WAIT: Workshop on the Advances in Information Technology, Budapest, 2018

Favorable approximating properties of the P-Bézier basis

István Kovács and Tamás Várady

Budapest University of Technology and Economics, Hungary

Abstract

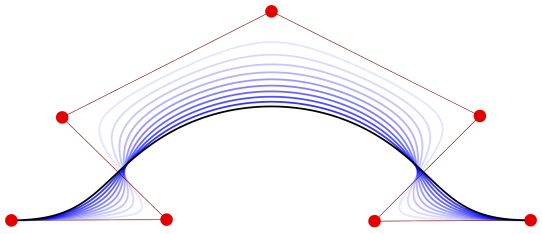
Proximity curves represent a family of curves that are associated with a control point based parametric curve. Proximity curves continuously sweep from the given curve towards its control polygon depending on a proximity value. A new representation, the proximity Bézier (or shortly P-Bézier) curves have been introduced in a recent paper; the basis functions are calculated by a simple algebra with an interesting geometric interpretation. They inherit many features of ordinary Bézier curves including C^∞ continuity in the interior and simple formulae to constrain C^n connections at the end points.
In this article we focus on the approximation properties of this new representation. Our experiments show that the P-Bézier basis is much more suitable for approximation than the standard Bernstein basis; enhancements include a natural placement for the control points and numerical stability for least-squares fitting.

1. Introduction

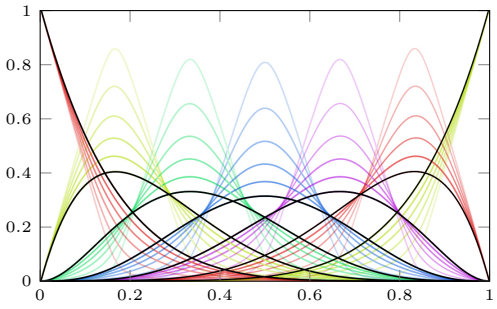
In geometric modelling the great majority of free-form curves and surfaces are defined by control point based representations. The control points are associated with a set of basis functions, that fundamentally determine the versatility of shape editing, the mathematical properties of the scheme and the computational aspects of related algorithms. While the most widely applied Bézier and B-spline schemes possess a well-established theory with a long list of attractive features, it is a recurring research topic to develop new schemes where certain improvements are possible. *Proximity curves* represent such an approach. A family of curves is constructed that are associated with a given control point based parametric curve, and sweep continuously from the curve to its control polygon. The location of the intermediate curves depends on a proximity parameter.

A new representation, the proximity Bézier – or shortly P-Bézier – curve has been introduced in a recent paper[8]. It was developed to overcome some deficiencies of ordinary Bézier curves and offer further flexibility for design. It is well-known that when the degree of a Bézier curve is raised, the weight of the related basis functions decreases, and the design effect of the control point gets weaker. With other words, high degree Bézier curves loosely approximate the control polygons, and large displacements of the control points change the shape only to a small extent. An example is shown in Figure 1(a), where a Bézier curve defined by seven control points (colored black) poorly reproduces the shape of the defining polygon. Now having P-Bézier curves we can adjust the vicinity of the curves to the control polygon and simultaneously set how strongly the control points should affect the shape. In Figure 1(b) the set of basis functions with different proximity values are shown. Note, that these basis function remain C^∞ continuous just their "steepness" over the parametric interval is distributed in different ways.

Various interesting properties and algorithms were presented in our recent paper[8]; here we are going to focus on the problem of approximating data points by means P-Bézier curves. After referring to a few important publications, we will present the basic construction, in particular the creation of the basis functions. These can be plugged into a standard least-squares



(a)



(b)

Figure 1: P-Bézier curves and basis functions with various proximity values. The initial Bézier curve and its Bernstein basis functions are colored black.

approximation algorithm, however, proximity opens up new issues, such as the placement of the control points and numerical stability, as will be discussed and demonstrated in the last part of the paper.

2. Previous work

There are many papers in the CAGD literature that attempt to extend and enhance the most popular representations of Bézier and B-spline curves and surfaces[4]. Significant efforts have been made to offer additional shape control and supplement control point based design[1, 2, 10]. This problem was intensively researched in the late seventies and eighties, nevertheless, some interesting papers were published in the last decade, as well[11, 3, 6, 12].

Our first proximity curve scheme – called P-curves[7] – offers global shape control, ensures C^∞ continuity and G^1 endpoint interpolation. The basis functions were constructed by means of generalized barycentric coordinates.[5]

The new family of P-Bézier and P-B-spline curves takes a different approach[8]: the basis functions are calculated by a much simpler algebra with a special geometric interpretation. We can reproduce standard curve representations and add proximity control. We can also easily maintain C^n end constraints. An interesting feature of proximity curves relates to the insertion of new control points: while in the standard schemes a new degree of freedom leads to repositioning the existing control points, in our case the new control points are always placed on some chord of the control polygon.

Approximation or interpolation with polynomials can be numerically unstable. One can get better results by using the Bernstein basis instead of the monomial basis.[9] Another possible workaround here is to use splines instead of polynomials.[4]; although these are more stable we loose high-degree continuity. With P-Bézier curves, the approximation is much more stable and the basis preserves C^∞ continuity.

3. P-Bézier curves

P-Bézier curves represent a family of curves being associated with an ordinary Bézier curve and a proximity parameter γ. For $\gamma = 1$ the Bézier curve is fully reproduced, for $\gamma = 0$ the curve is snapped onto the control polygon and for other values $0 < \gamma < 1$ intermediate curves are obtained. Note that γ is inversely related to proximity, i.e. for smaller values of γ the curve is dragged closer to the control polygon.

The equation of a P-Bézier curve can be formulated as

$$c^\gamma(t) = \sum_{i=0}^{n} \mathbf{P}_i M_i^{n,\gamma}(t), \tag{1}$$

where $M_0^{n,\gamma}(t), \ldots, M_n^{n,\gamma}(t)$ are special basis functions that depend on γ. (From now on we will omit the superscript n, and use only M_i^γ.)

We will use a sequence of *footpoints* $0 = u_0 < u_1 < \ldots < u_n = 1$, that denote the parametric image of the control points. The u_i-s are set to i/n, however, setting these in a non-uniform manner is also possible. Footpoints actually indicate the maximum position of the basis functions.

We are going to construct the $M_i^\gamma(t)$-s in a special manner by means of a particular set of variables $r_i^\gamma(t)$ and related weighting functions $f_i(t)$:

$$r_i^\gamma = r_i^\gamma(t) = \sqrt{(u_i - t)^2 + \gamma f_i(t)}. \tag{2}$$

The f_i-s are positive, real functions, and will have a crucial role to satisfy our constraints.

The general form of our basis functions are given as

$$\begin{aligned} M_0^\gamma(t) &= \frac{1}{2} + \frac{\Delta r_0}{2\Delta u_0}, \\ M_i^\gamma(t) &= \frac{\Delta r_i}{2\Delta u_i} - \frac{\Delta r_{i-1}}{2\Delta u_{i-1}}, \\ M_n^\gamma(t) &= \frac{1}{2} - \frac{\Delta r_{n-1}}{2\Delta u_{n-1}}, \end{aligned} \tag{3}$$

where $\Delta r_i = r_{i+1}^\gamma - r_i^\gamma$ and $\Delta u_i = u_{i+1} - u_i$.

It is easy to prove that the control polygon is reproduced for $\gamma = 0$, since $r_i(t) = |u_i - t|$, and it is easy to see that the basis $M_i^\gamma(t)$ has the partition of unity property. We are going to reproduce the Bernstein basis, so we seek weighting functions $f_i(t)$ such that for $\gamma = 1$,

$$M_i^1(t) = B_i^n(t) = \binom{n}{i} t^i (1-t)^{n-i}. \tag{4}$$

The solution of this system of equations yield the r_i^1 functions given below:

$$r_i^1(t) = t - \frac{i}{n} + \frac{2}{n} \sum_{j=0}^{i-1} (i-j) B_j(t). \tag{5}$$

Using these we can express the f_i-s as

$$f_i(t) = \left(t - \frac{i}{n} + \frac{2}{n} \sum_{j=0}^{i-1} (i-j) B_j(t) \right)^2 - \left(\frac{i}{n} - t \right)^2, \tag{6}$$

and derive r_i^γ-s for arbitrary γ as

$$r_i^\gamma(t) = \sqrt{\left(\frac{i}{n} - t \right)^2 + \gamma f_i(t)}. \tag{7}$$

Observe that the f_i-s are polynomials of degree $2n$, so the r_i^γ-s are given as square roots of degree $2n$ polynomials.

To sum it up, this construction ensures that using these weighting functions, the related r_i^γ-s and M_i^γ-s reproduce the Bernstein basis for $\gamma = 1$. For $0 < \gamma < 1$ the basis functions are positive, thus the P-Bézier curves represent a convex combination scheme inheriting important properties of Bézier curves. Since the f_i-s and the r_i^γ-s are C^∞ continuous, the P-Bézier curves have C^∞ continuity, as well.

4. Least-squares approximation

Least-squares approximation is a well-known technique. We wish to determine a parametric curve $c(t)$, that approximates a sequence of data points $\mathbf{Q}_0, \ldots, \mathbf{Q}_m$ with assigned parameter values t_j. We minimize the total squared distance error δ, and calculate the unknown control points of the curve denoted by $\mathbf{P}_0, \ldots, \mathbf{P}_n$. For P-Bézier curves with proximity parameter γ we obtain

$$\begin{aligned} \delta &= \sum_{j=0}^{m} \| c^\gamma(t_j) - \mathbf{Q}_j \|^2 = \\ &= \sum_{j=0}^{m} \left\| \sum_{i=0}^{n} \mathbf{P}_i M_i^{n,\gamma}(t_j) - \mathbf{Q}_j \right\|^2 \to \min, \end{aligned} \tag{8}$$

that leads to a linear system of equations. This can be written in matrix form as

$$M^\top M \mathbf{P} = M^\top \mathbf{Q} \tag{9}$$

where $\mathbf{P}$ is the vector of the control points, $\mathbf{Q}$ is the vector of data points and M is the collocation matrix, i.e.

$$M_{i,j} = M_i^{n,\gamma}(t_j). \tag{10}$$

By solving this linear system, we obtain the unknown $\mathbf{P}_i$-s.

The variance of the control points is highly correlated with the norm of $(M^\top M)^{-1}$; and we will analyse these values later in our test examples.

4.1. Parameter correction

In order to get better approximation it is advised to optimize the t_j parameters, too. This can be done by an iterative algorithm to tweak each parameter in gradient direction, and then recalculate the best-fit control points.

More precisely, let the new $t_j' = t_j + \Delta t_j$, where

$$\Delta t_j = \frac{\langle \mathbf{Q}_j - c(t_j), \dot{c}(t_j) \rangle}{\| \dot{c}(t_j) \|^2}. \tag{11}$$

4.2. Proximity correction

With a similar logic we can optimize γ, as well, i.e. find the best possible proximity value for a given set of data points. Let $\gamma' = \gamma + \Delta\gamma$, where

$$\Delta\gamma = \frac{1}{m+1} \sum_{j=0}^{m} \frac{\langle \mathbf{Q}_j - c(t_j), \frac{\mathrm{d}}{\mathrm{d}\gamma} c(t_j) \rangle}{\left\| \frac{\mathrm{d}}{\mathrm{d}\gamma} c(t_j) \right\|^2}. \tag{12}$$

We have achieved the most accurate results when we optimized both γ and t_j, iteratively. So first a t_j parameter correction was performed, then the control points were recalculated, then a γ correction, then recalculating the control points, etc.

5. Numerical stability

As it is well-known, the numerical stability of an approximation or interpolation system of equations depends on the properities of the collocation matrix. Ex-

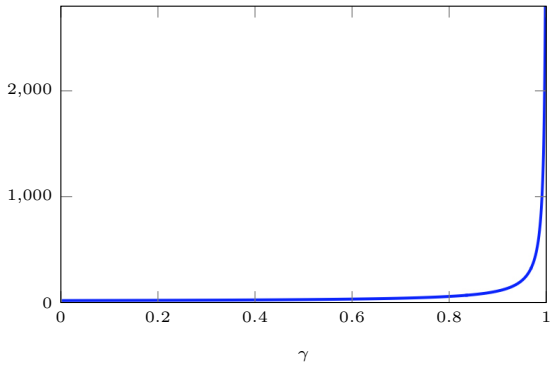


Figure 2: The condition number as the function of γ for $n = 10$. We can see a sharp drop around $\gamma = 1$.

perimental results show that if we change the Bernstein basis according to our proximity method, the approximation becomes much more stable, even for a small tweak in the proximity (for example using $\gamma = 0.98$ instead of 1). In fact, considering the condition number as a function of γ, there is a sharp drop in the vicinity of 1. See Figure 2.

Our main interest was the spectrum of the uniform collocation matrix,

$$M_{i,j} = M_i^{n,\gamma}(t_j), \tag{13}$$

where $t_j = \frac{j}{n}$. First of all, the largest eigenvalue of this matrix is always 1. For $\gamma = 1$ – where the Bernstein basis is reproduced – the smallest eigenvalue of the matrix is

$$\varepsilon_n = \frac{(n-1)!}{n^{n-1}} \approx \frac{\sqrt{2\pi n^3}}{e^n}$$

which is exponentially small as n grows. Since the condition number of a matrix can be expressed as the ratio of the largest and the smallest eigenvalue, the condition number of our matrix is the reciprocal value of the smallest eigenvalue, which is exponentially large as n grows.

However, for $\gamma = 0$ the collocation matrix is just the identity matrix, so all of its eigenvalues are 1. Heuristically what happens here is that the smallest eigenvalue goes from the really small value ε_n to 1, as γ goes from 1 to 0. So if we think of the smallest eigenvalue as a function of γ, this function is more or less linear around $\gamma = 1$. Therefore, the smallest eigenvalue can be expressed (approximately) as

$$\frac{c_n}{1 - \gamma - \varepsilon_n}$$

for some c_n constant. Since ε_n is small, this function behaves around 1 as the function $1/x$ around 0, can explains the sharp drop in the condition number.

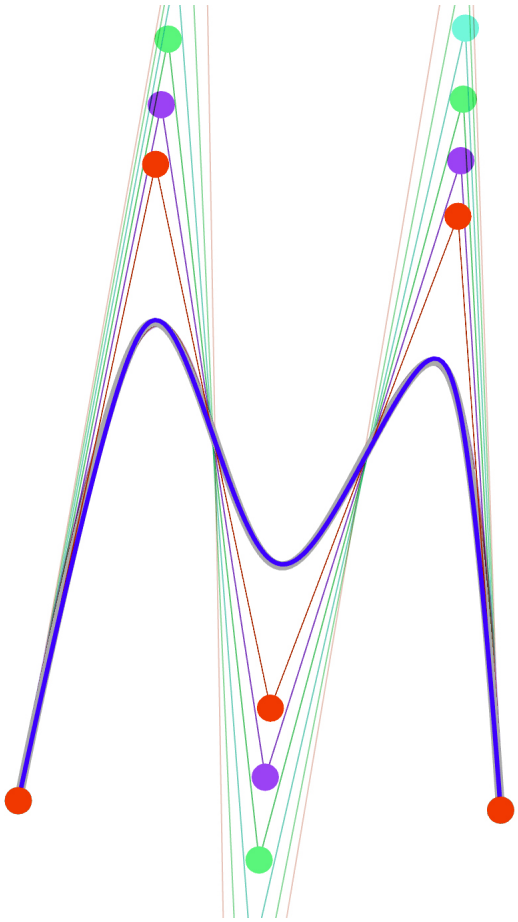


Figure 3: Approximation by P-Bézier curves using various proximity values.

6. Test examples

We demonstrate the benefits of using P-Bézier curves for approximation by means of two simple examples. Parameter correction has been applied.

6.1. Example 1

In the first example we have chosen data points with a relatively high curvature oscillation (see Figure 3). It can be seen that the least-squares fit by an ordinary, low-degree Bézier curve produces extreme, wiggling control points. The control points of P-Bézier curves with high proximity still wiggle. But when we further decrease the proximity value (see Table 1) the best-fit control polygon moves closer to the sample points, and this is favourable for geometric design and the forthcoming computations. In this example the accuracy of the approximation has improved and the matrix norm decreased.

Curve	**Error**	$\|(M^\top M)^{-1}\|$
Bézier ($n = 4$)	1.9%	1.35
P-Bézier ($n = 4, \gamma = 0.8$)	1.83%	0.68
P-Bézier ($n = 4, \gamma = 0.6$)	1.41%	0.42
P-Bézier ($n = 4, \gamma = 0.4$)	0.86%	0.23
P-Bézier ($n = 4, \gamma = 0.2$)	0.70%	0.13

Table 1: Numerical results of approximations in Example 1.

6.2. Example 2

In this example we have sampled a more complex curve and superimposed noise. As before, the data points were approximated by a Bézier and several P-Bézier curves, see Figure 4 and Table 2.

We can see, that in the case of Bézier curves, the matrix norm is extremely high, due to the fact that the Bernstein-Vandermonde matrix is ill-conditioned.[9] It is a surprising observation in Table 2 that the norm is exponentially falling as we linearly decrease the proximity parameter. This indicates that our matrices are numerically more stable, and the variation of the control points is much more smaller. When γ is close to one, we get a very similar basis and shape, but with significantly better numerical stability.

Figure 4(a) shows that the fitted Bézier curve is not appropriate for further editing, since the control points strongly oscillate and their placement is not intuitive, however, with a decreasing γ much better curve candidates are obtained. Note, that as we decrease proximity the approximation error starts increasing below a certain value. This is expected, as having "too rigid" basis functions will destroy the approximating power; so the question arises, for a given point set which proximity curve is the best. As was written in Section 4.2, the optimal value can be iteratively determined. For example, for the curve in Figure 3 the most accurate curve can be obtained at proximity value $\gamma = 0.85$.

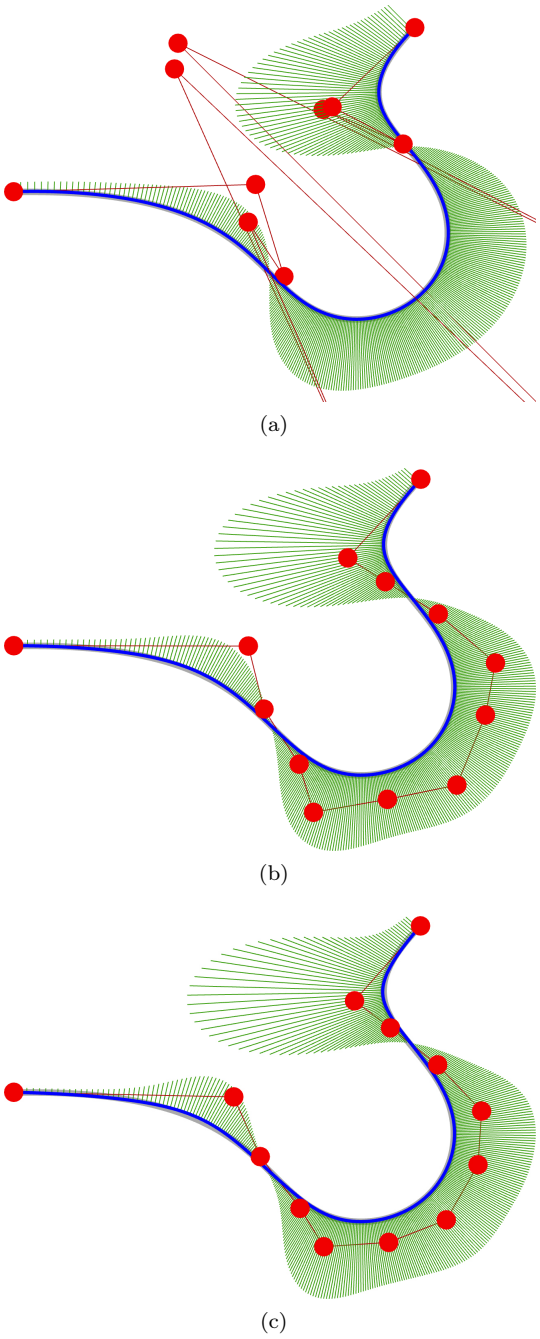


Figure 4: Curve approximations in Example 2, (a) Bézier, (b) P-Bézier ($\gamma = 0.8$), (c) P-Bézier ($\gamma = 0.6$).

7. Conclusion and future work

Recently we have proposed a new concept for parametric curve and surface representation[8], where proximity between a given shape and its control structure can be explicitly controlled. The scheme is capable to reproduce Bézier curves and surfaces, while it offers additional flexibility for geometric design, as well. In our current article we have examined the benefits of applying P-Bézier curves for approximating data points

Curve	**Error**	$\|(M^\top M)^{-1}\|$
Bézier ($n = 12$)	0.38%	242 312.0
P-Bézier ($n = 12, \gamma = 0.8$)	0.61%	308.2
P-Bézier ($n = 12, \gamma = 0.6$)	0.92%	43.8
P-Bézier ($n = 12, \gamma = 0.4$)	1.24%	14.1

Table 2: Numerical results of approximations in Example 2.

and demonstrated the efficiency of the scheme by a few examples.

Acknowledgements

This project has been supported by the Hungarian Scientific Research Fund (OTKA, No. 124727). We highly appreciate several inspiring technical discussions with our colleagues, Péter Salvi and Márton Vaitkus.

References

1. B. A. Barsky and J. C. Beatty. Local control of bias and tension in beta-splines. *ACM Transactions on Graphics (TOG)*, 2(2):109–134, 1983.
2. W. Boehm. Curvature continuous curves and surfaces. *Computer Aided Geometric Design*, 2(4):313–323, 1985.
3. J. Chen. Quasi-Bézier curves with shape parameters. *Journal of Applied Mathematics*, 2013:1–9, 2013.
4. G. E. Farin. *Curves and surfaces for CAGD: a practical guide*. Morgan Kaufmann, 2002.
5. M. S. Floater. Generalized barycentric coordinates and applications. *Acta Numerica*, 24:161–214, 2015.
6. J. Kosinka, M. A. Sabin, and N. A. Dodgson. Semi-sharp creases on subdivision curves and surfaces. In *Computer Graphics Forum*, volume 33, pages 217–226. Wiley Online Library, 2014.
7. I. Kovács and T. Várady. P-curves and surfaces: Parametric design with global fullness control. *Computer-Aided Design*, 90:113–122, 2017.
8. I. Kovács and T. Várady. P-Bézier and P-Bspline curves – new representations with proximity control. *Accepted*, 2018.
9. A. Marco, J.-J. Martı, et al. A fast and accurate algorithm for solving bernstein–vandermonde linear systems. *Linear algebra and its applications*, 422(2):616–628, 2007.
10. G. M. Nielson. Some piecewise polynomial alternatives to splines under tension. *In: R. E. Barnhill, R. F. Riesenfeld, Computer Aided Geometric Design*, pages 209–236, 1974.
11. T. Xiang, Z. Liu, W. Wang, and P. Jiang. A novel extension of Bézier curves and surfaces of the same degree. *Journal of Information and Computational Science*, 7(10):2080–2089, 2010.
12. L. Yan. Adjustable Bézier curves with simple geometric continuity conditions. *Mathematical and Computational Applications*, 21(4):44, 2016.