A G^n rational spline with an algebraic distance field

Ágoston Sipos

Budapest University of Technology and Economics

Abstract

Implicit curves have favorable properties for fitting, as they represent isolines of real-valued distance fields which can be interpreted as error functions. This makes it very convenient to use them in e.g. least-squares approximation. However, for most applications, efficient evaluation of curve points (using parameters or explicit formulas) is also required. In this paper we will give a curve class with both representations based on I-Patches, a class of interpolating implicit surfaces with interior control.

1. Introduction

I-Patches[1] are implicit surfaces that guarantee G^1 continuity to *primary* surfaces which are given in implicit form. They have coefficients which control the interior, so theoretically these could be used to fit a surface to given input data, however, a general problem with implicit surfaces is that they are not guaranteed to be a connected, continuous shape for every coefficient setting.

In this paper, our goal is to present a spline curve class based on this 3D implicit surface patch which has the advantage of having both an implicit $f(x,y) = 0$ and an explicit $y = g(x)$ representation where both can be evaluated efficiently: f is a polynomial with relatively low degree, while g is a rational function with the same degree as f in the numerator.

In section 2 we will review I-Patches, in section 3 we will present an algorithm to parameterize some cases of implicit curves. In section 4 we will formulate the interpolating splines. In section 5 and 6 fitting methods and their discussion will be presented.

2. Previous work

I-Patches were originally published by Várady et al.[1] in 2001 to be used in 3D surface modelling.

An I-Patch is an implicit patch defined by n implicit *primary* ($P_1(\mathbf{p}) = 0, \dots, P_n(\mathbf{p}) = 0$) and n implicit *boundary* ($B_1(\mathbf{p}) = 0, \dots, B_n(\mathbf{p}) = 0$) surfaces. ($\mathbf{p} \in \mathbb{R}^3$, arguments will be omitted when the meaning is unambiguous.) The patch equation is

$$I = \sum_{i=1}^{n} \left(w_i P_i \prod_{\substack{j=1 \\ j\neq i}}^{n} B_j^2 \right) + w_c \prod_{j=1}^{n} B_j^2, \qquad (1)$$

where w_i and w_c are scalar constants.

The main properties of the surface deriving from this equation are:

- On each B_i ($i = 1..n$), the I-Patch connects with P_i in first order (G^1) continuity.
- w_c only affects the inside of the patch, as B_j^2 vanishes on the edges.

A three-dimensional I-Patch can be seen in Figure 1 with its primaries and boundaries.

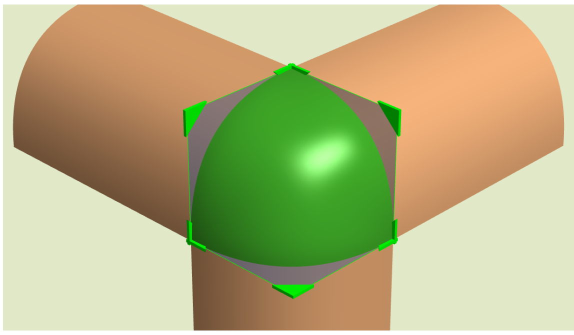

Figure 1: *An I-Patch in 3 dimensions. Brown: primary surfaces, transparent blue: boundary surfaces, green: I-Patch*

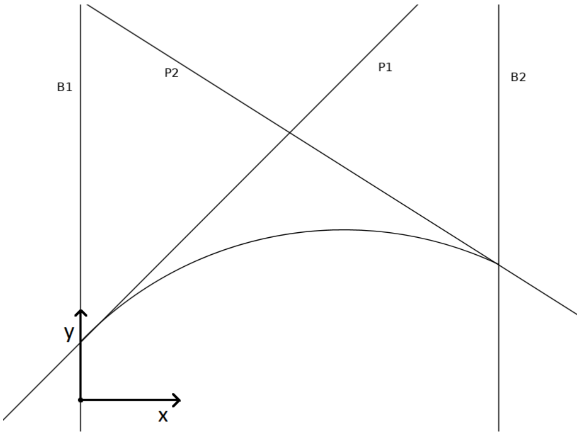

Figure 2: *Explicit I-segment configuration*

Implicit patches can be considered in any number of dimensions easily, in 2 dimensions, the equation will define an implicit **curve**, referred to as an ***I-segment***. As curves have two ends, it should have two primaries and two boundaries, giving an equation

$$I = w_1 P_1 B_2^2 + w_2 P_2 B_1^2 + w_c B_1^2 B_2^2. \tag{2}$$

A spline constructed from I-segments connected to each other will be called an ***I-spline***.

We can easily force higher order continuity: for G^n, the equation has to be

$$I = w_1 P_1 B_2^{n+1} + w_2 P_2 B_1^{n+1} + w_c B_1^{n+1} B_2^{n+1}. \tag{3}$$

3. Parametrization of explicit I-segments

Let us consider the configuration in Figure 2, where B_1 and B_2 are parallel straight lines. In this case we can define a coordinate system, in which the x axis is perpendicular to the boundaries, while y is parallel to them. Suppose, that P_1 and P_2 can be written as explicit functions of x (call it $f_n(x)$).

In this coordinate system, the implicit forms of the primaries and boundaries are $P_n(x,y) = y - f_n(x)$, $B_n(x,y) = B_n(x) = x - x_n$ $(n = 1,2)$. Then we can write up the implicit equation and solve it for y:

$$\sum_{n=1}^{2} \left(w_n (y - f_n(x))(x - x_{\overline{n}})^2 \right) + w_c (x - x_1)^2 (x - x_2)^2 = 0$$

$$y = \frac{\sum_{n=1}^{2} \left(w_n f_n(x)(x - x_{\overline{n}})^2 \right) - w_c (x - x_1)^2 (x - x_2)^2}{\sum_{n=1}^{2} (w_n (x - x_{\overline{n}})^2)}, \tag{4}$$

with the notation $\overline{n} = 3 - n$.

This means that the I-segment can be written as an explicit rational function of x. In the G^n continuous case, it is degree $2(n+1)$ over $(n+1)$.

4. Interpolating splines

Consider the interpolation problem where we have N base points $(x_1, ..., x_N)$ and in each point we have k values $y_i^{(0)}, ..., y_i^{(k-1)}$, the function value and $k-1$ specified derivatives in the ith point. We need a k order continuous interpolating spline.

Our approach is the following:

1. For each pair of data x_i, x_{i+1}, define $B_1 = (x - x_i)$, $B_2 = (x - x_{i+1})$, f_i, f_{i+1} such degree $k-1$ polynomials that they interpolate all specified derivatives in x_i and x_{i+1}: $f_i^{(j)}(x_i) = y_i^{(j)}$, $(j = 0, ..., k-1)$ and let $P_i = y - f_i(x)$
2. Set the I-segment coefficients w_1, w_2, w_c, according to some criteria (see section 5 for examples)
3. We can calculate the implicit form of the curve segment according to (3), and the explicit form using (4).

This will result in a degree $2k$ over k rational function. A well-known alternative solution is to use Hermite-splines[2] which will yield a degree $2k-1$ polynomial. That is simpler to evaluate, but does not have an algebraic distance field, offset curves can only be calculated via value transformation, i.e. $f(x) + c, c \in \mathbb{R}$.

In Figure 3 the difference between algebraic offsets and value-transformed curves can be seen. Our approach gives a better approximation of the distance between a point and the curve (although it is not an exact euclidean metric).

5. Fitting

In this part, two methods for setting the coefficients will be presented.

5.1. C^1 continuity spline

Instead of G^1 continuity, stronger C^1 continuity can be prescribed easily.

As the same P_i implicit curve is used in the interval adjacent to a given point from the right as in the one from the left, if both I-segments are C^1 continuous to this primary, they will be C^1 to each other, too.

Computationally:

$$\overleftarrow{\nabla I \left(x_i, y_i^{(0)} \right)} = w_1^{(i)} \nabla P_i \left(x_i, y_i^{(0)} \right) B_{i+1}^2 \left(x_i, y_i^{(0)} \right), \tag{5}$$

$$\overrightarrow{\nabla I \left(x_i, y_i^{(0)} \right)} = w_2^{(i-1)} \nabla P_i \left(x_i, y_i^{(0)} \right) B_{i-1}^2 \left(x_i, y_i^{(0)} \right), \tag{6}$$

where $w_j^{(i)}$ is the jth coefficient of the curve segment in the interval between x_i and x_{i+1}.

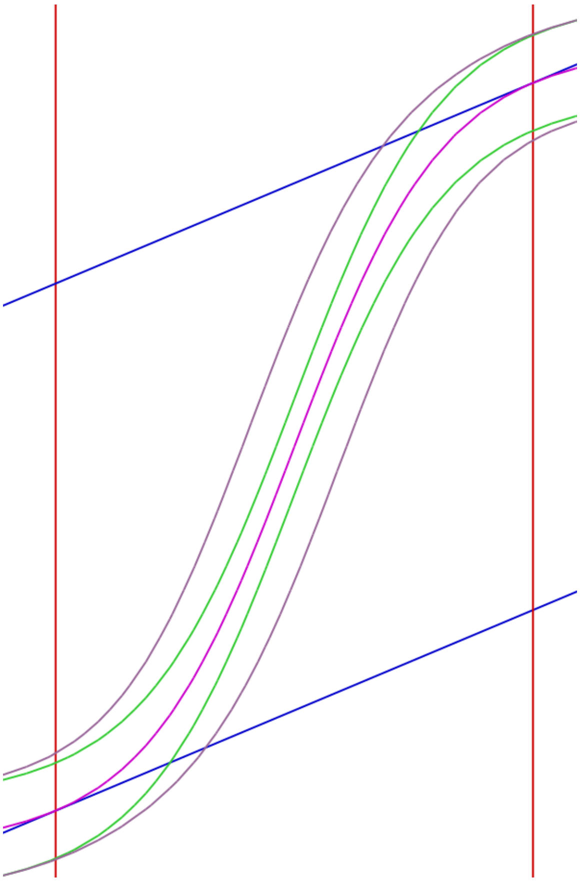

Figure 3: *I-spline segment (purple) and its $I(x,y) = \pm c$ algebraic offset curves (grey) in contrast with the $y = f(x) \pm c$ transformed curves (green). Curve is from approximating a tanh function on $[-1;1]$*

Thus, if we set coefficients so that $w_1^{(i)} B_{i+1}^2 \left(x_i, y_i^{(0)}\right) = w_2^{(i-1)} B_{i-1}^2 \left(x_i, y_i^{(0)}\right)$, C^1 continuity will be achieved. Substituting the boundary equations yields:

$$w_1^{(i)} (x_i - x_{i+1})^2 = w_2^{(i-1)} (x_{i+1} - x_i)^2, \tag{7}$$

$$w_1^{(i)} = w_2^{(i-1)}. \tag{8}$$

This means we lose one degree of freedom in each of the intervals. The conditions do not constrain w_c, so it can still be used for approximation.

5.2. Least squares fitting

As implicit curves are essentially algebraic error functions, it is easy to write up the least squares problem for given data points $\mathbf{p_i} = (x_i, y_i), i = 1..n$, which are in the inside of the interval being considered. This is of course different from

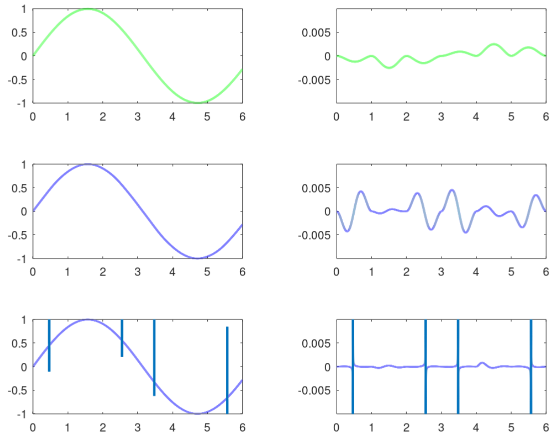

Figure 4: *Fitting splines on synthetic (uniformly sampled) data from a sine function. Top row: cubic Hermite-spline and its error, middle row: C^1 I-spline with midpoint constraint, bottom row: I-spline with two-parameter fitting. Errors are on the same scale.*

the "conventional" least squares result where in each data point the $f(x_i) - y_i$ error is taken into account, our method will consider the algebraic distance of 2D points from the curve.

For each interval, we are using a 3-element polynomial basis and we need the 3 coefficients. However, as the coefficients can be scaled (i.e. if we multiply them with the same scalar we get the same curve), one of them can be fixed to one. When fixing $w_c = 1$ we need to solve the 2x2 matrix equation:

$$\begin{bmatrix} \sum_i F_i^2 & \sum_i F_i G_i \\ \sum_i F_i G_i & \sum_i G_i^2 \end{bmatrix} \cdot \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} = - \begin{bmatrix} \sum_i F_i H_i \\ \sum_i G_i H_i \end{bmatrix}, \tag{9}$$

where

- $F_i = P_1(\mathbf{p_i}) B_2^k(\mathbf{p_i})$
- $G_i = P_2(\mathbf{p_i}) B_1^k(\mathbf{p_i})$
- $H_i = B_1^k(\mathbf{p_i}) B_2^k(\mathbf{p_i})$

If we used the previous method for C^1 continuity or something else that constrains a degree of freedom, a least-squares equation can be written for the w_c coefficient:

$$w_c = -\frac{\sum_i (w_1 P_1(\mathbf{p_i}) B_2^k(\mathbf{p_i}) + w_2 P_2(\mathbf{p_i}) B_1^k(\mathbf{p_i}))}{\sum_i (B_1^k(\mathbf{p_i}) B_2^k(\mathbf{p_i}))}. \tag{10}$$

If we have one point inside each interval, this equation

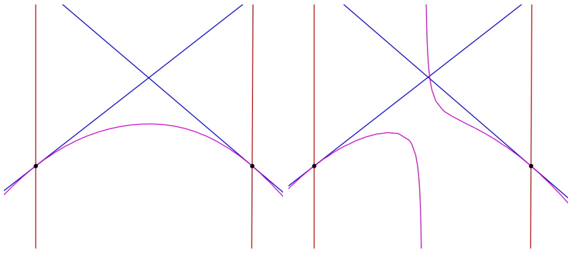

Figure 5: *Left: nonsingular curve, right: singularity in interval*

guarantees that it will be exactly interpolated by the curve. An example for this interpolation is plotted in Figure 4, middle row.

However, direct least squares fitting can often result in a singular curve (with a vertical asymptote, see Figure 4, bottom row). In the next section we give sufficient conditions to prevent this.

6. Discussion

6.1. Singularity

With implicit fitting, the results are often unacceptable, although they fulfill all the continuity requirements because the curve is not guaranteed to be connected.

In our case, the problem arises when the curve has a singularity. The parametric equation can always be evaluated, unless the denominator is zero: $w_1(x-x_2)^k + w_2(x-x_1)^k = 0$.

If k is even (e.g. in the G^1 continuous case), and $w_1w_2 > 0$ this is guaranteed not to happen as the denominator has a constant sign. Otherwise, if $w_1w_2 < 0$ the curve has an infinite discontinuity in $x = \frac{\sqrt[k]{|w_1|}\cdot x_2 - \sqrt[k]{|w_2|}\cdot x_1}{\sqrt[k]{|w_1|} - \sqrt[k]{|w_2|}}$. As in the $2|k$ case $\sqrt[k]{|w_i|} > 0$, this point always falls inside the interval.

The difference between the singular and nonsingular case can be seen in Figure 5.

If k is odd, it is the other way around: as $(x-x_2)^k$ and $(x-x_1)^k$ have different signs, $w_1w_2 > 0$ implies a singularity inside the interval, and $w_1w_2 < 0$ prevents it. G^2 I-splines set by this rule can be seen in Figure 6.

It is interesting to note that while in the G^1 case, the two-parameter least-squares fitting often results in a singular curve, here, the result is always good. It could be interesting future research to find the mathematical reasons for this.

6.2. Higher-order continuity

It has to be noted, that in spite of the nice properties listed, higher-order continuity I-splines ($G^n, n \geq 3$) do not seem to be practical because of their rigidity. An example is shown in Figure 7. This rigidness is caused by using a high degree polynomial to represent the curve, but with just a few degrees of freedom.

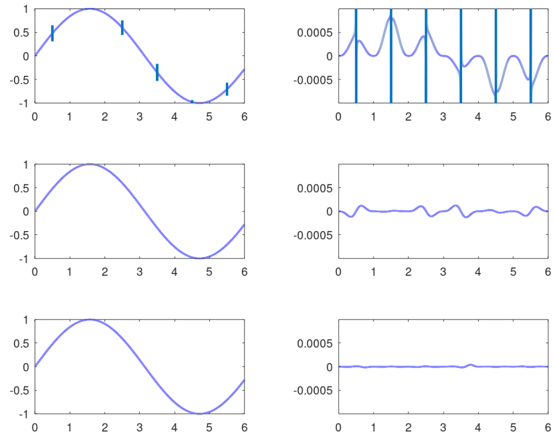

Figure 6: *G^2 I-splines and their errors. Top row: C^1 spline with midpoint interpolation, middle row: flipped coefficients according to Section 6's odd exponent part, bottom row: two-parameter least-squares fitting. Note that error scale is one-tenth of G^1 version's*

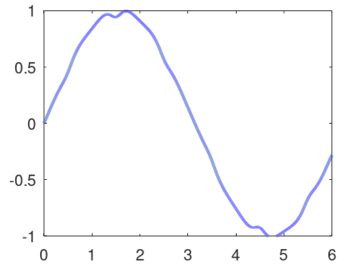

Figure 7: *A too rigid G^3 I-spline. Note that it is everywhere continuous, but unlikely to be very useful.*

Conclusion

We have presented a class of spline curves with a dual (explicit-implicit) representation. They are advantageous because for different interrogations the better suited evaluation can be used. However, we have also pointed out, that guaranteeing that an implicit curve is nonsingular is not always easy, but in this simplified case it can be done with the help of parametrization.

Acknowledgements

This work was supported by the Hungarian Scientific Research Fund (OTKA, No. 124727).

The author thanks Tamás Várady and Péter Salvi for interesting discussions and helpful comments about the article.

References

1. Tamás Várady, Pál Benkő, Géza Kós, and Alyn Rockwood. Implicit surfaces revisited—I-patches. In *Geometric Modelling*, pages 323–335. Springer, 2001.
2. Thomas W Sederberg. Computer Aided Geometric Design. chapter 12.3.2. 2012.