HOPI: A Novel High Order Parametric Interpolation in 2D

Jia Xu[1,2] and Zhiyong Huang[1,2]

[1]Department of Computer Science, School of Computing, National University of Singapore
[2]{xujia, huangzy}@comp.nus.edu.sg

Abstract
*This paper presents a novel smooth and convergent high order parametric interpolation method called **HOPI** with a formal treatment. It employs high order derivatives information and provides more freedom on control of curves. It can be applied to shape design and analysis using curves. This paper reports the work in 2D.*

Categories and Subject Descriptors (according to ACM CCS): G.1.1 [Numerical Analysis]: Interpolation formulas, Spline and piecewise polynomial interpolation I.3.5 [Computer Graphics]: Curve, surface, solid, and object representations

1. Introduction

Parametric representation is a classical problem in computer aided geometric design. Many approaches have been developed. The problem can be described as: given a distinct sample points set **P**, find a smooth function F which interpolates or approximates all points and agrees with the local information associated with each **Differential Point** if any. This local information is the derivative of different order at that point.

We propose a novel smooth and convergent high order interpolation method called **HOPI**, which exploits the derivative information based on **Taylor Theorem**. Our solution separates the derivative information (eg, normal) from neighborhood providing more freedom for user control and capability to represent very complicated curves. The derivative information could be estimated from neighboring points cloud, specified by user, or computed from some complex equations. So it can be used in both curve design and curve representation.

Compared with existing interpolation methods like B-Splines [[SP95], [HL93]], Hermite Interpolation [[BB98], [Hil56], [Sze75]] and Catmull-Rom [CR74], our interpolation function can be easily constructed (equation 2, section 2.5 and 2.6.1) without solving a large linear system or a complicated recursion. We present two ways to construct a basis I function of order m. One is to perform transformation onto a function in $\mathcal{A}_m$; the other more general method is to use about $\log_2 m$ steps of compositions based on the iteration theory 2.10. Then we use the basis I function to construct all I_i^m in equation 2. It requires no further conditions on the input sample data for smooth connection of neighboring pieces as the choice of I_i function guarantees the smoothness. The representation is more compact for a curve given the same error bound (section 2.4.2). And our method has a good locality property: it requires only 2 sample points to define a piece of curve (equation 9) due to the use of differential sample points with more information rather than only the positions.

2. High Order Interpolation of 2D points

This section begins with a formal definition of the problem. Then we present a framework and the smoothness and convergence requirements in a formal way. Finally, we provide some complete solutions by constructing T_i and I_i functions which satisfy the smoothness and convergence conditions.

2.1. Formal Description of Problem

Definition *(**HOPI** problem)* Given a set **P** of n plane points with derivatives up to order m,

$$\mathbf{P} = \{p_i | 1 \le i \le n, p_i = (x_i, y_i^0, y_i^1, y_i^2 \dots y_i^m)\},$$

where $a = x_1 < x_2 < \cdots < x_n = b$, find a function $F: [a,b] \to \mathbf{R}$, such that

$$\forall 1 \le i \le n, \forall 0 \le k \le m, F^{(k)}(x_i) = y_i^k. \qquad (1)$$

We denote this problem as $\mathbf{HOPI_n^m}$. Here **HOPI** stands for "high order parametric interpolation", n is the number of points and m is the highest order of derivatives.

Remark $\mathbf{HOPI_n^0}$ is the normal historic interpolation problem. $\mathbf{HOPI_1^m}$ is resolved by **Taylor Theorem**.

Remark **HOPI** preserves local extrema and critical points.

2.2. Structure of Solution

Assume $f\colon [a,b] \to \mathbf{R}$ is the target function implied by the points set $\mathbf{P}$, which we attempt to approximate. We define our interpolation function $F^f_{n,m}\colon [a,b] \to \mathbf{R}$

$$F^f_{n,m}(x) = \sum_{i=1}^{n} T_i^m(x) \cdot I_i^m(x). \qquad (2)$$

Or simply, $F^f_{n,m}(x) = \sum_{i=1}^{n} T_i(x) \cdot I_i(x)$. $F^f_{n,m}$ may also be denoted by $F_{\mathbf{P}}$ or $F_{\mathbf{P_n^m}}$.

Here, T_i captures all local information associated with point p_i. And I_i works like a weighting function which specifies the range and extent of influence of T_i on the neighborhood.

The underlying principles are, given $\mathbf{P}$ is all information we know about f besides the assumption that f is smooth to some extent (for some m, $f \in C^{(m)}$):

- Each sample point carries full information we can know for sure of f at that point;
- Each sample point has only local influence (the ability to predict/approximate the status of f on neighborhood) on the implied function f. The more information (eg. derivatives) we know about a sample point, the larger influence the sample point has on f.

Consequently, in our approach, local modifications, like changing, removing or adding points have only local effects.

In order to satisfy condition (1), we assume functions T_i and functions I_i satisfy the following conditions :

$$\forall 1 \le i \le n, \forall 0 \le k \le m, T_i^{(k)}(x_i) = y_i^k. \qquad (3)$$

$$\forall 1 \le i, j \le n, I_i(x_j) = \delta_{ij}. \qquad (4)$$

$$\forall 1 \le i, j \le n, \forall 1 \le k \le m, I_i^{(k)}(x_j) = 0. \qquad (5)$$

Theorem 2.1 *(Interpolation)* If T_i satisfy condition (3) and I_i satisfy conditions (4), (5), then F satisfies condition (1), i.e, F is an interpolation function of points set $\mathbf{P}$.

2.3. Smoothness

Theorem 2.2 *(Smoothness)* If $\forall 1 \le i \le n, T_i \in C^{(m)}[a,b]$, and $I_i \in C^{(m)}[a,b]$, then $F \in C^{(m)}[a,b]$.

2.4. Convergence

Definition *(Sampling of f)* Given $f\colon [a,b] \to \mathbf{R} \in C^{(m)}[a,b]$ and n, m, define $\mathbf{P_n^m}(f)$

$$\mathbf{P_n^m}(f) = \{p_i | 1 \le i \le n, p_i = (x_i, f(x_i), f'(x_i) \dots f^{(m)}(x_i))\},$$

where x_i are any numbers such that $a = x_1 < x_2 < x_3 \cdots < x_n = b$. Let $\Delta_i = x_{i+1} - x_i, i = 1,2,3 \dots n-1$. $\Delta_{max} = \max_{1 \le i < n} \Delta_i$.

Definition *($\{F_n\}$ and $\{F^m\}$)* Given $\mathbf{P_n^m}(f)$ with Δ_{max}, we construct an interpolation function $F_{\mathbf{P_n^m}(f)}$ using equation (2). If we fix m, $\forall n \ge 2$, we can construct a (not unique) function $F_n = F_{\mathbf{P_n^m}(f)}$. These functions form a sequence of functions $\{F_n\}$. Similarly, if we fix n, $\forall m \ge 0$, we can construct a (not unique) function $F^m = F_{\mathbf{P_n^m}(f)}$. These functions form another sequence of functions $\{F^m\}$.

2.4.1. $\mathbf{F^f_{n,m}}$ converges to f with respect to n

We find our previous assumptions (3), (4), (5) are not enough to guarantee the convergence of F. We need more conditions:

$$\forall 1 \le i \le n, \forall x \in [a,b], f(x) = T_i(x) + o\left((x - x_i)^m\right). \qquad (6)$$

$$\forall 1 \le i < n, \forall x \in [x_i, x_{i+1}], I_i(x) + I_{i+1}(x) = 1. \qquad (7)$$

$$\forall 1 \le i \le n, \forall x \in [a,b], 0 \le I_i(x) \le 1. \qquad (8)$$

Exhausting the principle of localization, we restrict the influence of T_i to its neighborhood.

$$\forall 1 \le i \le n, \forall x \in [a,b] \backslash (x_{i-1}, x_{i+1}), I_i(x) = 0, \qquad (9)$$

where $x_0 = x_1, x_{n+1} = x_n$. The consequence of this condition is that F is a piecewise function and only 2 neighboring sample points define a piece.

Theorem 2.3 Given $\mathbf{P_n^m}(f)$ with Δ_{max}, if T_i satisfy condition (6) and I_i satisfy conditions (7) (8) (9), then $\{F_n\}$ uniformly converges to f as $\Delta_{max} \to 0$ (so $n \to +\infty$).

Remark The speed of convergence, which is polynomial in Δ_{max} with order m, depends on m. The larger m is, the faster convergence is.

Remark The condition (6) is actually the assumption that $\mathbf{HOPI_1^m}$ converges.

2.4.2. $\mathbf{F^f_{n,m}}$ converges to f with respect to m

Theorem 2.4 Given $\mathbf{P_n^m}(f)$ with $\Delta_{max} < 1$, if T_i satisfy condition (6) and I_i satisfy conditions (7) (8) (9), then $\{F^m\}$ uniformly converges to f as $m \to +\infty$.

Remark If T_i is **Taylor Expansion**, the condition $\Delta_{max} < 1$ can be released to Δ_{max} is bounded.

Remark The speed of convergence, which is exponential with base Δ_{max}, depends on Δ_{max} (so on n indirectly). The smaller Δ_{max} is (the larger n is), the faster the convergence is.

Remark There are two approaches to represent a curve or to approximate an implied complex function: 1. Know a few (eg, only position) at a lot of points; 2. Know a lot (eg, differential point with derivatives) at a few points. Approach 1 includes most existing parametric presentation methods like B-Splines and $\mathbf{HOPI_n^{m_0}}$ with fixed m_0. And such methods with interpolation function S, usually have polynomial convergence speed to the target curve function f, i.e $|S(\mathbf{x}) - f(\mathbf{x})| = o(\Delta_{max}^M)$ for some M (here the o notation, the same as the one appears in condition (6), is the little o notation in calculus). While $\mathbf{HOPI_{n_0}^m}$ with fixed n_0 converges to the target exponentially. Hence we claim, our method could have a more compact data size to represent a curve given the same error bound.

2.5. Find function T: $\mathbf{HOPI_1^m}$

Finding function $T \in C^{(m)}[a,b]$ compliant to conditions (3)(6) is actually the general problem $\mathbf{HOPI_1^m}$.

Theorem 2.5 Taylor expansion of order m at $x = x_i$ satisfy conditions (3)(6).

$$\forall 1 \le i \le n, T_i(x) = \sum_{k=0}^{m} \frac{f^{(k)}(x_i)}{k!}(x - x_i)^k = \sum_{k=0}^{m} \frac{y_i^k}{k!}(x - x_i)^k.$$

Theorem 2.6 If $p_n(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n$ is a polynomial of order n, and T_i is the **Taylor expansion**, then $\forall m \ge n, l > 1, F_{l,m}^{p_n(x)} = p_n(x)$.

2.6. Find function I: $\mathbf{HOPI_2^m}$

Finding function $I \in C^{(m)}[a,b]$ compliant to (4)(5)(7)(8)(9) is actually solving an instance of $\mathbf{HOPI_2^m}$.

In this section, we first construct a common basis I function. Then construct the general I_i functions by applying transformation onto the basis I function. We will provide two methods to construct the basis I function of order m.

Definition *(Function class $\mathcal{A}$)* Define $\mathcal{A}_m$ a set of functions $g\colon [0,1] \to \mathbf{R} \in C^{(m)}[0,1]$, such that

$$g(0) = 1, g(1) = 0;$$
$$\forall 1 \le k \le m, g^{(k)}(0) = g^{(k)}(1) = 0;$$
$$\forall x \in [0,1], g(x) \ge 0.$$

Definition *(Function class $\mathcal{B}$)* Define $\mathcal{B}$ a set of functions $g\colon [0,1] \to \mathbf{R}$, such that

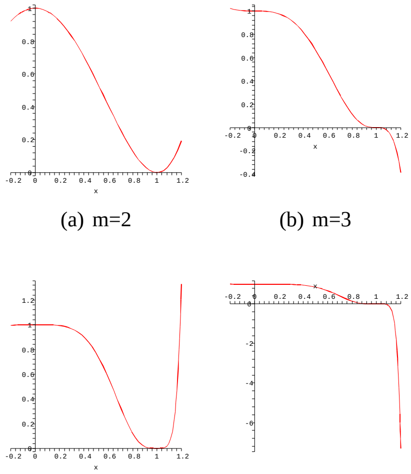
$$\forall x \in [0,1], g(x) + g(1-x) = 1.$$

Definition *(Basis I function of order m)* A basis I function of order m is any function $g\colon [0,1] \to \mathbf{R} \in \mathcal{A}_m \bigcap \mathcal{B}$.

Lemma 2.7 $g(x) = (1 - x^m)^m \in \mathcal{A}_{m-1}$

The curves of $g(x)$ for some small m are shown in Figure 1.

Remark Note $g(x)$ is definitely not the minimal polynomial in $\mathcal{A}_{m-1}$. The minimal polynomial should have order $2m - 1$. But $g(x)$ has a simple explicit expression and is easy to compute.



(a) m=2 (b) m=3

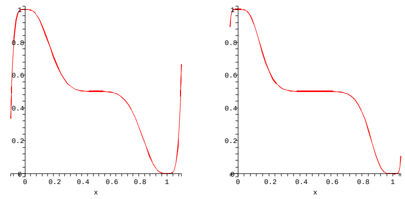
(c) m=4 (d) m=5

Figure 1: *Curves of* $(1 - x^m)^m$

We can construct a function in $\mathcal{A}_m \bigcap \mathcal{B}$ by performing transformation onto a function in $\mathcal{A}_m$.

Theorem 2.8 *(Candidate 1 of basis I function)* If $g\colon [0,1] \to \mathbf{R} \in \mathcal{A}_m$, then $G\colon [0,1] \to \mathbf{R} \in \mathcal{A}_m \bigcap \mathcal{B}$, i.e, G is a basis I function of order m, where G is defined as follow:

$$G(x) = \begin{cases} 1 - \frac{1}{2} g(-2x+1) & \text{if } x \in [0, \frac{1}{2}]; \\ \frac{1}{2} g(2x-1) & \text{if } x \in [\frac{1}{2}, 1]. \end{cases}$$



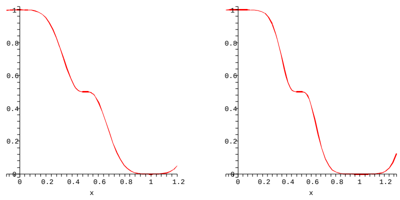
(a) m=4 (b) m=6

Figure 2: *Curves of G with* $g = (1 - x^m)^m$

Theorem 2.9 *(Candidate 2 of basis I function)* If $g\colon [0,1] \to \mathbf{R} \in \mathcal{A}_m$, then $J\colon [0,1] \to \mathbf{R} \in \mathcal{A}_m \bigcap \mathcal{B}$, i.e, J is a basis I function of order m, where J is defined as follow:

$$J(x) = \begin{cases} \frac{1}{2} + \frac{1}{2} g(2x) & \text{if } x \in [0, \frac{1}{2}]; \\ \frac{1}{2} - \frac{1}{2} g(2-2x) & \text{if } x \in [\frac{1}{2}, 1]. \end{cases}$$

Remark **Theorem**(2.9) is almost identical to **Theorem**(2.8) except the way of transformation, more precisely the choice of connect point of two pieces of graph of g. See Figure 2 for **Theorem**(2.8) and Figure 3 for **Theorem**(2.9).



(a) m=4 (b) m=6

Figure 3: *Curves of J with $g = (1-x^m)^m$*

Theorem 2.10 *(Iteration)* If $g \in \mathcal{A}_{m_1} \bigcap \mathcal{B}$, and $h \in \mathcal{A}_{m_2} \bigcap \mathcal{B}$ then $g(h(1-x)) \in \mathcal{A}_{m_1 m_2 + m_1 + m_2} \bigcap \mathcal{B}$.

Remark The significance of this theorem is that we can construct arbitrary higher order basis I function from any low order basis function— finding one means finding all.

Remark This theorem can be proved using **Faà di Bruno's formula**.

Theorem 2.11 *(Candidate 3 of basis I function)* Function $\varphi := \frac{1}{2} + \frac{1}{2}\cos(\pi x) \in \mathcal{A}_1 \bigcap \mathcal{B}$. And $\varphi_m \in \mathcal{A}_{2^m - 1} \bigcap \mathcal{B}$, where φ_n is defined as follow

$$\varphi_1(x) := \varphi(x), \ \varphi_{n+1}(x) := \varphi \circ \varphi_n(1-x),$$

where $\circ$ is the function composition operator.

Remark Note the choice of φ function is not unique. For example, $2x^3 - 3x^2 + 1 \in \mathcal{A}_1 \bigcap \mathcal{B}$ and $-6x^5 + 15x^4 - 10x^3 + 1 \in \mathcal{A}_2 \bigcap \mathcal{B}$. And using these polynomial functions can avoid the numeric problem raised by cosine. Figure 4 shows the graphs of φ_n for some small n.

2.6.1. Construction of I_i function

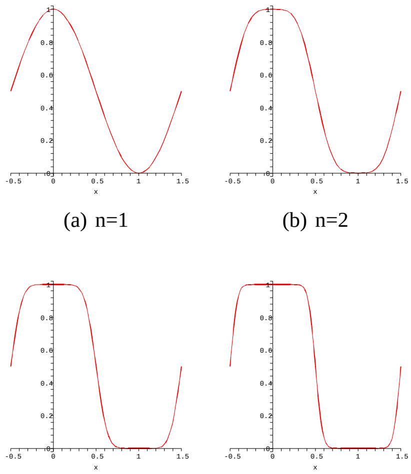
If g is a basis I function of order m, then we define I_i of $\mathbf{HOPI^m_n}$ as follow

$$I_1(x) = \begin{cases} g\left(\frac{x-x_1}{x_2-x_1}\right) & \text{if } x \in [x_1, x_2]; \\ 0 & \text{otherwise.} \end{cases}$$

$$I_n(x) = \begin{cases} g\left(\frac{x_n-x}{x_n-x_{n-1}}\right) & \text{if } x \in [x_{n-1}, x_n]; \\ 0 & \text{otherwise.} \end{cases}$$

$$\forall 1 < i < n, \ I_i(x) = \begin{cases} g\left(\frac{x-x_i}{x_{i+1}-x_i}\right) & \text{if } x \in [x_i, x_{i+1}]; \\ g\left(\frac{x_i-x}{x_i-x_{i-1}}\right) & \text{if } x \in [x_{i-1}, x_i]; \\ 0 & \text{otherwise.} \end{cases}$$

We can verify that the above definition of I_i satisfy all conditions including (4)(5)(7)(8)(9).



(a) n=1 (b) n=2

(c) n=3 (d) n=4

Figure 4: *Curves of φ_n*

3. Conclusion

We propose a novel smooth and convergent interpolation method called **HOPI** which employs derivatives associated with the sample points. Our method constructs the solution to $\mathbf{HOPI^m_n}$ with the solution to $\mathbf{HOPI^m_1}$ and the solution to an instance of $\mathbf{HOPI^m_2}$. This method can be applied to shape design and analysis using curves.

References

[BB98] Bartels R. H.; Beatty J. C., Barsky B. A.: *Hermite and Cubic Spline Interpolation*. Morgan Kaufmann, San Francisco, CA, 1998.

[CR74] Catmull E., Rom R.: A class of local interpolating splines. *Computer Aided Geometric Design* (1974), pp. 317–326.

[Hil56] Hildebrand F. B.: *Introduction to Numerical Analysis*. McGraw-Hill, New York, 1956.

[HL93] Hoschek J., Lasser D.: *Fundamentals of computer aided geometric design*. A. K. Peters, Ltd., Natick, MA, USA, 1993. Translator-Larry L. Schumaker.

[SP95] Shikin E. V., Plis A. I.: *Handbook on Splines for the User*. CRC Press, Inc, Florida, 1995.

[Sze75] Szego G.: *Orthogonal Polynomials, 4th ed.* Amer. Math. Soc., Providence, RI, 1975.