times \ddot{\mathbf{r}}|} + q(u)\frac{\dot{\mathbf{r}} \times \ddot{\mathbf{r}}}{|\dot{\mathbf{r}} \times \ddot{\mathbf{r}}|}]du,$$

where $p(u), q(u)$ are arbitrary integrable functions, or in the form

$$\mathbf{z}(u) = \int [P_1(u)\dot{\mathbf{r}} + P_2(u)\ddot{\mathbf{r}} + Q(u)(\dot{\mathbf{r}} \times \ddot{\mathbf{r}})]du \tag{6}$$

where $P_i(u),\ i = 1, 2,\ Q(u)$ are arbitrary integrable functions, too.

Remark 2.1 *Infinitesimal deformations of special kind where considered at [17].*

3 CurveBend

It is interesting to see influence of infinitesimal bending field on flexible curves and surfaces and their corresponding bent shapes. CurveBend is our visualization tool devoted to visual representation of infinitesimally bent curves. We have previously developed the tool named SurfBend, aimed to create 3D presentation and visualize application of infinitesimal bending on flexible torus like surfaces. It was partially presented at the ESI Conference Rigidity and Flexibility, Viena, 2006 [23]. Those rotational surfaces were obtained by revolution of a meridian in the shape of polygon. It was also able to show circles formed by apices of polygon and its infinitesimally deformed shape, as well as, to visually present surfaces created during such deformation [16].

We have moved our research further and added subsystem named Curve Bend, purposely to widen application of infinitesimal bending to a class of non rigid curves, both planar and spatial. Spatial curves laying on some well known surfaces are also examined. Our goals are to create an easy to use tool for:

- definition of curves and deformation given by (4). The ability to symbolically define curve C, also functions $\mathbf{z}$, p and q is given;
- visual presentation which incorporate quick basic and 3D calculations. It is very useful and illustrative to interactively examine bent curves and obtained surfaces and the influence of infinitesimal bending fields on them.

CurveBend is developed in Object Oriented language C++. It uses explicitly defined functions with n independent variables. It implements parse once-evaluate many times type of parsing for mathematical expressions given as strings. This mathematical expression parser component parses and evaluates a mathematical expression that may contain variables, constants and functions over a set of elementary functions. To be efficient in repeated calculations, parser creates an expression tree at first and reuses this expression tree for each evaluation without the need to reparse. The expression tree is optimized by calculating constant expression sections at once so that further evaluation requests will be quicker.

Each of elementary functions is wrapped by appropriate class, and we created a class hierarchy to support building a tree structure as an expression tree of the function. Every class in the hierarchy has overridden abstract members:

```
double evaluate (double * arguments);
Function * derive(int argNumber);
```

Function `evaluate` is designed to calculate the value of wrapped elementary function and as an argument it takes an array with double values of n parameters for which we want to calculate function value.

Function `derive` is designed to build a new tree structure according to the derivation rules for elementary function, widened with rules for composite functions, having as argument the ordinal of i-th independent variable on which the partial derivation is wanted. The return value is also of the `Function *` type, so we want to point that this enables us to build the expression tree structure for arbitrary order partial derivation, and also calculate values of the obtained function by calling `evaluate` member function.

The starting point is explicitly defined function entered as input string, then parse it and check its consistency. We use formal parsing techniques, that includes the grammar describing such functions. The grammar, we have used, can be found in [2] and as parsing tool we have used GOLD Parser [4]. GOLD Parser is a free, multi language, pseudo open source parsing system that can be used to develop programming languages, scripting languages and interpreters. After parsing we build an internal tree structure - an expression tree of the function as described in Object Oriented design pattern Composite [5]:

```
MainFunction * pF;
if( ManagerFunction::parse( string functionInscription )
    pF = ManagerFunction::build(string functionInscription );
```

We also used famous OO patterns Singleton, Abstract Factory [5] in producing function objects and building trees and evaluating functions.

According to theorems 2.1 and 2.2 we consider infinitesimal bending fields $\mathbf{z}$ for closed curves and values of parameter $u \in [0, 2\pi]$. If we calculate definite integral

$$\mathbf{z}(a) - \mathbf{z}(0) = \int_0^a [p(u)\mathbf{n}(u) + q(u)\mathbf{b}(u)]du, \tag{7}$$

we have

$$\mathbf{z}(a) = \int_0^a [p(u)\mathbf{n}(u) + q(u)\mathbf{b}(u)]du + \mathbf{z}(0). \tag{8}$$

Additive constant *const* mentioned in (4) is here $\mathbf{z}(0)$. Having in mind that we can find partial derivatives, and calculate their values for passed arguments, we can apply numerical methods for calculation (7) to produce values of bending field on discrete division points $a \in [0, 2\pi]$. We use well known generalized Simpson formula [15] for numerical integration, and a user is able to supply a number of division points, as well as, a number of points inside every segment bounded by successive division points. Further, we can calculate values of the supplied curve C in division points and knowing ε we have points of bent curve

$$C_\varepsilon : \mathbf{r}_\varepsilon = \mathbf{r}(\theta) + \varepsilon \mathbf{z}(\theta), \quad \theta \in [0, 2\pi], \tag{9}$$

Visualization of bent curves C_ε is obtained using OpenGL [14], [6] standard. It should therefore be portable, although it has only been tested on *Microsoft Windows* platform. Rising control to interactive level has been done using *MFC* [18]. We are able to rotate a 3D object and see it from different angles and points of view. It is possible to use sliders, to easily, interactively, adjust important parameters for bending calculations like ε, number of segments and number of inner points inside segments for numerical integration, as well as, additive constants in each of three dimension. During deformation, curves describe surfaces and we are able to give visualization of such surfaces in fill or wire mode with ability to adjust semi transparency of hidden lines.

CurveBend is a free software and is available from
http //www.pmf.ni.ac.yu/pmf/licne_prezentacije/103/software.php

The following examples are obtained using our visualization tool CurveBend.

4 Examples

Infinitesimal bending of curves suppose that deformations are small and can not be seen by naked eye. To make its visible, in following examples, we will take much larger values for parameter ε.

Example 4.1 Let us have a curve given by parametric equation

$$C: \ \mathbf{r}(u) = \big(8cos(u) + sin(2u), 8sin(u) + cos(2u), 0\big).$$

We choose infinitesimal bending field $z(u)$ given by (7) where

$$p(u) = 0, \quad q(u) = cos(2u).$$

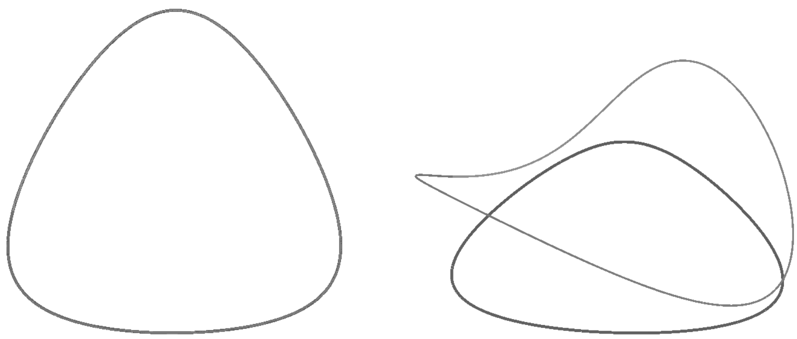

Figure 4.1.

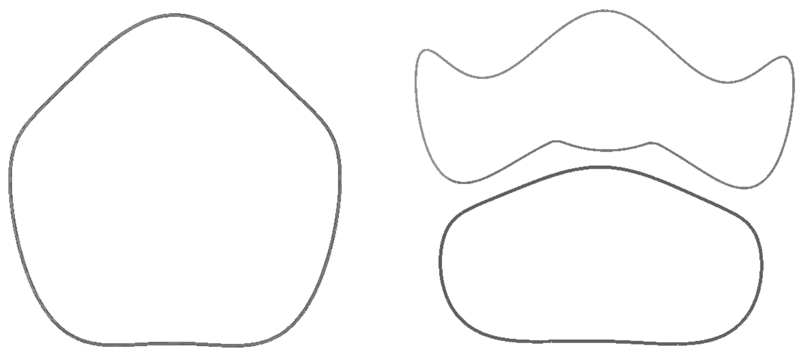

Figure 4.2.

The Figure 4.1. presents both initial curve C and deformed curve C_ε, $\varepsilon = 2.60$ for $u \in [0, 2\pi]$ in this case.

Example 4.2 Let us have a next curve given by parametric equation

$$C: \ \mathbf{r}(u) = \big(4cos(u) + 0.25cos(4u), 4sin(u) + 0.25cos(4u), 0\big).$$

We choose infinitesimal bending field $z(u)$ given by (7) where

$$p(u) = 0, \quad q(u) = sin(4u).$$

The Figure 4.2. presents both initial curve C and deformed curve C_ε, $\varepsilon = 2.50$ for $u \in [0, 2\pi]$ in this case.

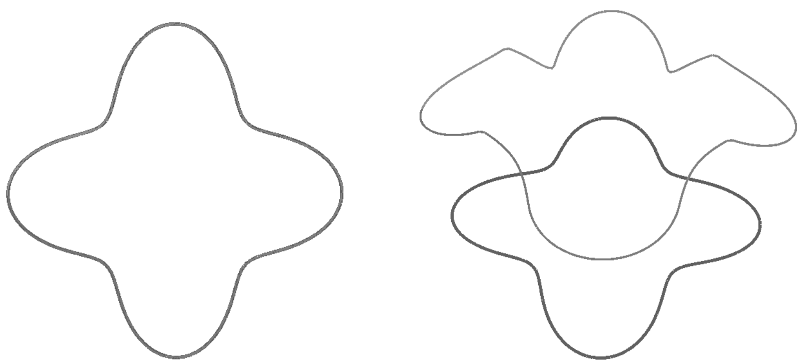

Figure 4.3.

Example 4.3 Let us have an closed curve given by parametric equation

$$C: \mathbf{r}(u) = \big((4 + cos(4u))cos(u), (4 + cos(4u))sin(u), 0\big).$$

We choose infinitesimal bending field $z(u)$ given by (7) where

$$p(u) = 0, \quad q(u) = cos(u)sin(u).$$

The Figure 4.3. presents both initial curve C and deformed curve C_ε, $\varepsilon = 4.0$ for $u \in [0, 2\pi]$ in this case.

References

[1] Aleksandrov, A. D., *O beskonechno malyh izgibaniyah neregulyarnyh poverhnostei* Matem. sbornik, 1(43), 3 (1936) 307-321.

[2] Ćirić M., Rančić S. R., *Parsing in different languages,* Facta Universitatis (Niš), Ser. Elec. Energ. 18 (2) (2005), 299-307.

[3] Cohn-Vossen, S., *Unstarre geschlossene Flachen,* Math. Ann., 102 (1930) 10-29.

[4] Cook, D., GOLD Parser Builder, www.devincook.com/goldparser.

[5] Gamma, E., Helm, R., Johnston, R. and Vlisides, J., *Design Patterns - Elements of Reusable Object-Oriented Software,* Addison-Wesley, 1995.

[6] Glasser, G., Stachel, H., *Open Geometry: OpenGL + Advanced Geometry,* Springer, 1999.

[7] Gray, A., *Modern differential geometry of curves and surfaces with Mathematica,* CRC Press, Boca Raton, 1998.

[8] Efimov, N. V., *Kachestvennye voprosy teorii deformacii poverhnostei,* UMN **3.2** (1948) 47-158.

[9] Liebman, H., *Uber die Verbiegung von Ringflache*, Gottinger Nachr. (1901) 39-53

[10] Liebman, H., *Uber die Verbiegung von Rotationsflachen,* Leipzig Ber. 53 (1901) 215-234.

[11] Ivanova Karatopraklieva, I., Sabitov, I Kh., *Surfaces deformation,* J. Math. Sci., New Yourk 70 $N^o 3$ (1995) 1685-1716.

[12] Ivanova Karatopraklieva, I., Sabitov, I Kh., *Bending of surfaces II,* J. Math. Sci., New Yourk 74 $N^o 3$ (1995) 997-1043.

[13] Kon-Fossen, S. E., *Nekotorye voprosy differ. geometrii v celom,* Fizmatgiz, Moskva **9** (1959).

[14] McReynolds, T., Blythe, D., *Advanced Graphics Programming Using OpenGL*, Morgan Kaufmann publishers , 2005.

[15] Milovanović G., V., *Numerička analiza,*Naučna knjiga, Beograd, 1985.

[16] Rančić, S. R.; Velimirović, L. S., *Visualization of infinitesimal bending of some class of toroid,* International Journal of Pure and Applied Mathematics, 2008; 42 $N^o 4$, 507-514.

[17] Sabitov, I Kh., *Isometric transformations of a surface inducing conformal maps of the surface onto itself,*Mat. Sb., (1998) 189:1, 119132.

[18] Shepherd, G., Kruglinski, D., *Programming with Microsoft Visual C++.NET*, 6th ed. Microsoft Press , (2003).

[19] Velimirović, L. S., *Change of area under infinitesimal bending of border curve,* Buletins for Applied Mathematics (BAM) Hungary PC-129 (2000).

[20] Velimirović, L. S., *Change of geometric magnitudes under infinitesimal bending,* Facta Universitatis - Series: Mechanics, Automatic Control and Robotics Vol.3, No 11, (135-148).

[21] Velimirović, L. S., *Infinitesimal bending of curves,* Matematicki bilten Skopje, Makedonija 25(LI), (25-36).

[22] Velimirović, L. S., Rančić S. R., Zlatanović, M. L., *Graphical presentation of infinitesimal bending of curves,* Proceedings of 24th nacional and 1st international scientific conference, MonGeometrija, Vrnjačka banja, september 25th - 27th 2008, (383-393).

[23] Velimirović, L. S., Rančić S. R., *Higher order infinitesimal bending of a class of toroids,* ESI Conference Rigidity and Flexibility, Vienna, 2006, preprint.

Svetozar R. Rančić, Ljubica S. Velimirović, Milan Lj. Zlatanović,
University of Niš, Faculty of Science and Mathematics,
E-mails: `rancicsv@yahoo.com, vljubica@pmf.ni.ac.rs, zlatmilan@yahoo.com`