Weighted G^1-Multi-Degree Reduction of Bézier Curves

Abedallah Rababah
Department of Mathematics,
Jordan University of Science and Technology
Irbid 22110 Jordan

Salisu Ibrahim
Department of Mathematics,
Jordan University of Science and Technology
Irbid 22110 Jordan

***Abstract*—In this paper, weighted G^1-multi-degree reduction of Bézier curves is considered. The degree reduction of a given Bézier curve of degree n is used to write it as a Bézier curve of degree $m, m < n$. Exact degree reduction is not possible, and, therefore, approximation methods are used. The weight function $w(t) = 2t(1-t),\ t \in [0,1]$ is used with the L_2-norm in multi-degree reduction with G^1-continuity at the end points of the curve. Since we consider boundary conditions this weight function improves approximation in the middle of the curve. Numerical results and comparisons show that the proposed method produces fewer errors and outperform all existing methods.**

***Keywords*—Bézier curves; multiple degree reduction; G^1-continuity; geometric continuity**

I. Introduction

Degree reduction of Bézier curves and surfaces is an important issue in Computer Aided Geometric Design (CAGD) that is tackled by many researchers. It facilitates data exchange, compression, transfer, and comparison. In degree reduction, we approximate a Bézier curve of degree n by a Bézier curve of degree $m, m < n$; moreover, the boundary conditions have to be satisfied and gives minimum error. The struggles of finding a solution are disturbed by the requirement of solving a non-linear problem, in which numerical methods have to be used. In 2000, J. Peter and U. Reif proved in [5] that degree reduction of Bézier curves in the L_2 norm equals best Euclidean approximation of Bézier points. These results are generalized to the constrained case by Ahn et. al. in [1], and discrete cases have been studied in [2], [8].
The existing methods to find degree reduction have many issues including accumulate round-off errors, stability issues, complexity, accuracy, losing conjugacy, requiring the search direction to be set to the steepest descent direction frequently, experiencing ill-conditioned systems, leading to a singularity, and the most challenging difficulty is in applying the methods (difficulty and indirect). A. Rababah and S. Mann presented in [10] a method to find the G^2-degree reduction for Bézier Curves based on exploiting additional parameters as in [7]. These results are expressive to researchers as well as to industrial practitioners. Their examples show that the C^2 method fails to reproduce the inner loop of the heart, while their C^1/G^2 method reproduces the loop and provides a better approximation elsewhere along the curve.

In all existing degree reducing methods, the conditions and free parameters were applied at the end points. So, there is a need to better approximate those parts close to the centre of the curve. In this paper, we introduce a weight to take care of the centre of the curve, it is appropriate to consider degree reduction with the weight function $w[t] = 2t(1-t)$, $t \in [0,1]$. The examples show that the proposed methods provide better approximation at the centre of the curves with minimum error and also reproduced these loops correctly better than existing methods.

II. Preliminaries

A Bézier curve $P_n(t)$ of degree n is defined algebraically as follows:

$$P_n(t) = \sum_{i=0}^{n} p_i B_i^n(t), \qquad 0 \le t \le 1, \tag{1}$$

where

$$B_i^n(t) = \binom{n}{i}(1-t)^{n-i}t^i, \quad i = 0, 1, \ldots, n,$$

are the Bernstein polynomials of degree n, and $p_0, p_1, \ldots, p_n$ are called the Bézier control points or the Bézier points, for more see [4].

The first derivative of the Bézier curve is given by:

$$\frac{d}{dt}P_n(t) = n\sum_{i=0}^{n-1} \Delta p_i B_i^{n-1}(t),$$

where

$$\Delta p_i = p_{i+1} - p_i,\ i = 0, 1, \ldots, n-1.$$

The multiplication of two Bernstein polynomials with the weight function $w(t) = 2t(1-t)$ is given by

$$B_i^m(t)B_j^n(t)2t(1-t) = \frac{2\binom{m}{i}\binom{n}{j}}{\binom{m+n+2}{i+j+1}} B_{i+j+1}^{m+n+2}(t). \tag{2}$$

We define the Gram matrix $G_{m,n}$ as $(m+1) \times (n+1)$-matrix with weight function as follows:

$$\begin{aligned} g_{ij} &= \int_0^1 B_i^m(t)B_j^n(t)2t(1-t)dt \\ &= \frac{2\binom{m}{i}\binom{n}{j}}{(m+n+3)\binom{m+n+2}{i+j+1}}, i = 0, \ldots, m,\ j = 0, \ldots, n. \end{aligned} \tag{3}$$

The matrix $G_{m,m}$ with weight function is real, symmetric, and positive definite like the case in [10].

III. Degree Reduction of Bézier curves

Degree reduction is approximating a given Bézier curve of degree n by a Bézier curve of degree $m, m < n$. It is approximative process in nature and exact degree reduction is not possible. In this paper, our aim is to find a Bézier curve $R_m(t)$ of degree m with control points $\{r_i\}_{i=0}^m$ that approximates a given Bézier curve $P_n(t)$ of degree n with control points $\{p_i\}_{i=0}^n$, where $m < n$. The Bézier curve R_m has to satisfy the following two conditions:

1) P_n and R_m are G^1-continuous at the end points, and
2) the weighted L_2-error between P_n and R_m is minimum.

We can write the two Bézier curves $P_n(t)$ and $R_m(t)$ in matrix form as

$$P_n(t) = \sum_{i=0}^{n} p_i B_i^n(t) =: B_n P, \quad 0 \le t \le 1, \tag{4}$$

and similarly

$$R_m(t) = \sum_{i=0}^{m} r_i B_i^m(t) =: B_m R, \quad 0 \le t \le 1.$$

In the following sections we investigate the case of G^1-continuity with weighted degree reduction of Bézier curves.

IV. Weighted G^1-Degree Reduction

$P_n(t)$ and $R_m(t)$ are G^1-continuous at $t = 0, 1$ if they satisfy the following conditions

$$R_m(i) = P_n(s(i)), \quad i = 0, 1. \tag{5}$$

$$R'_m(i) = s'(i) P'_n(s(i)), \quad s'(i) > 0, \quad i = 0, 1. \tag{6}$$

This means that the two curves P_n and R_m have to have common end points

$$r_0 = p_0, \quad r_m = p_n,$$

and the direction of the tangent at the two end points of P_n and R_m should coincide, but they need not to be of equal length. To simplify the problem and have a linear system, the authors in [10] used $s'(i) = \delta_i$, for $i = 0, 1$. We analogously use these substitutions for the case of weighted degree reduction to get

$$R'_m(i) = \delta_i P'_n(i), \quad i = 0, 1. \tag{7}$$

Using $s'(i) = \delta_i$, $i = 0, 1$, we can solve (5) and (7) for the two control points at either ends of the curve to get

$$r_0 = p_0,$$

$$r_1 = p_0 + \frac{n}{m} \Delta p_0 \delta_0,$$

$$r_{m-1} = p_n - \frac{n}{m} \Delta p_{n-1} \delta_1,$$

$$r_m = p_n.$$

The points r_0, r_1, r_{m-1} and r_m are determined by G^1-continuity conditions at the boundaries; accordingly, the elements of R_m can be decomposed into two parts stated as follows. The boundaries part $R_m^c = [r_0, r_1, r_{m-1}, r_m]^t$ and the interior part with interior points $R_m^f = R_m \backslash R_m^c = [r_2, \ldots, r_{m-2}]^t$. Similarly, B_m is decomposed in the same way into B_m^c and B_m^f.

The weighted distance between P_n and R_m is measured using weighted L_2-norm; therefore, the error term becomes:

$$\begin{aligned} \varepsilon &= \int_0^1 ||B_n P_n - B_m R_m||^2 \; 2t(1-t) dt \\ &= \int_0^1 ||B_n P_n - B_m^c R_m^c - B_m^f R_m^f||^2 \; 2t(1-t) dt. \end{aligned} \tag{8}$$

Differentiating ε with respect to the unknown control points R_m^f we get

$$\frac{\partial \varepsilon}{\partial R^f} = 2 \int_0^1 ||B_n P_n - B_m^c R_m^c - B_m^f R_m^f|| \; B_m^f \; 2t(1-t) \; dt.$$

Evaluating the integral and equating to zero gives

$$\frac{\partial \varepsilon}{\partial R^f} = G_{m,n}^p P_n - G_{m,m}^c R_m^c - G_{m,m}^f R_m^f = 0, \tag{9}$$

where

$$\begin{aligned} G_{m,n}^p &:= G_{m,n}(2, \ldots, m-2; 0, 1, \ldots, n), \\ G_{m,m}^c &:= G_{m,m}(2, \ldots, m-2; 0, 1, m-1, m), \\ G_{m,m}^f &:= G_{m,m}(2, \ldots, m-2; 2, \ldots, m-2), \end{aligned}$$

and $G_{m,n}(\ldots;\ldots)$ is the sub-matrix of $G_{m,n}$ formed by the indicated rows and columns.

Differentiating ε with respect to δ_i and equating to zero gives

$$\frac{\partial \varepsilon}{\partial \delta_0} = \left(G_{m,n}^1 P_n - G_{m,m}^{1;c} R_m^c - G_{m,m}^{1;f} R_m^f \right) \cdot \Delta p_0 = 0, \tag{10}$$

$$\frac{\partial \varepsilon}{\partial \delta_1} = \left(G_{m,n}^{m-1} P_n - G_{m,m}^{m-1;c} R_m^c - G_{m,m}^{m-1;f} R_m^f \right) \cdot \Delta p_{n-1} = 0, \tag{11}$$

where for $q = 1, m-1$:

$$\begin{aligned} G_{m,n}^q &:= G_{m,n}(q; 0, 1, \ldots, n), \\ G_{m,m}^{q;c} &:= G_{m,m}(q; 0, 1, m-1, m), \\ G_{m,m}^{q;f} &:= G_{m,m}(q; 2, \ldots, m-2). \end{aligned} \tag{12}$$

Note that (9) are point valued equations while (10) and (11) are scalar valued equations, expanding (9) into its $x, y, z, \ldots$ coordinates and joining them together with (10) and (11) yields $d(m-3)+2$ equations in $d(m-3)+2$ unknowns, see Rababah-Mann [10].

In the planar case, the control points of the Bézier curve are expanded into their x and y components. Therefore, the variables of our system of equations are $r_k^x, r_k^y, k = 2, \ldots, m-2$, δ_0 and δ_1. To express the system in a clear form, we have to decompose each of r_1 and r_{m-1} into a constant part and a part involving δ_0 and δ_1, respectively. Let v_1 and v_{m-1} be the constant parts of r_1 and r_{m-1} respectively. Hence

$$v_1 \quad = p_0, \quad v_{m-1} = p_n.$$

The following vectors are defined to express the linear system in explicit form:

$$\begin{array}{rcl} P_n^C &=& [p_0^x, ..., p_n^x, p_0^y, ..., p_n^y]^t, \\ R_m^F &=& [r_2^x, \ldots, r_{m-2}^x, r_2^y, \ldots, r_{m-2}^y, \delta_0, \delta_1]^t, \\ R_m^C &=& [r_0^x,\ v_1^x,\ v_{m-1}^x,\ r_m^x,\ \ r_0^y,\ v_1^y,\ v_{m-1}^y,\ r_m^y]^t. \end{array}$$

Let $\oplus$ be the direct sum. Define the matrices

$$\begin{array}{rcl} G_{m,n}^{p+} &=& G_{m,n}^p \oplus G_{m,n}^p, \\ G_{m,m}^{c+} &=& G_{m,m}^c \oplus G_{m,m}^c, \\ G_{m,m}^{f+} &=& G_{m,m}^f \oplus G_{m,m}^f. \end{array} \tag{13}$$

The Gram matrix $G_{m,m}^{f+}$ has the same properties of the matrix $G_{m,m}^f$.
Write $G := G_{m,m}$ and define

$$C = \begin{bmatrix} \Delta p_0 \Delta p_0 G(1,1) & \Delta p_0 \Delta p_{n-1} G(1, m-1) \\ \Delta p_0 \Delta p_{n-1} G(m-1,1) & \Delta p_{n-1} \Delta p_{n-1} G(m-1, m-1) \end{bmatrix},$$

$$= \begin{bmatrix} \Delta p_0 & 0 \\ 0 & \Delta p_{n-1} \end{bmatrix} \begin{bmatrix} G(1,1) & G(1, m-1) \\ G(m-1,1) & G(m-1, m-1) \end{bmatrix} \times \begin{bmatrix} \Delta p_0 & 0 \\ 0 & \Delta p_{n-1} \end{bmatrix}.$$

Further define $L_{m,n}, L_{m,m}^c, L_{m,m}^f$ as

$$L_{m,n} = \begin{bmatrix} G_{m,n}^1 \Delta p_0^x & G_{m,n}^1 \Delta p_0^y \\ G_{m,n}^{m-1} \Delta p_{n-1}^x & G_{m,n}^{m-1} \Delta p_{n-1}^y \end{bmatrix},$$
$$L_{m,m}^c = \begin{bmatrix} G_{m,m}^{1;c} \Delta p_0^x & G_{m,m}^{1;c} \Delta p_0^y \\ G_{m,m}^{m-1;c} \Delta p_{n-1}^x & G_{m,m}^{m-1;c} \Delta p_{n-1}^y \end{bmatrix},$$
$$L_{m,m}^f = \begin{bmatrix} G_{m,m}^{1;f} \Delta p_0^x & G_{m,m}^{1;f} \Delta p_0^y \\ G_{m,m}^{m-1;f} \Delta p_{n-1}^x & G_{m,m}^{m-1;f} \Delta p_{n-1}^y \end{bmatrix},$$

Further define $L_{m,n}, L_{m,n}^f$ as

$$L_{m,n} = \begin{bmatrix} G_{m,n}^2 \Delta p_0^x & G_{m,n}^2 \Delta p_0^y \\ G_{m,n}^{m-2} \Delta p_{n-1}^x & G_{m,n}^{m-2} \Delta p_{n-1}^y \end{bmatrix},$$
$$L_{m,n}^f = \begin{bmatrix} G_{m,n}^{c;2} \Delta p_0^x & G_{m,n}^{c;2} \Delta p_0^y \\ G_{m,n}^{c;m-2} \Delta p_{n-1}^x & G_{m,n}^{c;m-2} \Delta p_{n-1}^y \end{bmatrix},$$

where $G_{m,n}^q, G_{m,m}^{q;c}$, and $G_{m,n}^{q;f}$ are defined in (12). The matrices C, $L_{m,n}$, $L_{m,m}^c$, and $L_{m,m}^f$ are obtained from (10) and (11), (the derivatives with respect to the δ_is).

The coordinate form of the expansion of (9) becomes

$$G_{m,m}^F R_m^F = G_{m,n}^{PC} P_n^C - G_{m,m}^C R_m^C, \tag{14}$$

where

$$G_{m,n}^{PC} = \begin{bmatrix} G_{m,n}^{p+} \\ L_{m,n} \end{bmatrix},$$
$$G_{m,m}^C = \begin{bmatrix} G_{m,m}^{c+} \\ L_{m,m}^c \end{bmatrix},$$
$$G_{m,m}^F = \begin{bmatrix} G_{m,m}^{f+} & \frac{n}{m}(L_{m,m}^f)^t \\ L_{m,m}^f & \frac{n}{m} C \end{bmatrix}.$$

From (14) and because $G_{m,m}^F$ is invertible, we can find our unknowns as

$$R_m^F = (G_{m,m}^F)^{-1} \Big(G_{m,n}^{PC} P_n^C - G_{m,m}^C R_m^C \Big). \tag{15}$$

V. Applications

In this section, some examples are given to illustrate the effectiveness of the proposed method of weighted G^1-degree reduction. Comparisons with other existing methods are also presented in this section.

Example 1: Given the Bézier curve (spiral) $P_n(t)$ of degree 19 with the control points, see Fig. 11 in [10]:
$\mathbf{P}_0 = (37, 38)$, $\mathbf{P}_1 = (43, 37)$, $\mathbf{P}_2 = (39, 27)$, $\mathbf{P}_3 = (29, 26)$, $\mathbf{P}_4 = (23, 36)$,
$\mathbf{P}_5 = (26, 50)$, $\mathbf{P}_6 = (45, 56)$, $\mathbf{P}_7 = (58, 47)$, $\mathbf{P}_8 = (58, 29)$, $\mathbf{P}_9 = (46, 14)$,
$\mathbf{P}_{10} = (26, 6)$, $\mathbf{P}_{11} = (5, 15)$, $\mathbf{P}_{12} = (0, 40)$, $\mathbf{P}_{13} = (3, 58)$, $\mathbf{P}_{14} = (24, 68)$,
$\mathbf{P}_{15} = (50, 75)$, $\mathbf{P}_{16} = (79, 69)$, $\mathbf{P}_{17} = (79, 36)$, $\mathbf{P}_{18} = (65, 12)$, $\mathbf{P}_{19} = (50, 0)$,

It is reduced to Bézier curve $R_m(t)$ of degree 8. Fig. 1 depicts the original curve in solid-blue and weighted G^1-degree reduction in dashed-red curve. Fig. 2 shows the curves with control polygons; original curve (dashed-Black); weighted G^1 (dashed-Green). Fig. 3 shows the error plots for weighted G^1-degree reduction in Example 1.

Example 2: Given the Bézier curve $P_n(t)$ of degree 10 with the control points, see [6]:
$\mathbf{P}_0 = (0, 1.2)$, $\mathbf{P}_1 = (0.04, 0.6)$, $\mathbf{P}_2 = (0.15473790322581, 0.507)$,
$\mathbf{P}_3 = (0.32207661290323, 0.878)$, $\mathbf{P}_4 = (0.30897177419355, 0.086)$,
$\mathbf{P}_5 = (0.51864919354839, 0)$, $\mathbf{P}_6 = (0.62449596774194, 0.8)$, $\mathbf{P}_7 = (0.89, 0.874)$,
$\mathbf{P}_8 = (0.92, 0.6)$, $\mathbf{P}_9 = (0.92, 0.3)$, $\mathbf{P}_{10} = (0.75352822580645, 0)$.
This curve (blue) is reduced to Bézier curve (red) $R_m(t)$ of degree 6 using Weighted G^1 method. The corresponding degree reduced Bézier curve plot is depicted in Fig. 4 and the error plot is depicted in Fig. 5.

Example 3: Given the Bézier curve $P_n(t)$ of degree 13 with double loop control points, see [10]:
$\mathbf{P}_0 = (4, 9)$, $\mathbf{P}_1 = (23, 2)$, $\mathbf{P}_2 = (49, 19)$, $\mathbf{P}_3 = (67, 20)$, $\mathbf{P}_4 = (52, 48)$,
$\mathbf{P}_5 = (0, 23)$, $\mathbf{P}_6 = (26, 0)$, $\mathbf{P}_7 = (71, 4)$, $\mathbf{P}_8 = (71, 37)$, $\mathbf{P}_9 = (30, 54)$,
$\mathbf{P}_{10} = (4, 25)$, $\mathbf{P}_{11} = (24, 5)$, $\mathbf{P}_{12} = (41, 0)$, $\mathbf{P}_{13} = (62, 1)$,
This curve (solid-Blue) is reduced to Bézier curve of degree 8

(dashed-Red) using weighted G^1 method, see Fig. 6. Comparing this plot of double loop with the example from [10] shows that our method produces better approximations and makes the loops that other methods did not.

Example 4: This example focuses on a "heart" data set , given a Bézier curve $P_n(t)$ of degree 13 with control points; see [10].
$\mathbf{P}_0 = (22, 10)$, $\mathbf{P}_1 = (37, 5)$, $\mathbf{P}_2 = (86, 18)$, $\mathbf{P}_3 = (81, 23)$, $\mathbf{P}_4 = (69, 56)$,
$\mathbf{P}_5 = (14, 26)$, $\mathbf{P}_6 = (40, 3)$, $\mathbf{P}_7 = (85, 7)$, $\mathbf{P}_8 = (85, 40)$, $\mathbf{P}_9 = (44, 57)$,
$\mathbf{P}_{10} = (18, 29)$, $\mathbf{P}_{11} = (38, 9)$, $\mathbf{P}_{12} = (55, 3)$, $\mathbf{P}_{13} = (77, 5)$.
The heart (solid-Blue) is reduced to Bézier curve of degree 8 (dashed-Red) using weighted G^1-degree reduction. The corresponding degree reduced Bézier curves and the example of heart in [10] are depicted in Fig. 7. Again the plot of double loop example from [10] shows that our method produces better approximations and makes the loops what other methods did not.

The examples show that considering a weight with geometric degree reduction is of great benefit. The results are better than the equivalent methods of C^1/G^k-methods considered by [10].

VI. Conclusion

In this paper, we have presented a method of weighted G^1-degree reduction. The weighted G^1-degree reduction is better than the G^1-degree reduction method in [10]. Referring to the examples in Fig. 1, Fig 3, Fig. 4, Fig 6, and Fig 7, the weighted G^1-degree reduction is the best approximation and provides less error than the linear G^1-degree reduction method and the linear C^1/G^2 method in [10]. The examples in Fig 3, Fig. 4, Fig 6, and Fig 7 show the effectiveness of our proposed weighted G^1-degree reduction method. Our weighted G^1-degree reduction reproduced these loops correctly and is better than existing methods.

Acknowledgment

The authors would like to thank the reviewers for invaluable comments that improve the manuscript.

References

[1] Y. Ahn, B.G. Lee, Y. Park, and J. Yoo, *Constrained polynomial degree reduction in the L_2-norm equals best weighted Euclidean approximation of Bézier coefficients*, Computer Aided Geometric Design 21: 181-191, 2004.

[2] R. Ait-Haddou, *Polynomial degree reduction in the discrete L_2-norm equals best Euclidean approximation of h-Bézier coefficients*, BIT Numer Math. http://dx.doi.org/10.1007/s10543-015-0558-9, 2015.

[3] R.T. Farouki and V.T. Rajan, *Algorithms for polynomials in Bernstein form*, Computer Aided Geometric Design 5: 1-26, 1988.

[4] K. Höllig, J. Hörner, *Approximation and Modeling with B-Splines*, SIAM, Titles in Applied Mathematics 132, 2013.

[5] J. Peters and U. Reif, *Least squares approximation of Bézier coefficients provides best degree reduction in the L_2-norm*, Journal of Approximation Theory 104: 90-97, 2000.

[6] L. Lu and G. Wang, *Optimal multi-degree reduction of Bézier curves with G^2-continuity*, Computer Aided Geometric Design 23: 673-683, 2006.

[7] A. Rababah, *Taylor theorem for planer curves*, Proceedings of the American Mathematical Society, Vol. 119, No.3: 803-810, 1993.

[8] A. Rababah, *Distances with rational triangular Bézier surfaces*, Applied Mathematics and Computation 160, 379-386, 2005.

[9] A. Rababah, B.G. Lee, and J. Yoo, *Multiple degree reduction and elevation of Bézier curves using Jacobi-Bernstein basis transformations*, Numerical Functional Analysis and Optimization 28(9-10): 1179-1196, 2007.

[10] A. Rababah and S. Mann, *Linear Methods for G^1, G^2, and G^3-Multi-Degree Reduction of Bézier Curves*, Computer-Aided Design 45(2):405414, 2013.

[11] P. Woźny and S. Lewanowicz, *Multi-degree reduction of Bézier curves with constraints, using dual Bernstein basis polynomials*, Computer Aided Geometric Design. 26: 566-579, 2009.

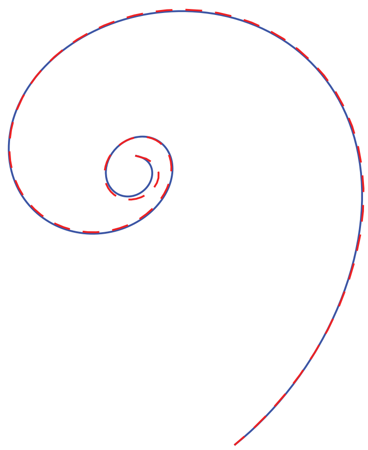

Fig. 1. Spiral Curve: Original curve degree 19 (Solid-Blue); reduced to degree 8 with weighted G^1-degree reduction (dashed-Red).

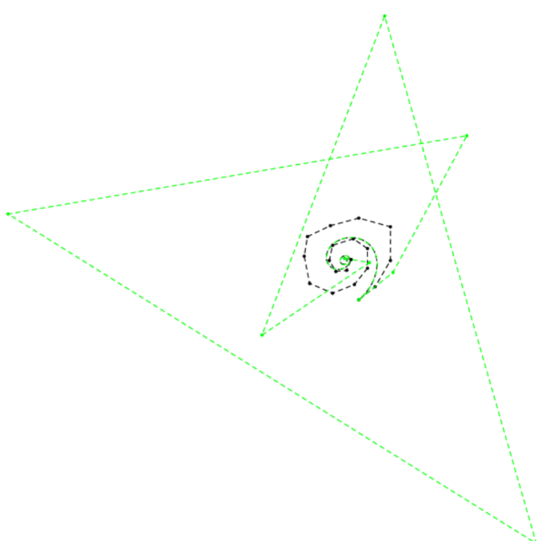

Fig. 2. Curves with control polygons; original curve (dashed-Black); weighted G^1 (dashed-Green).

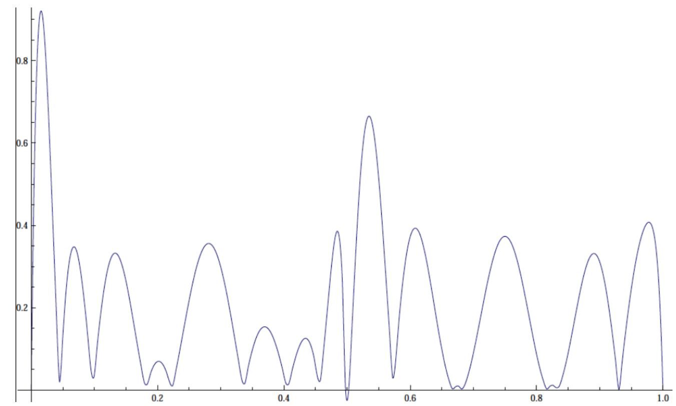

Fig. 3. Error plots for weighted G^1-degree reduction in Example 1.

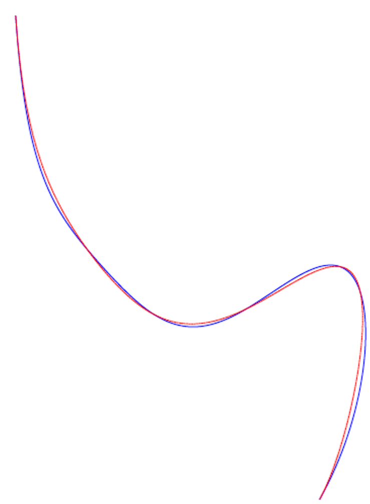

Fig. 4. Curve of degree 10 (Blue) reduced to degree 6 with weighted G^1 method (Red).

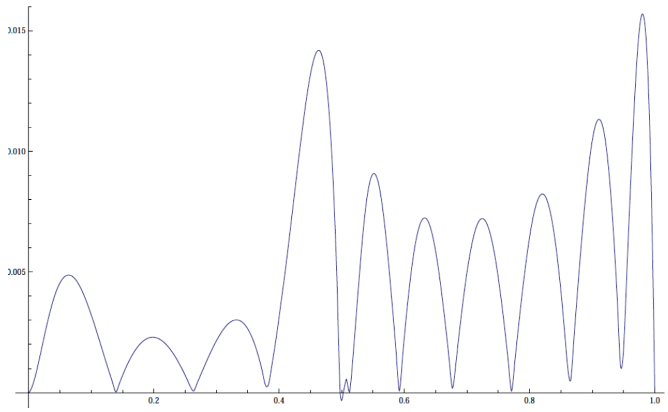

Fig. 5. Error plot for weighted G^1-degree reduction in Example 2.

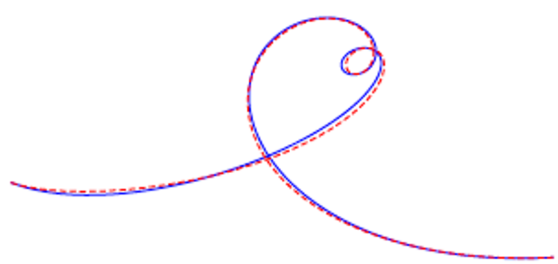

Fig. 6. Original curve in (solid-Blue); weighted G^1-degree 13 reduce to 8 (dashed-Red).

Fig. 7. Original curve in (solid-Blue); weighted G^1-degree 13 reduce to 8 (dashed-Red).